International Scholarly Research Network
ISRN Artificial Intelligence
Volume 2012, Article ID 820364, 12 pages
doi:10.5402/2012/820364

Research Article

Neural Discriminant Models, Bootstrapping, and Simulation

Masaaki Tsujitani,[1] Katsuhiro Iba,[2] and Yusuke Tanaka[3]

[1] *Department of Engineering Informatics, Osaka Electro-Communication University, 18-8 Hatsu-chou, Neyagawa, Osaka 572-8530, Japan*
[2] *Department of Clinical Research and Development, Otsuka Pharmaceutial Co., Ltd., Osaka, Japan*
[3] *Clinical Information Division Data Science Center, EPS Corporation, Japan*

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

Received 8 October 2011; Accepted 30 November 2011

Academic Editor: J. J. Chen

This paper considers the feed-forward neural network models for data of mutually exclusive groups and a set of predictor variables. We take into account the bootstrapping based on information criterion when selecting the optimum number of hidden units for a neural network model and the deviance in order to summarize the measure of goodness-of-fit on fitted neural network models. The bootstrapping is also adapted in order to provide estimates of the bias of the excess error in a prediction rule constructed with training samples. Simulated data from known (true) models are analyzed in order to interpret the results using the neural network. In addition, the thyroid disease database, which compares estimated measures of predictive performance, is examined in both a pure training sample study and in a test sample study, in which the realized test sample apparent error rates associated with a constructed prediction rule are reported. Apartment house data of the metropolitan area station with four-class classification are also analyzed in order to assess the bootstrapping by comparing leaving-one-out cross-validation (CV).

1. Introduction

The neural network model is considered for the multiclass classification problem of assigning each observation into one of multiclass, which is referred to as a *multiple-group neural discriminant model*. As two-class problems are much easier to solve, we focus on neural networks for multiclass classification with respect to statistical techniques in order to derive the maximum likelihood estimators (MLE) [1–7]. Statistical techniques are formulated in terms of the principle of the likelihood of the neural discriminant model, in which the connection weights of the network are treated as unknown parameters.

Besides the theoretical and empirical properties of the bootstrapping [8, 9] in the multiple-group neural discriminant model, there are at least two other reasons to use a bootstrap procedure. First, the criterion based on bootstrapping is demonstrated to be favorable when selecting the optimum number of hidden units. A number of model selection procedures (i.e., methods for the selection of the optimum number of hidden units), such as Akaike information criterion (AIC), BIC (Baysesian information criterion) and cross-validation [10–13] have been proposed. The bootstrap method, however, provides the percentile for the deviance, allowing evaluation of the overall goodness-of-fit and estimation of the bias of the excess error in prediction based on the selected model. Therefore, there is no extra cost for subsequence inference via the bootstrap samples generated for model selection. If a model is selected by a cross-validation method and the bootstrap is used for the subsequence inference, the extra cost of computations is required in resampling for cross-validation. Second, the bootstrap procedures developed in the multiple-group neural discriminant model can be extended, without any theoretical derivation, to more complicated problems such as the generalized additive models (GAM) [14, 15], support vector machines (SVM) [16–19], and vector generalized additive models (VGAM) [20].

The remainder of this paper is organized as follows. In Section 2 we focus on the selection of the optimum number of hidden units and evaluation of the overall goodness-of-fit with the optimum number of hidden units. A neural network can approximate any reasonable function with arbitrary precision if the number of hidden units

tends to infinity [21]. The output of the network fits the training sample too closely if the number of hidden units is increased and the noise is modeled in addition to the desired underlying function. The bootstrapping is also adapted in order to provide estimates of the bias of the excess error in a prediction rule constructed with training samples [22, 23]. Simulated data from known (true) models are used to demonstrate the approximate realization of continuous mapping by neural networks in Section 3. In Section 4 the methods are illustrated using a thyroid disease database in order to show that the overfitting leads poor generalization. Apartment house data of the metropolitan area station with four-class classification are also analyzed in order to assess the bootstrapping by comparing leaving-one-out CV. Finally, in Section 5 we discuss the relative merits and limitations of the methods.

2. Materials and Methods

2.1. Multiple-Group Neural Discriminant Model

2.1.1. Statistical Inference. The functional representation of the neural network model is considered, as shown in Figure 1. The connection weight between the ith unit in the input layer ($i = 0, \ldots, I$) and the jth unit in the hidden layer ($j = 1, \ldots, H$) is α_{ij}. Similarly, the weight between the jth unit in the hidden layer ($j = 0, \ldots, H$) and the kth unit in the output layer ($k = 1, \ldots, K$) is β_{jk}. The input to the jth hidden unit is a linear projection of the input vector $\mathbf{x} = (x_1, \ldots, x_I)$, that is,

$$u_j = \sum_{i=0}^{I} \alpha_{ij} x_i, \quad x_0 \equiv 1, \tag{1}$$

where α_{0j} is a bias. This is the same idea as incorporating the constant term in the design matrix of a regression by including a column of 1's [1]. The output of the jth hidden unit is

$$y_j = f\left(u_j\right) = f\left(\sum_{i=0}^{I} \alpha_{ij} x_i\right), \tag{2}$$

where $f(\cdot)$ is a nonlinear activation function. The most commonly used activation function is the logistic (sigmoid) function:

$$y_j = \frac{1}{1 + \exp\left(-u_j\right)}. \tag{3}$$

The input to the kth output unit is

$$v_k = \sum_{j=0}^{H} \beta_{jk} y_j, \quad y_0 \equiv 1, \tag{4}$$

where β_{0j} is a bias. The activation function of network outputs for the mutually exclusive groups can be achieved using the softmax activation (normalized exponential) function:

$$o_k = \frac{\exp(v_k)}{\sum_{k=1}^{K} \exp(v_k)} = \frac{1}{1 + \exp(-V_k)}, \tag{5}$$

$$V_k = v_k - \ln\left\{\sum_{k' \neq k}^{K} \exp(v_{k'})\right\}, \tag{6}$$

which can be regarded as a multiclass generalization of logistic function.

From (1)–(6), o_k can be written in the form

$$\begin{aligned} o_k &= \frac{\exp(v_k)}{\sum_{k'=1}^{K} \exp(v_{k'})} = \frac{\exp\left(\sum_{j=1}^{H} \beta_{jk} y_j\right)}{\sum_{k'=1}^{K} \exp\left(\sum_{j=0}^{H} \beta_{jk'} y_j\right)} \\ &= \frac{\exp\left\{\sum_{j=0}^{H} \beta_{jk} / \left(1 + \exp\left(-\sum_{i=0}^{I} \alpha_{ij} x_i\right)\right)\right\}}{\sum_{k'=1}^{K} \exp\left\{\sum_{j=0}^{H} \beta_{jk'} / \left(1 + \exp\left(-\sum_{i=0}^{I} \alpha_{ij} x_i\right)\right)\right\}}. \end{aligned} \tag{7}$$

The output o_K to the Kth group can be calculated as $o_K = 1 - \sum_{k=1}^{K-1} o_k$. For example, in the case of $K = 3$, it follows that

$$o_1 = \frac{e^{v_1}}{e^{v_1} + e^{v_2} + e^{v_3}}, \qquad o_2 = \frac{e^{v_2}}{e^{v_1} + e^{v_2} + e^{v_3}}. \tag{8}$$

From $o_1 + o_2 + o_3 = 1$, $o_3 = 1 - (o_1 + o_2)$. Thus the number of unit for output layer is 2 ($= K - 1$).

By setting the teach value

$$t_k^{\langle d \rangle} = \begin{cases} 1: & d\text{th input vector } \mathbf{x}^{\langle d \rangle} = \left(x_1^{\langle d \rangle}, x_2^{\langle d \rangle}, \ldots, x_I^{\langle d \rangle}\right), \\ & \text{is from the } k\text{-group} \\ 0: & \text{o.w.} \end{cases} \tag{9}$$

the log likelihood function for the total sample size D is

$$\begin{aligned} &\ln L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{t}) \\ &\quad = \sum_{d=1}^{D} \sum_{k=1}^{K} t_k^{\langle d \rangle} \ln o_k^{\langle d \rangle}, \quad 0 \leq o_k^{\langle d \rangle} \leq 1, \sum_{k=1}^{K} t_k^{\langle d \rangle} = 1, \end{aligned} \tag{10}$$

where $\mathbf{t}^{\langle d \rangle} = (t_1^{\langle d \rangle}, t_2^{\langle d \rangle}, \ldots, t_K^{\langle d \rangle})$ and $\mathbf{o}^{\langle d \rangle} = (o_1^{\langle d \rangle}, o_2^{\langle d \rangle}, \ldots, o_K^{\langle d \rangle})$ are the teach and output vectors, respectively, for the dth observation, $\mathbf{t} = (\mathbf{t}^{\langle 1 \rangle}, \mathbf{t}^{\langle 2 \rangle}, \ldots, \mathbf{t}^{\langle D \rangle})$, $\mathbf{x} = (\mathbf{x}^{\langle 1 \rangle}, \mathbf{x}^{\langle 2 \rangle}, \ldots, \mathbf{x}^{\langle D \rangle})$, $\boldsymbol{\alpha} = \{\alpha_{ij}\}$, and $\boldsymbol{\beta} = \{\beta_{jk}\}$. As usual, the negative log likelihood gives the cross-entropy error function. The unknown parameters $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$ can be estimated by maximizing the log likelihood ((10) with output (7)) by use of batch backpropagation including momentum, in which the training values for unknown parameters are chosen at random. The number of parameters included in the multiple-group neural discriminant model is $p = H(I + K - 1) + H + K - 1$.

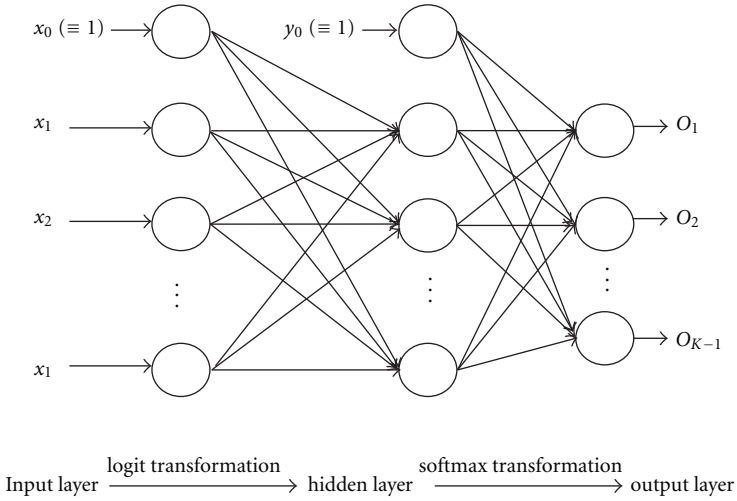


Figure 1: Single hidden layer neural network model.

2.1.2. Determination of the Optimum Number of Hidden Units. The criterion based on bootstrapping is demonstrated to be favorable when selecting the optimum number of hidden units. In conventional statistics, various criteria have been developed for assessing the generalization performance. AIC provides us with a decision as to which of several competing network architectures are best for a given problem. However, the usage of AIC may not be justified theoretically when considering a neural network as an approximation to an underlying model [7, 24]. A bootstrap type nonparametric resampling estimator of Kullback-Leibler information by Ishiguro and Sakamoto [25], Konishi and Kitagawa [26], Ishiguro et al. [27], Kullback and Leibler [28], and Shibata [29] and Shao [30] can provide an alternative to AIC computed from a skewed discrete distribution.

Let the training samples $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_d, \ldots, \mathbf{X}_D\}$, $\mathbf{X}_d = \{\mathbf{x}^{\langle d\rangle}, \mathbf{t}^{\langle d\rangle}\}$, and $\mathbf{x}^{\langle d\rangle} = \{x_1^{\langle d\rangle}, x_2^{\langle d\rangle}, \ldots, x_I^{\langle d\rangle}\}$ for $d = 1, 2, \ldots, D$ be independently distributed in an unknown distribution F. Let $\widehat{F}$ be the empirical distribution function that places a mass equal to $1/D$ at each point $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_d, \ldots, \mathbf{X}_D$. We propose the bootstrap sampling algorithm given as follows.

Step 1. Generate B samples $\mathbf{X}^*$, each of size D, drawn with replacement from the training sample $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_d, \ldots, \mathbf{X}_D\}$. Denote the bth sample as $\mathbf{X}_b^*$, $b = 1, 2, \ldots, B$.

Step 2. For each bootstrap sample $\mathbf{X}_b^*$, $b = 1, 2, \ldots, B$, fit a model to obtain the estimator $\widehat{\boldsymbol{\theta}}(\mathbf{X}_b^*)$.

Step 3. The bootstrap estimator of bias C^* is given as

$$C^* \cong \frac{1}{B}\sum_{b=1}^{B}\left[\ln L\left\{\mathbf{X}_b^*; \widehat{\boldsymbol{\theta}}\left(\mathbf{X}_b^*\right)\right\} - \ln L\left\{\mathbf{X}; \widehat{\boldsymbol{\theta}}\left(\mathbf{X}_b^*\right)\right\}\right], \quad (11)$$

where C^* is the average of differences between log likelihood on the bootstrap sample $\ln L\{\mathbf{X}_b^*; \widehat{\boldsymbol{\theta}}(\mathbf{X}_b^*)\}$ and that on the training sample $\ln L\{\mathbf{X}; \widehat{\boldsymbol{\theta}}(\mathbf{X}_b^*)\}$, given $\widehat{\boldsymbol{\theta}}(\mathbf{X}_b^*)$.

Thus, Extended Information Criterion (EIC) proposed by Ishiguro et al. [27] is defined as

$$\mathrm{EIC} = -2\ln L\left(\mathbf{X}; \widehat{\boldsymbol{\theta}}(\mathbf{X})\right) + 2C^*. \quad (12)$$

EIC approach selects the number of hidden units with the minimum value of (12) as Shibata [29] and Shao [30] point out that this method is asymptotically equivalent to leaving-one-out CV and AIC.

Note that the bootstrap algorithm requires refitting of the model (retraining the network) B times [31]. The number of replications B is in the range $20 \le B \le 200$, and so $B = 200$ bootstrap replications are used. The competing networks share the same architecture with the only exception being the number of hidden units.

2.1.3. Bootstrapping the Deviance. No standard procedure by which to assess the overall goodness-of-fit of the multiple-group neural discriminant model has been proposed. By introducing the maximum likelihood principle, the deviance allows us to test the overall goodness-of-fit of the model:

$$\mathrm{Dev} = 2\left[\ln L_f - \ln L\left(\mathbf{X}; \widehat{\boldsymbol{\theta}}\right)\right] = 2\left[\sum_{d=1}^{D}\sum_{k=1}^{K}\left\{t_k^{\langle d\rangle}\ln\left(\frac{t_k^{\langle d\rangle}}{\widehat{o}_k^{\langle d\rangle}}\right)\right\}\right], \quad (13)$$

where $\ln L(\mathbf{X}; \widehat{\boldsymbol{\theta}})$ denotes the maximized log likelihood under a current neural discriminant model. Since the log likelihood for the full model $\ln L_f = 2\sum_{d=1}^{D}\sum_{k=1}^{K}\{t_k^{\langle d\rangle}\ln t_k^{\langle d\rangle}\}$ is zero by using the definition $0\ln 0 = 0$, we have

$$\mathrm{Dev} = -2\sum_{d=1}^{D}\sum_{k=1}^{K}\left\{t_k^{\langle d\rangle}\ln\widehat{o}_k^{\langle d\rangle}\right\}. \quad (14)$$

Note that the deviance is two times log likelihood Equation (10). The greater the deviance, the poorer the fit of the model. However, the deviance given in (14) is not even approximately distributed as χ^2 for the case in which binary (Bernoulli) responses are available [32–35]. We therefore provide the bootstrap estimator of the percentile (i.e., the critical point) for the deviance given in (14) according to the following algorithm.

Step 1. Generate B (= 200) bootstrap samples $\mathbf{X}^*$ drawn with the replacement from the training sample $\mathbf{X}$ with the optimum number of hidden units which was determined by the way in Section 2.1.2.

Step 2. For the bootstrap sample $\mathbf{X}_b^*$, $b = 1, 2, \ldots, B$, the deviance given in (14) is computed as

$$\mathrm{Dev}(b) = -2\ln L\left(\mathbf{X}_b^*; \hat{\boldsymbol{\theta}}\left(\mathbf{X}_b^*\right)\right). \tag{15}$$

This process is independently repeated B times, and the computed values are arranged in ascending order.

Step 3. The value of the jth order statistic Dev* of the B replications can be taken as an estimator of the quantile of order $j/(B+1)$.

Step 4. The estimator of the $100(1-\alpha)$-th percentile (i.e., the 100α% critical point) of Dev* is used to test the goodness-of-fit of the model using a specified significance level $\alpha = 1 - j/(B+1)$. If the value of the deviance given in (14) is greater than the estimate of the percentile, then the model fits poorly.

2.1.4. Excess Error Estimation. Let error rate $e(F; \mathbf{X})$ be the probability of incorrectly predicting the outcome of a new observation drawn from an unknown distribution F, given a prediction rule on a training sample $\mathbf{X}$. This error rate is defined as the *actual error rate*, which is of interest in performance assessment of prediction rules. Let $\hat{F}$ be the empirical distribution function that places a mass equal to $1/D$ at each point $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_d, \ldots, X_D$. We apply a prediction rule η to this training sample $\mathbf{X}$ and form the realized prediction rule $\eta_{\hat{F}}(\mathbf{x}^{\langle 0\rangle})$ for a new observation $\mathbf{X}_0 = \{\mathbf{x}^{\langle 0\rangle}, \mathbf{t}^{\langle 0\rangle}\}$. Let $Q(\mathbf{t}^{\langle 0\rangle}, \eta_{\hat{F}}(\mathbf{x}^{\langle 0\rangle}))$ indicate the discrepancy between an observed value $\mathbf{t}^{\langle 0\rangle}$ and its predicted value $\eta_{\hat{F}}(\mathbf{x}^{\langle 0\rangle})$. Let error rate $e(\hat{F}; \mathbf{X})$, referred to as the *apparent error rate*, be the probability of incorrectly predicting the outcome for the sample drawn from the empirical distribution of the training sample, $\hat{F}$. Because the training sample is used for both forming and assessing the prediction rule, this proportion (i.e., apparent error rate) underestimates the actual error rate. The difference $e(\hat{F}; \mathbf{X}) - e(F; \mathbf{X})$ is the *excess error*. The *expected* excess error (i.e., *bias*) of a given prediction rule [22, 23, 36, 37] is

$$b(F) = E\left[e\left(\hat{F}; \mathbf{X}\right) - e(F; \mathbf{X})\right]. \tag{16}$$

When the prediction rule by multiple-group neural discriminant model is allowed to be complicated, overfitting becomes a real danger, and excess error estimation becomes important. Thus we will consider the bootstrapping to estimate the expected excess error when fitting a multiple-group neural discriminant model to the data. The algorithm can be summarized as follows.

Step 1. Generate bootstrap samples $\mathbf{X}^*$ from $\hat{F}$ as described in Section 2.1.2. Let $\hat{F}^*$ be the empirical distribution of $\mathbf{X}_1^*, \mathbf{X}_2^*, \ldots, \mathbf{X}_d^*, \ldots, \mathbf{X}_D^*$.

Step 2. For each bootstrap sample $\mathbf{X}^*$, fit a model to obtain the estimator $\hat{\boldsymbol{\theta}}(\mathbf{X}^*)$ and construct the realized prediction rule $\eta_{\hat{F}^*}$ based on $\mathbf{X}^*$.

Step 3. The bootstrap estimator of the expected excess error in (16) is given by

$$\begin{aligned} R^* = &\frac{1}{D}\sum_{d=1}^{D} Q\left(\mathbf{t}^{\langle d\rangle *}, \eta_{\hat{F}^*}\left(\mathbf{x}^{\langle d\rangle *}\right)\right) \\ &- \frac{1}{D}\sum_{d=1}^{D} Q\left(\mathbf{t}^{\langle d\rangle}, \eta_{\hat{F}^*}\left(\mathbf{x}^{\langle d\rangle}\right)\right), \end{aligned} \tag{17}$$

where

$$Q\left(\mathbf{t}^{\langle d\rangle *}, \eta_{\hat{F}^*}\left(\mathbf{x}^{\langle d\rangle *}\right)\right) = \begin{cases} 1: & \text{incorrect discriminant.} \\ 0: & \text{o.w.} \end{cases} \tag{18}$$

Step 4. Repeat Step 1–Step 3 for bootstrap samples $\mathbf{X}_b^*$, $b = 1, 2, \ldots, B(= 200)$ to get R_b^*. The bootstrap estimator of the expected excess error can be obtained as

$$b\left(\hat{F}\right) \cong \frac{1}{B}\sum_{b=1}^{B} R_b^*. \tag{19}$$

Step 5. The actual error rate with bootstrap bias correction is

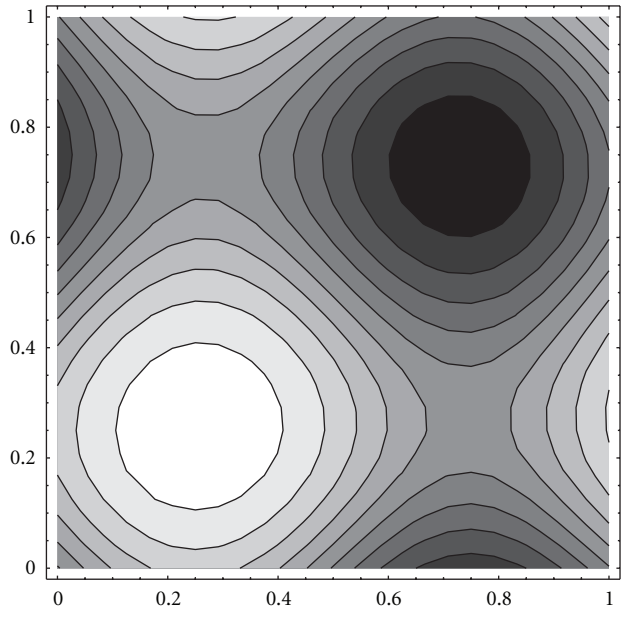
$$e_{\mathrm{boot}} = e\left(\hat{F}; \mathbf{X}\right) - b\left(\hat{F}\right). \tag{20}$$

3. Simulation Study

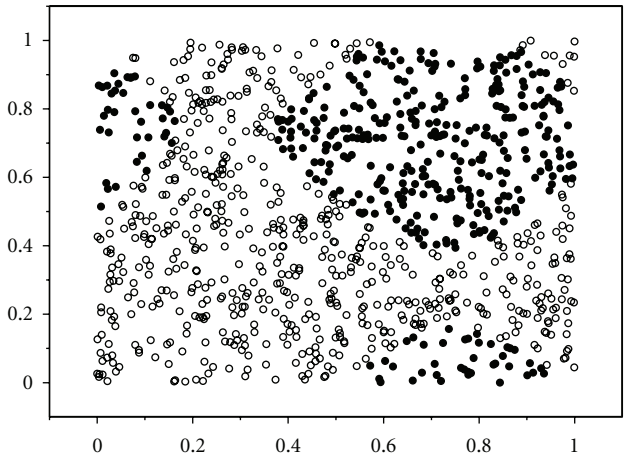
Since the model generally does not encompass unknown functions, but rather only approximations thereof, the model is inherently misspecified. Therefore, we demonstrate results from some Monte Carlo simulations to evaluate the performance. The criterion based on bootstrapping is demonstrated to be favorable when selecting the optimum number of hidden units [38–41]. Vach et al. [41] investigated how regression functions can be approximate specific regression from the class

$$f(x) = \beta_0 + \sum_{i=1}^{I}\beta_i x_i + \sum_{i=1}^{I}\gamma_i x_i^2 + \sum_{i<i'}\delta_{ii'} x_i x_{i'}, \tag{21}$$

and pointed out that the comparison using members of this is a little bit unfair. We thus show the superiority of



(a) Contour plot



(b) Class membership indicator

Figure 2: Contour plots: (a) darker grey scale levels represent lower probabilities of $y = 0$ and (b) • and ◦ show the class membership indicators $y = 0$ and $y = 1$, respectively, for the covariates (x_1, x_2).

neural network model by using the function of the existence of several local extrema. The influence can be illustrated through a simple simulation using a neural network model with two inputs x_1 and x_2, because we can visualize the contour plot of unknown population.

3.1. Two-Class Classification. The influence can be illustrated through a simple simulation using a neural network model with two inputs, one output and a varying number of hidden units. For two independent continuous covariates x_1 and x_2, we simulated the following known (true) model:

$$f(x_1, x_2) = \frac{1}{1 + \exp[-\sin(2\pi x_1) - x_1 x_2 - \sin(2\pi x_2)]}. \tag{22}$$

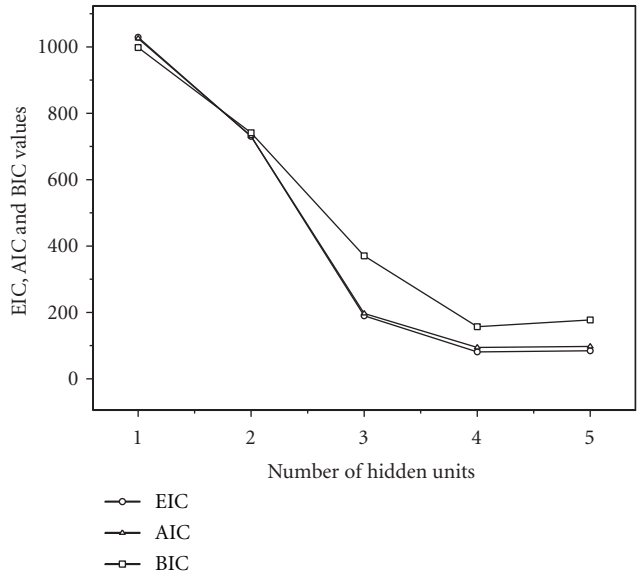


Figure 3: EIC, AIC, and BIC values for the simulation using only one training sample (note that the series of EIC and AIC are indistinguishable).

Training and test samples of size 1000 were considered in the present study. Input data (x_1, x_2) are chosen from data that are uniformly distributed over $[0, 1] \times [0, 1]$, and the binary response y is labeled with 1 if $f(x_1, x_2) > (1/2)$ and otherwise with 0. Figure 2 shows the distribution of the covariates (x_1, x_2) and the class membership indicator in the training sample.

EIC values with $B = 200$ replications based on bootstrapping pairs for the training sample are shown in Figure 3 after fitting the neural discriminant models having one to five hidden units. For the purpose of comparison, the values of AIC and BIC are also provided. In the case of the simulation study, the known (true) model given in (22) is included in the population. Thus, the differences between EIC and AIC values are slight.

Using the simulated training sample, the feed-forward neural networks were fit to the known (true) model given in (22). The tendency of mapping performed by neural networks with hidden units $h = 1, 2, 3$ to implausibly fit the function given in (22) can also be illustrated.

The bootstrap estimate of the 95th percentile Dev* (i.e., the 5% critical point) for the training sample with four hidden units is Dev* = 203.10. Comparison to the deviance given in (14) (Dev = 39.40) suggests that the multiple-group neural discriminant model fits the data fairly well because Dev = 39.40 is far from the 5% critical point Dev* = 203.10.

The actual error rate with the bootstrap bias correction given in (20) for the multiple-group neural discriminant models with four hidden units is calculated as $e_{\text{boot}} = 0.009$. Figure 4 illustrates the apparent error rates observed in the training sample- and test sample-based error rates. The apparent error rates for both samples decreased with the increase in the number of hidden units from $h = 1$ to $h = 4$ and then remained constant.

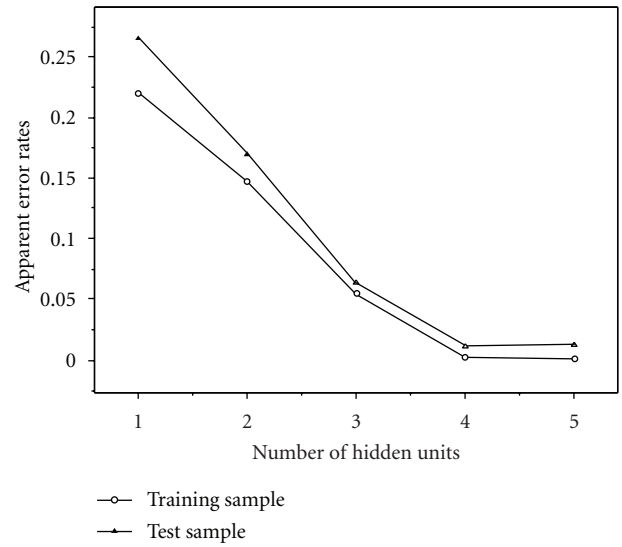


Figure 4: Apparent error rates for simulated data after fitting neural networks with one to five hidden units.

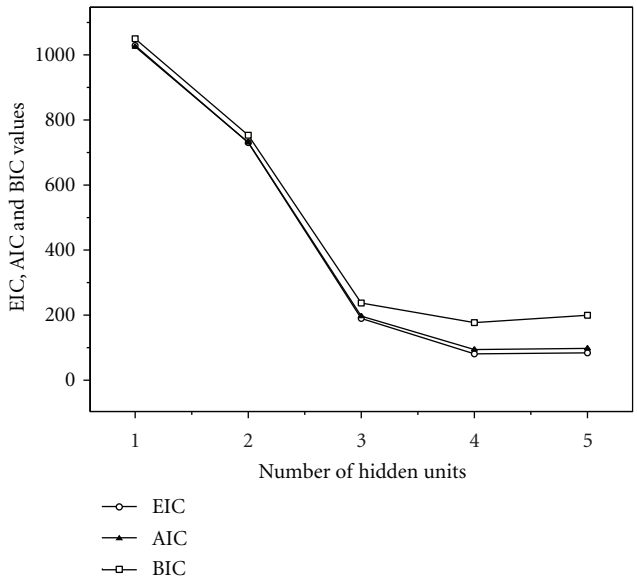


Figure 5: Average values of EIC, AIC and BIC for the simulated sample with 100 replications.

Figures 3 and 4 are based on only one simulated data set. However, the efficacy of the bootstrap procedures would be more convincingly illustrated in a simulation study based on multiple samples. Figure 5 shows the average values of EIC, AIC, and BIC based on multiple samples with 100 replications after fitting the neural discriminant models having one to five hidden units. Figure 6 shows the box-and-whisker plots for EIC in order to evaluate the standard errors and other statistics. Figure 7 illustrates the mean apparent error rates observed in multiple test samples with 100 replicates. Figure 8 also shows the box-and-whisker plots for the mean apparent error rates in multiple test samples with 100 replicates. For the purpose of comparison, the estimates

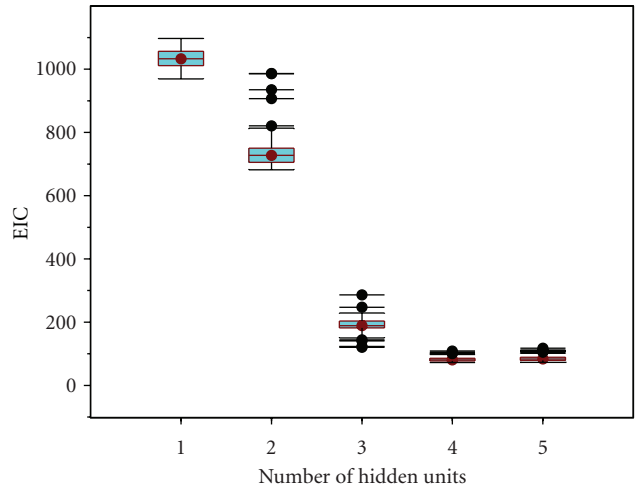


Figure 6: Box-and-whisker plots for EIC.

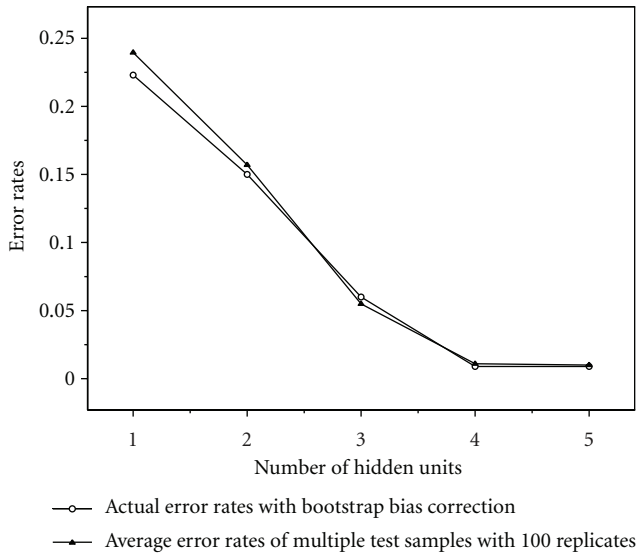


Figure 7: Error rates of multiple test samples with 100 replicates and actual error rate with bootstrap bias.

of the actual error rates with bootstrap bias correction for the training sample [42] are also shown in Figure 7. It is concluded that EIC identifies the optimal number of hidden units (i.e., 4) more often than AIC. In addition, the differences between the average values of EIC and AIC are somewhat similar to Figure 3, and the average values of the bootstrap-corrected estimate of the prediction error rate vary around the average apparent error rates for the multiple test samples.

3.2. Multiclass Classification. Input data (x_1, x_2) are generated from uniformly distributed over $\mathbf{x} = (x_1, x_2) \in [-1, 1] \times [-1, 1]$. By substituting $\mathbf{x} = (x_1, x_2)$ into

$$\Pr_k = \frac{\exp\{f_k(x_1, x_2)\}}{1 + \sum_{k=1}^{K-1} \exp\{f_k(x_1, x_2)\}}, \quad k = 1, 2, \ldots, K-1 \tag{23}$$

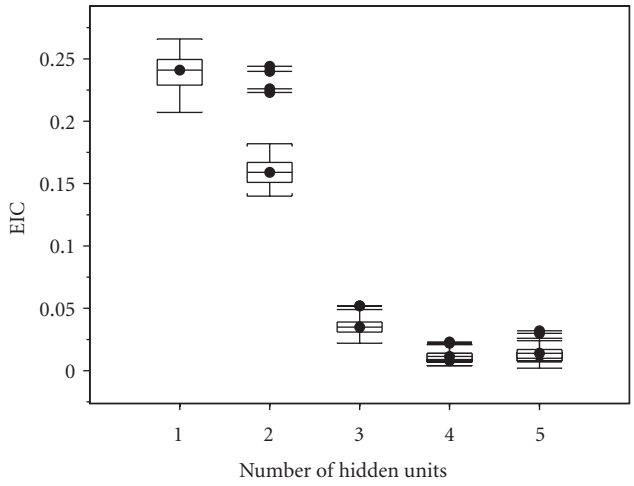


Figure 8: Box-and-whisker plots for the mean apparent error rates in multiple test samples with 100 replicates.

$\mathbf{x}$ can be grouped into four classes in the case of $K = 4$. As nonlinear function $f_k(x_1, x_2)$,

$$f_1(x_1, x_2) = 5\sin(2\pi x_1) - 3\sin(2\pi x_2),$$
$$f_2(x_1, x_2) = \sin(2\pi x_1) + 2\cos(2\pi x_2),$$
$$f_3(x_1, x_2) = 2 - 8(x_1 - 0.5)(x_1 - 0.5) - 8(x_2 - 0.5)(x_2 - 0.5) \quad (24)$$

can be used with $\sum_{k=1}^{K} \Pr_k = 1$ [41, 43].

By use of definition of t_k in (9), the observations can be divided into four class by multinomial random number

$$\mathbf{t} \sim \text{Multinorm}(1, \mathbf{Pr}) \begin{cases} \text{Class } 1 : \mathbf{t} = (1, 0, 0, 0) \\ \text{Class } 2 : \mathbf{t} = (0, 1, 0, 0) \\ \text{Class } 3 : \mathbf{t} = (0, 0, 1, 0) \\ \text{Class } 4 : \mathbf{t} = (0, 0, 0, 1) \end{cases} \quad (25)$$

with $\mathbf{t} = (t_1, t_2, t_3, t_4)$, $\mathbf{Pr} = (\Pr_1, \Pr_2, \Pr_3, \Pr_4)$.

In this paper, training and test samples of size 400 were considered. The apparent error rates for training and test samples of several models are given in Table 1. From Table 1, it is found that the apparent error rates of training and test sample for multiple-group neural discriminant model is the smallest.

4. Results and Discussion

Prediction accuracy (error rate) is the most important consideration in the development of prediction model. The assessment of goodness-of-fit is a useful exercise. In particular the goodness-of-fit and error rate from the training data are meaningful because of overfitting issue. The main purpose is to predict the future samples accurately. In other words, in real applications, the test sample population may be different from the training samples. A benchmark data set is thus used to illustrate the advantages of the models and methods developed herein. A multiple-group neural discriminant model having a single hidden layer was applied to a data set of 3,772 training instances and 3,428 testing instances of a thyroid disease database. All of these data sets are available on the World Wide Web at http://archive.ics.uci.edu/ml/datasets/Thyroid+Disease. The present study considered three groups: hypothyroid, hyperthyroid, and normal. The laboratory profiles on which the differential diagnosis is made consist of 21 attributes (15 attributes are binary, and six attributes are continuous):

x_1: *age*, x_2: *sex*, x_3: *on thyroxine*, x_4: *query thyroxine*, x_5: *on antithyroid*, x_6: *sick*, x_7: *pregnant*, x_8: *thyroid surgery*, x_9: *I131 treatment*, x_{10}: *query hypothyroid*, x_{11}: *query hyperthyroid*, x_{12}: *lithium*, x_{13}: *goitre*, x_{14}: *tumour*, x_{15}: *hypopituitary*, x_{16}: *psych*, x_{17}: *TSH*, x_{18}: *T3*, x_{19}: *TT4*, x_{20}: *T4U*, x_{21}: *FTI*.

Table 2 is a list of the first-five observations for the 21 attributes and the group with respect to the training sample. The training sample is used to determine the neural network model structure. Table 3 is a list of the first-five observations for 21 attributes and the group with respect to the test sample. The goal of discrimination is to assign new observations to one of the mutually exclusive groups. The data in Tables 2 and 3 include six continuous and 15 binary attributes. Fisher's discriminant model assumed that the inputs are normal distributed. However, it is worth noting that the posterior class probabilities for neural discriminant model can be given by maximizing log likelihood Equation (10) without the normal distributed assumption for inputs.

A thyroid disease database has been used as a benchmark test for the neural network model shown in Figure 1 with $I = 21$ and $K = 3$. EIC values are shown in Figure 9 after fitting the multiple-group neural discriminant models having one to four hidden units. In this case, the true model is not included in the population. For the purpose of comparison, AIC and BIC values are also provided.

Figure 9 indicates that the minimum EIC value is obtained for the model having two hidden units, which has an apparent error rate $e(\hat{F}; X)$ of 0.0090. Figure 10 shows a histogram of the bootstrap replications R_b^* that are used to estimate the expected excess error. The values of the mean and standard deviation of R_b^* are -0.0033 and 0.0022, respectively. The actual error rate with the bootstrap bias correction given in (20) for the multiple-group neural discriminant models with two hidden units is calculated as $e_{\text{boot}} = 0.012$.

Table 1: Comparison of various discriminant methods for simulated data.

Model	Apparent error rate	
	Training sample	Test sample
Multiple-group neural discriminant model ($h = 5$)	0.290	0.308
Proportional odds model	0.550	0.543
Linear discriminant model	0.545	0.525
Quadratic discriminant model	0.518	0.500
Tree-based model	0.290	0.345

Table 2: List of the first-five observations for 21 attributes and the group with respect to the training sample.

0.73	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0.00060	0.015	0.120	0.082	0.146	3
0.24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00025	0.030	0.143	0.133	0.108	3
0.47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00190	0.024	0.102	0.131	0.078	3
0.64	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00090	0.017	0.077	0.090	0.085	3
0.23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00025	0.026	0.139	0.090	0.153	3

Table 3: List of the first-five observations for 21 attributes and the group with respect to the test sample.

0.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00600	0.028	0.111	0.131	0.085	2
0.32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00100	0.019	0.084	0.078	0.107	3
0.35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00000	0.031	0.239	0.100	0.239	3
0.21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00100	0.018	0.087	0.088	0.099	3
0.22	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0.00000	0.022	0.134	0.135	0.099	3

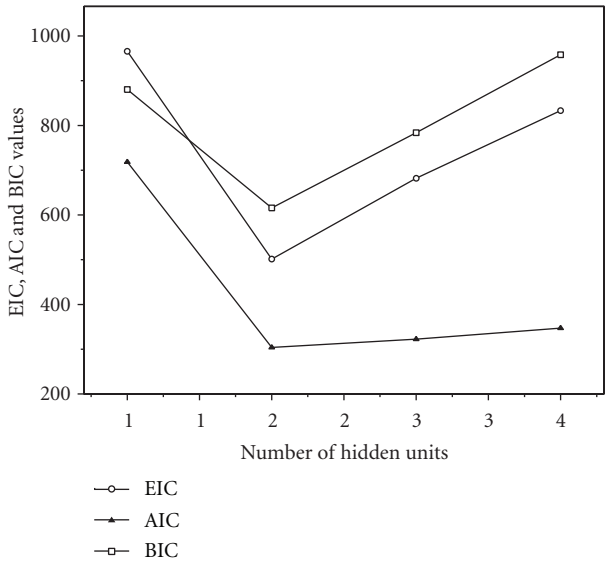


Figure 9: EIC, AIC and BIC values for the training sample of a thyroid disease database.

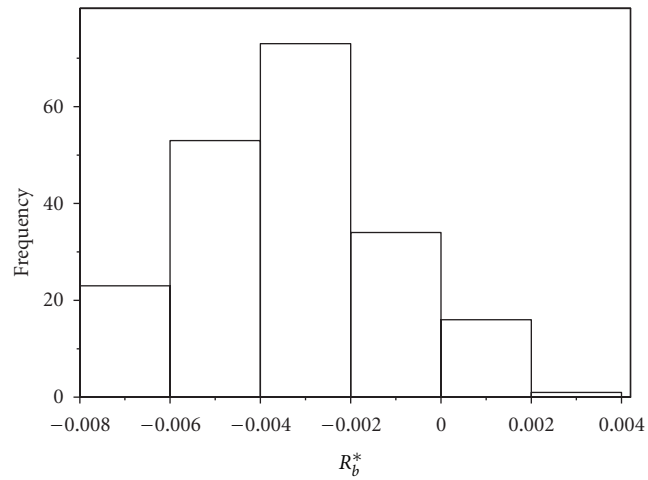


Figure 10: Histogram of bootstrapped replications R_b^*.

The histogram of the bootstrapped Dev(b) for $B = 200$ is provided in Figure 11. The bootstrap estimate of the 95th percentile Dev* (i.e., the 5% critical point) for the thyroid disease training sample with two hidden units is Dev* = 344.41. Comparison to the deviance given in (14) (Dev = 204.03) suggests that the multiple-group neural discriminant model fits the data fairly well. For reference, the Q-Q plot of the bootstrapped Dev(b) for $B = 200$ is shown in Figure 12.

Alternatively, if the deviance Equation (14) asymptotically follows the χ^2 distribution with $D - p = 3772$ degrees of freedom under the null hypothesis that the model is correct, the probability density function of the χ^2 distribution with 3772 degrees of freedom is shown in Figure 13. However, because of large sample size $D = 3772$, the distribution is extremely skewed. By comparing Figure 13 with Figure 11, it is found that the distribution of deviance Equation (14) can not be approximated by χ^2 distribution. Furthermore, the mean and deviance of bootstrapped Dev(b) are E[Dev(b)] = 205.55 and Var[Dev(b)] = 4436.90, respectively, which are not close to those of the χ^2 distribution with 3772 d.f., that is, $E[\chi^2]$ = d.f. = 3772 and Var$[\chi^2] = 2 \times$ d.f. = 7544. It should be noted that the deviance asymptotically follows χ^2 distribution for grouped binary (i.e., binomial) response and a set of predictor variables, as described in Tsujitani and Aoki [44].

The apparent error rates after fitting the multiple-group neural discriminant models having one to four hidden units are shown in Figure 14. Figure 14 indicates that (i) the multilayer feedforward neural network can approximate virtually any function up to some desired level of approximation with the number of hidden units increased *ad libitum* for the training sample, (ii) the actual error rate for the test sample is the smallest when the number of hidden units is two, and (iii) a neural network with a large number of hidden units has a higher error rate for the test sample, because the noise is modeled in addition to the underlying function.

Although the model fits the training sample as well as possible by increasing the number of hidden units, the model does not generalize very well to the test sample, which is the goal. The apparent error rates for training and test samples of several models are given in Table 4: (i) the multigroup logistic discriminant model with linear effect [6, 45] by use

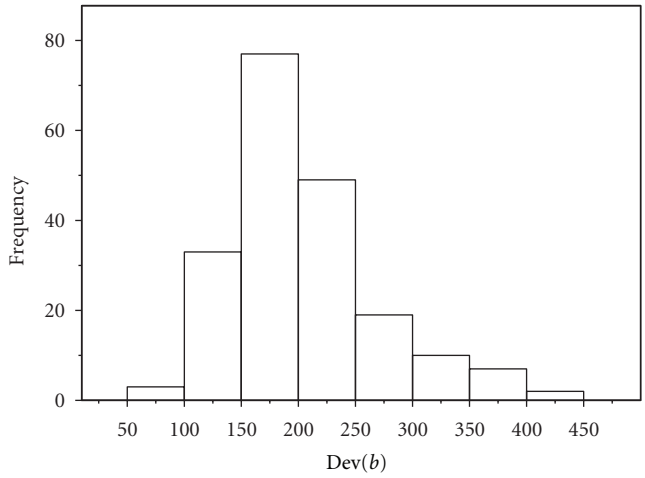


Figure 11: Histogram of bootstrapped deviance Dev(b) for $B = 200$.

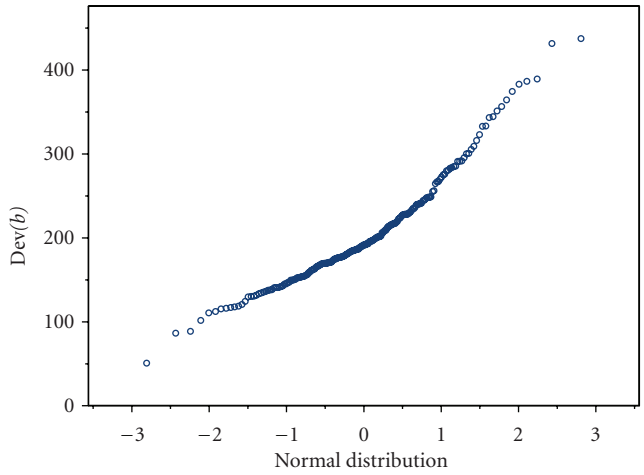


Figure 12: Q-Q plot of bootstrapped deviance Dev(b) for $B = 200$.

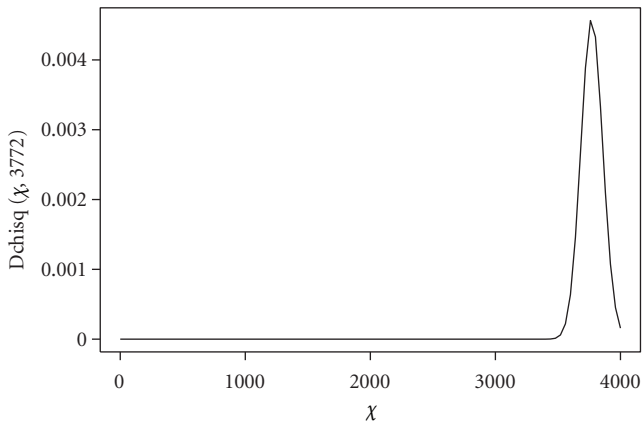


Figure 13: Probability density function of χ^2 distribution on 3772 d.f.

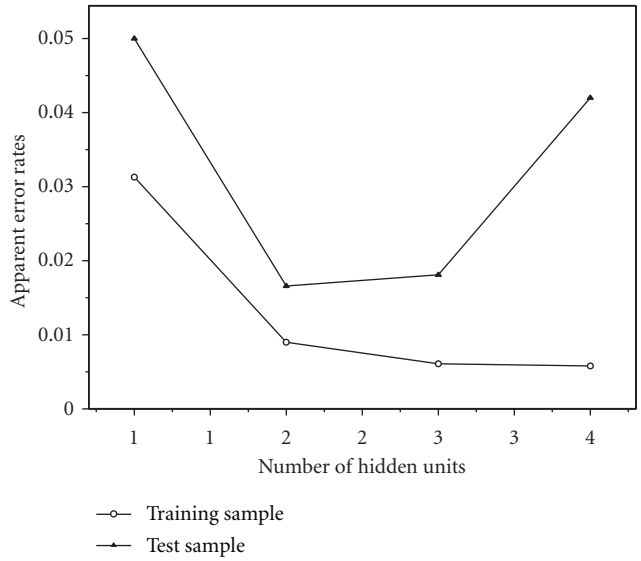


Figure 14: Apparent error rates for the thyroid disease database after fitting neural networks with one to hidden units.

Table 4: Comparison of various discriminant methods for a thyroid disease database.

Model	Apparent error rate	
	Training sample	Test sample
Multiple-group neural discriminant model ($h = 2$)	0.009	0.017
Multiple-group logistic discrimination with linear effects of covariates	0.028	0.041
Multiple-group logistic discrimination with linear + quadratic effects of covariates	0.019	0.032
Tree-based model	0.002	0.073
2-nearest neighbor	0.017	0.074
3-nearest neighbor	0.038	0.089
Kernel smoother	0.054	0.062
SVM	0.045	0.052
Proportional odds model	0.048	0.055
VGAM	0.000	0.021

of library{VGAM} in free software R [15], (ii) multiple-group logistic discrimination models with linear + quadratic effects, (iii) the tree-based model with mincut = 5, minsize = 10, mindev = 0.01 as tuning parameters [46] by use of library{rpart} in R, (iv) the nearest neighbor smoother using a nonparametric method to derive the classification criterion [6, 47] by use of library (knn) in R, (v) the kernel smoother [47] using normal distribution and a radius $r = 1.1$ to specify a kernel density by use of library{ks} in R, (vi) the support vector machine using the "one-against-one" approach [48, 49] by use of library{e1071} in R, (vii) the proportional odds model [14], and (viii) VGAM based on the proportional odds model with optimum smoothing parameters selected by leaving-one-out cross-validation [20] by use of library{VGAM} in R.

Table 5: Apartment house data for assessment of land value by the metropolitan area stations.

Name of metropolitan area stations	Average price of house built for sale	Average house rent	Yield	Assessment of station value by the number of passengers getting on and off	Group
Shinjyuku	5850	25.3	5.2	88	I
Kita-kashiwa	2810	9.6	4.1	80	IV
Minami-kashiwa	2890	9.9	4.1	80	III
Asakusa-bashi	4310	19.3	5.4	84	II
Yokohama	3740	18.4	5.9	100	I

From Table 4, it is found that multiple-group neural discriminant model ($h = 2$) has the smallest error rate for test sample preserving relatively small error rate for training sample. In order to overcome the stringent assumption of the additive and purely linear effects of the covariates, multiple-group logistic discrimination models with linear and quadratic effects were included. The improvement obtained by the inclusion of the quadratic effect is slight. It should be noted that the apparent error rates for training of VGAM are the smallest, but that for test samples are large. This overfitting leads poor generalization. For example, the estimated smooth function of the covariate "age" for VGAM in Figure 15 shows the overfitting.

Table 5 is apartment house data for assessment of land value by the metropolitan area stations, of the metropolitan area stations with four-class classification [50]. By using the four covariates (average price of house built for sale, average house rent, yield, assessment of station value by the number of passengers getting on and off), and assessment of land value by the metropolitan area stations may be grouped into four categories:

(i) the most comfortable,

(ii) very comfortable,

(iii) s little comfortable,

(iv) not comfortable.

Figure 16 indicates the values of EIC, AIC, and leaving-one-out CV (See the Appendix). The leaving-one-out CV is also included in order to assess the bootstrapping. The minimum EIC and leaving-one-out CV values are obtained for the model having two hidden units. However, the number of hidden unit with the minimum AIC value is three. The actual error rates in the case using EIC and leaving-one-out CV with two hidden units are 0.276 and 0.273, respectively. The bootstrapping is assessed from the point of leaving-one-out CV. The apparent error rates for training samples of several models are given in Table 6. From Table 6, it is found that multiple-group neural discriminant model ($h = 2$) has the smallest error rate.

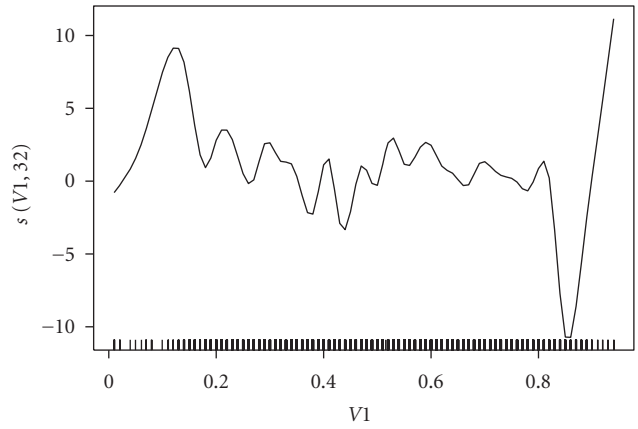


Figure 15: the estimated smooth function of the covariate "age" for VGAM.

5. Conclusions

We discussed the learning algorithm by maximizing the log likelihood function. Statistical inference based on the likelihood approach for the multiple-group neural discriminant model was discussed, and a method for estimating bias on the expected log likelihood in order to determine the optimum number of hidden units was suggested. The key idea behind bootstrapping is to focus on the optimum tradeoff between the unbiased approximation of the underlying model and the loss in accuracy caused by increasing the number of hidden units. In the context of applying bootstrap methods to a multiple-group neural discriminant model, this paper considered three methods and performed experiments using two data sets to evaluate the methods. The three methods are bootstrap pairs sampling algorithm, goodness-of-fit statistical test, and excess error estimation algorithm.

There are two broad limitations to our approach. First, the use of batch backpropagation algorithm including momentum prevents an maximum likelihood estimates from getting trapped in a local minimum, not global minimum. So far, our discussion of neural networks has focused on the maximum likelihood to determine the network parameters (weights and biases). However, a Bayesian neural network approach [51] might provide a more formal framework in which to incorporate a prior parameter distribution. Second, our neural network models assumed the independence of the predictor variables $\mathbf{x} = (x_1, \ldots, x_I)$. More generally, it may be preferable to visualize interactions between predictor variables. The smoothing spline ANOVA models can provide an excellent means for data of mutually exclusive groups and a set of predictor variables [43, 52]. We expect that flexible

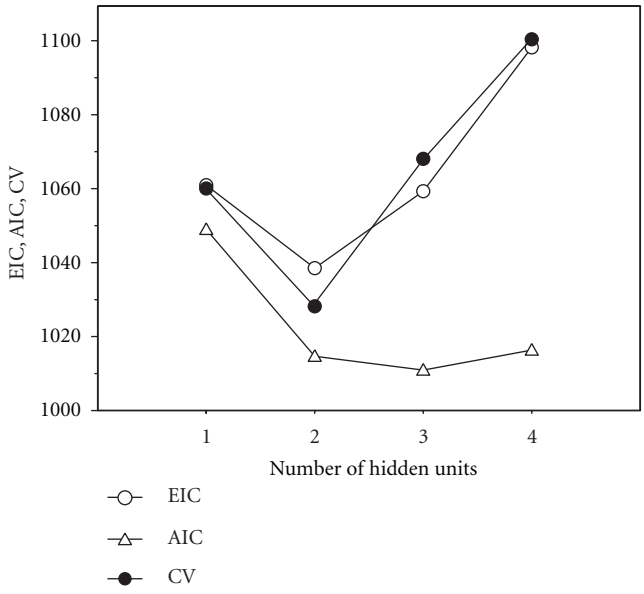


Figure 16: EIC, AIC and (leaving-one-out) CV values for the training sample of apartment house data.

Table 6: Comparison of various discriminant methods for apartment house data.

Model	Apparent error rate
Multiple-group neural discriminant model ($h = 2$)	0.262
Multigroup logistic discrimination	0.268
Proportional odds model	0.294
2-nearest neighbor	0.348

methods for discriminant model using machine learning theory [47, 53–55] such as penalized smoothing splines and support vector machine [17–19] will be very useful in these real-world contexts.

Appendix

Leaving-One-Out CV

An alternative model selection strategy for the bias correction Equation (14) of the log likelihood is leaving-one-out CV for a multiple-group neural discriminant model, which is asymptotically equivalent to TIC [29]. Let the training sample $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_d, \ldots, \mathbf{X}_D\}$ be independently distributed in an unknown distribution. We then obtain the leaving-one-out CV algorithm.

Step 1. Generate the training samples $\mathbf{X}_{[d]} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{d-1}, \mathbf{X}_{d+1}, \ldots, \mathbf{X}_D\}$, $d = 1, 2, \ldots, D$. The subscript $[d]$ of a quantity indicates the deletion of the dth data point X_d from the training sample $\mathbf{X}$.

Step 2. Using each training sample, fit a model. Then, estimate unknown parameters denoted by $\theta(X_{[d]})$ and predict the output $o_{k[d]}^{\langle d \rangle}$ for the deleted sample point $X_{[d]}$.

Step 3. The average predictive log likelihood of the deleted sample point is

$$\frac{1}{D} \ln \left[\prod_{d=1}^{D} \prod_{k=1}^{K} \left\{ o_{k[d]}^{\langle d \rangle} \right\}^{t_k^{\langle d \rangle}} \right]. \tag{A.1}$$

As a matter of convention, the cross-validation criterion is often stated as that of minimizing

$$\mathrm{CV} = -2 \ln \left[\prod_{d=1}^{D} \prod_{k=1}^{K} \left\{ o_{k[d]}^{\langle d \rangle} \right\}^{t_k^{\langle d \rangle}} \right]. \tag{A.2}$$

The leaving-one-out CV criterion finds an appropriate degree of complexity by comparing the predictive probability density $\prod_{k=1}^{K} \{o_{k[i]}^{\langle i \rangle}\}^{t_k^{\langle i \rangle}}$ for different model specifications. Anders and Korn [24] have shown that the CV criterion does not rely on any probabilistic assumption based on the properties of maximum likelihood estimators for misspecified models and is not affected by identification problems.

References

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

[2] J. S. Bridle, "Probabilistic interpretation of feed-forward classification network outputs, with relationships to statistical pattern recognition," in *Neurocomputing: Algorithms, Architectures and Applications*, F. F. Soulie and J. Herault, Eds., pp. 227–236, Springer, New York, NY, USA, 1990.

[3] B. Cheng and D. M. Titterington, "Neural networks: a review from statistical perspective," *Statistical Science*, vol. 9, pp. 2–54, 1994.

[4] H. Gish, "Maximum likelihood training of neural networks," in *Artificial Intelligence Frontiers in Statistics*, D. J. Hand, Ed., pp. 241–255, Chapman & Hall, New York, NY, USA, 1993.

[5] M. D. Richard and R. P. Lippmann, "Neural network classifiers estimate Bayesian a posteriori probabilities," *Neural Computation*, vol. 3, pp. 461–483, 1991.

[6] B. D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, New York, NY, USA, 1996.

[7] H. White, "Some asymptotic results for learning in single hidden-layer feedforward network models," *Journal of the American Statistical Association*, vol. 84, pp. 1003–1013, 1989.

[8] M. Aitkin and R. Foxall, "Statistical modelling of artificial neural networks using the multi-layer perceptron," *Statistics and Computing*, vol. 13, no. 3, pp. 227–239, 2003.

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

[10] 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.

[11] G. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[12] J. Shao, "Linear model selection by cross-validation," *Journal of the American Statistical Association*, vol. 88, pp. 486–494, 1993.

[13] P. Zhang, "Model selection via multifold cross validation," *Annals of Statistics*, vol. 21, pp. 299–313, 1993.
[14] T. J. Hastie and R. J. Tibshirani, *Generalized Additive Models*, Chapman & Hall, New York, NY, USA, 1990.
[15] S. N. Wood, *Generalized Additive Models: An Introduction with R*, Chapman & Hall, New York, NY, USA, 2006.
[16] N. Cristianini and J. Shawe-Tylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Method*, Cambridge University Press, Cambridge, UK, 2000.
[17] Y. J. Lee and S. Y. Huang, "Reduced support vector machines: a statistical theory," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 1–13, 2007.
[18] E. Romero and D. Toppo, "Comparing support vector machines and feedforward neural networks with similar hidden-layer weights," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 959–963, 2007.
[19] Q. Tao, D. Chu, and J. Wang, "Recursive support vector machines for dimensionality reduction," *IEEE Transactions on Neural Networks*, vol. 19, no. 1, pp. 189–193, 2008.
[20] T. W. Yee and C. J. Wild, "Vector generalized additive models," *Journal of the Royal Statistical Society Series B*, vol. 58, pp. 481–493, 1996.
[21] K. I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol. 2, no. 3, pp. 183–192, 1989.
[22] 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.
[23] M. C. Wang, "Re-sampling procedures for reducing bias of error rate estimation in multinomial classification," *Computational Statistics and Data Analysis*, vol. 4, no. 1, pp. 15–39, 1986.
[24] U. Anders and O. Korn, "Model selection in neural networks," *Neural Networks*, vol. 12, no. 2, pp. 309–323, 1999.
[25] M. Ishiguro and Y. Sakamoto, "WIC: an estimation-free information criterion," Research Memorandum of the Institute of Statistical Mathematics, Tokyo, Japan, 1991.
[26] S. Konishi and G. Kitagawa, "Generalised information criteria in model selection," *Biometrika*, vol. 83, no. 4, pp. 875–890, 1996.
[27] 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, 1997.
[28] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.
[29] R. Shibata, "Bootstrap estimate of Kullback-Leibler information for model selection," *Statistica Sinica*, vol. 7, no. 2, pp. 375–394, 1997.
[30] J. Shao, "Bootstrap Model Selection," *Journal of the American Statistical Association*, vol. 91, no. 434, pp. 655–665, 1996.
[31] R. Tibshirani, "A comparison of some error estimates for neural network models," *Neural Computation*, vol. 8, no. 1, pp. 152–163, 1996.
[32] D. Collett, *Modeling Binary Data*, Chapman & Hall, New York, NY, USA, 2nd edition, 2003.
[33] D. E. Jennings, "Outliers and residual distributions in logistic regression," *Journal of the American Statistical Association*, vol. 81, pp. 987–990, 1986.
[34] 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.
[35] D. Pregibon, "Logistic regression diagnostics," *Annals of Statistics*, vol. 9, pp. 705–724, 1981.
[36] B. Efron, "Estimating the error rate of a prediction rule: improvement on cross-validation," *Journal of the American Statistical Association*, vol. 78, pp. 316–331, 1983.
[37] B. Efron, "How biases is the apparent error rate of a prediction rule?" *Journal of the American Statistical Association*, vol. 81, pp. 461–470, 1986.
[38] S. Eguchi and J. Copas, "A class of logistic-type discriminant functions," *Biometrika*, vol. 89, no. 1, pp. 1–22, 2002.
[39] O. Intrator and N. Intrator, "Interpreting neural-network results: a simulation study," *Computational Statistics and Data Analysis*, vol. 37, no. 3, pp. 373–393, 2001.
[40] G. Schwarzer, W. Vach, and M. Schumacher, "On the misuses of artificial neural networks for prognostic and diagnostic classification in oncology," *Statistics in Medicine*, vol. 19, no. 4, pp. 541–561, 2000.
[41] W. Vach, R. Roßner, and M. Schumacher, "Neural networks and logistic regression: Part II," *Computational Statistics and Data Analysis*, vol. 21, no. 6, pp. 683–701, 1996.
[42] M. Tsujitani and T. Koshimizu, "Neural discriminant analysis," *IEEE Transactions on Neural Networks*, vol. 11, no. 6, pp. 1394–1401, 2000.
[43] X. Lin, *Smoothing spline analysis of variance for polychotomous response data*, Ph.D. thesis, University of Wisconsin, Madison, Wis, USA, 1998.
[44] M. Tsujitani and M. Aoki, "Neural regression model, resampling and diagnosis," *Systems and Computers in Japan*, vol. 37, no. 6, pp. 13–20, 2006.
[45] E. Lesaffre and A. Albert, "Multiple-group logistic regression diagnosis," *Journal of Applied Statistics*, vol. 38, pp. 425–440, 1989.
[46] J. M. Chambers and T. J. Hasti, *Statistical Models in S*, Chapman & Hall, New York, NY, USA, 1992.
[47] T. J. Hastie, R. J. Tibshirani, and J. Friedman, *The Elements of Statistical Learning-Data Mining, Inference and Prediction*, Springer, New York, NY, USA, 2001.
[48] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, NY, USA, 1998.
[49] T. S. Lim, W. Y. Loh, and Y. S. Shih, "Comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms," *Machine Learning*, vol. 40, no. 3, pp. 203–228, 2000.
[50] Y. Sakurai and Y. Yashiki, "Assessment of land value by the metropolitan area stations," *Weekly Takarajima*, no. 572, pp. 24–42, 2002.
[51] R. M. Neal, *Bayesian Learning for Neural Networks*, Springer, New York, NY, USA, 1996.
[52] C. Gu, *Smoothing Spline ANOVA Models*, Springer, New York, NY, USA, 2002.
[53] B. Baesens, T. Van Gestel, M. Stepanova, D. Van Den Poel, and J. Vanthienen, "Neural network survival analysis for personal loan data," *Journal of the Operational Research Society*, vol. 56, no. 9, pp. 1089–1098, 2005.
[54] D. R. Mani, J. Drew, A. Betz, and P. Datta, "Statistics and data mining techniques for life-time value modeling," in *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 94–103, San Diego, Calif, USA, 1999.
[55] W. N. Street, "A neural network model for prognostic prediction," in *Proceedings of the 15th International Conference on Machine Learning*, pp. 540–546, Wisconsin, Wis, USA, 1998.