8-16-1996

Estimating Models with Binary Dependent Variables: A Neural Network Approach

Adam S. Huarng
Purdue University

William Pracht
University of Memphis

Ravi Nath
University of Memphis

Estimating Models with Binary Dependent Variables: A Neural Network Approach

Adam S. Huarng, Indiana - Purdue University, Fort Wayne, IN 46805
William Pracht and Ravi Nath, University of Memphis, Memphis, TN 38152

Introduction

Recently there has been a considerable interest in artificial neural network models that learn to produce "outputs" from repeated experiences with a set of input and output pairs [8] [9] [26] [27] [41]. These network models achieve computational power via dense interconnection of simple processing elements. Neural net models have the greatest potential in applications requiring the simultaneous parallel operation of many processing elements. One such area of application is the classification problem in discriminant analysis. It involves the development of a decision rule to assign entities (individuals or observations) having q (q>=1) characteristics to one of the two or more given groups. The classification criterion is derived using the historical sample data for which the group memberships are already known and the resulting criterion is then used to place unclassified entities into one of the many prespecified groups.

The classification analysis has widespread applications in business. Some significant areas where this technique has been proven useful are: pattern recognition [15][22], processing loan or credit card applications [7][34], evaluating candidates for a job [42], likelihood of the success or failure of a product [13], and business failure prediction [3][40]. There exist a myriad of techniques which have been proposed to solve the classification problem. The well known solution to this problem is by Fisher [9]. It appears in almost all "canned" statistical software packages, and is also the most frequently used by researchers. It is referred to as the Fisher's linear discriminant analysis (FLDA). FLDA is known to be optimal in terms of minimizing the overall misclassification (error) rate provided the following two conditions are met: (1) the theoretical distribution of the q characteristics of the entities for each group is multivariate normal with known parameters, and (2) the variance-covariance matrices of the multivariate normal distributions in (1) are the same for all groups, i.e., all groups have the same variance-covariance structure. In practice, however, these conditions are seldom met and/or are difficult to verify. To overcome these inherent limitations of the FLDA, several alternative approaches requiring less stringent underlying assumptions about the population distributions have been proposed.

Predominant among these alternatives are the quadratic discriminant analysis (QDA) [39], linear programming (LP) approaches [2][10][11][12], and logit discriminant (LDA) analysis. Each of these techniques has been shown to perform well under a specific set of conditions. For example, for the normal distributions the QDA is suited when the assumption of group variance-covariance homogeneity is violated [5][23][32], and the LDA has higher predictive ability in case of nonnormality of the data [38]. Further the recent evidence suggests that certain linear programming-based techniques provide good alternatives under some specialized environments[2][12][20].

In this paper, a neural network-based approach to solve the classification problem is introduced and compared to some traditional procedures via Monte Carlo simulation studies. This approach unlike many other techniques does not require any statistical assumptions regarding the data.

Neural Networks

Artificial neural network systems are once again receiving a surge of interest in the artificial intelligence community. After nearly two decades of obscurity, these systems, also called connectionist systems, neurocomputers or neural networks are again in the information processing technology limelight. Work on the first artificial neural networks began more than twenty years ago [16][20][31][27][44]. These systems consisted of a single layer of artificial neurons and were enthusiastically applied to such diverse problems as weather prediction and electrocardiogram analysis. Early optimism was dashed, however, when Minsky

and Papert published proof [31] that the single-layer networks were theoretically incapable of solving any simple problems. Minsky and Papert's book, *Perceptron*, discouraged all but a few neural network researchers [1][14][18][24][28]. The work of these persistent researches has provided a much more sound theoretical foundation upon which the powerful multilayer networks of today are built.

Neural network systems are essentially a type of information processing technology which has been inspired by studies of the brain and nervous system. The basic processing unit in neural networks is the artificial neuron or processing element (PE), which is designed to mimic the first-order characteristics of the biological neuron. Figure 1 shows the workings of a processing element diagrammatically. The PE has a set of $k + 1$ inputs, $x0, x1,...,xk$. These inputs are modulated by corresponding connection weights, $wi0, wi1,...,wik$ (analogous to neuron connection strengths) to change the stimulation (activation) level internal to the neuron. The internal activation level a is determined by

$$ai = \sum_{j=0}^{k} wijxj$$

(a weighted sum of the inputs). Based on this internal stimulation (activation) level the neuron produces an output, yi which is sent to other PEs (or output of the system). The value of this output is related to the internal activation level, ai by a transfer function, call it f(ai). Several variations of the transfer function are available:

1) Linear. In this case the output is set equal to the activation level of the PE, i.e., $yi=ai$.

2) Linear threshold. Here the output is set to 1 if the activation level exceeds 0 and is set to 0 otherwise, i.e.,

$$yi = \begin{cases} 1 \text{ if } ai > 0 \\ 0 \text{ if } ai \leq 0 \end{cases}$$

3) Continuous Sigmoid. In this very common variant, the output is nonlinearly related to the activation level of the PE according to the logistic function:

$$yi = (1 + e^{-a})^{-1}$$

A neural network consists of many PE's joined together into a network. the simplest network is a group of PE's (m+1 of them) arranged in one layer, as shown in figure 2. Each processing element in the layer receives inputs, $x0,x1,...,xk$ from some external source and produces an output array, ($y0,y1,...,ym$). The early neural networks were this simple. Each PE simply outputs a weighted sum of the inputs to the network. The limited processing capability of these simple networks lead to the development of more complex networks. Processing elements are usually organized into groups called layers. A typical network consists of a sequence of layers with full (or random) connections between successive layers. Figure 3 shows a multilayer network consisting of two layers with connections to the outside world; the lower connections allows input to be presented to the system while the upper connections reveal the output.

Neural Network for the Classification Problem

This section formally presents the setup necessary to solve the classification problem using neural network models. Let G1 and G2 demote the two groups for which the classification is to be performed (the setup for more than 2 group is very similar). Suppose n column vector $Xi = (Xi1, Xi2, \ldots, Xiq)^T$, ($i$ = 1,2,....,n) denote sample data on q attributes of the n entities (the superscript "T" denotes the transpose of a matrix.) Also, assume that the first $n1$ vectors represent data from group G1 and the last $n2 = n-n1$ vectors represent

data from group G*2*. Furthermore, let *p1* and *p2*, respectively, (*p1* + *p2* = 1) demote prior probabilities for groups G*1* and G2.

The topology of the neural network model consists of *q* inputs (one for each discriminant variable) and one output (one less than the number of groups). The target output t*i* for a given input Xi is defined as:

$$t_i = \begin{cases} 1 & \text{if } Xi \text{ belongs to group G 1} \\ 0 & \text{if } Xi \text{ belongs to group G 2} \end{cases}$$

Experimental Study

In order to assess the relative effectiveness of the five classification techniques, Monte Carlo simulation procedures were used. The five techniques we consider are: Fisher's linear discriminant analysis(FLDA), quadratic discriminant analysis(QDA), logit discriminant analysis (LDA), an LP-based procedure (OSD), and the neural network (NN) model. Because this was intended to be a preliminary study and to keep it to a manageable level, only the case of q=4 discriminating variables in the two-group set up was considered. The estimation sample was set at 50 and the holdout sample was fixed at 200. For each procedure, the percent of correctly classified cases in the holdout sample served as the measure of its effectiveness. three variations of the neural network model were initially tried for a limited number of simulation runs. Each model consisted of a single hidden layer but with a different number of nodes in this layer. Models with two, three and four nodes in the hidden layer were tried. The results showed that there was no significant difference in the hit rates obtained from the three neural network setups. Therefore, to keep the model parsimonious, the neural network model consisting of four input nodes (one for each discriminating variable), one hidden layer with 2 nodes, and 1 output node (one less than the number of groups) was adopted.

Data for group 1 were generated such that they came from a distribution **f** with mean *1* = 0 and identity dispersion matrix *1* = **I**. Data for group 2 had the same distribution as that for group 1 but the mean of the distribution was *2* and the dispersion matrix was *2*. The mean of group 2 differed from the mean of group 1 by *2* . From now on, it will be referred to as MEAN. Two values of MEAN were selected: MEAN = (.5,.5,.5,.5) and (1,1,1,1). These choices, respectively, corresponded to low (high), and high (low) separation(overlap) between the groups. The dispersion matrix of group 2, was chosen such that it was proportional to the dispersion matrix of group 1, i.e., *2* = **I.**

Three values of (henceforth called LAMBDA) were selected: 1, 2, and 4. In the case of LAMBDA = 1, the dispersion matrices were homogeneous. For the other values of LAMBDA, the variance-covariance structures were heterogeneous. The choices of the distribution **f** were restricted to multivariate normal and lognormal, which respectively, represent symmetric and skewed distributions. Both the estimation and the holdout sample were generated from a given distribution **f** with a specific parameter configuration. The prior probabilities p*1* and p*2* each were set at .50 and the sample sizes *n1* and *n2* were proportional to these prior probabilities. The holdout sample was split the same way. Because of the preliminary nature of this Monte Carlo study, other combinations of the prior probabilities were not considered. First, the estimation sample was used to develop the classification rules, and then the efficacy of each technique was calculated by recording the percent of correctly classified observations in the holdout sample. This process was replicated five times. The five hit rates were averaged to obtain an estimate of the effectiveness of each procedure.

The above descried simulation experiment is a 2x2x3 (s distributions, 2 values of MEAN, and 3 values of LAMBDA) factorial design - yielding 12 combinations of factor levels. As all five classification approaches are applied within each of the 12 cells and the process repeated five times, the experimental design is a generalized block design [33] consisting of 12 blocks with 5 replications per block.

Results

Table 4 and 5 display the simulated mean and median hit rates of the five procedures for various settings of the parameters. To compare the relative effectiveness of these procedures, and analysis of variance (ANOVA) was performed by considering the hit rate as the dependent variable. Since hit rate is a proportion, the transformation was applied before carrying out the ANOVA procedure. The analysis showed that the mean hit rates for the five procedures are different ($F_{4,240} = 105.61$; $p < .0001$). A post-hoc multiple comparisons analysis using Duncan's procedure at the .05 level, resulted in grouping the five techniques into two distinct groups: {NN}, and {FLDA, QDA, LDA, OSD}.

Figures 7 through 10 graphically display the hit rates of the procedures as a function of the parameter LAMBDA. Figures 7 and 8 are for the normal distribution and Figures 9 and 10 are for the lognormal distribution. These figures clearly indicate that in the case of the normal distributions, the neural network approach out-performs its competitors by a wide margin for all values of LAMBDA except when LAMBDA = 1 (the case when the two groups have the same variance-covariance structures). For this specific case, the FLDA is known to be the optimal procedure and our results confirm it. For the lognormal distribution, however, the performance of the neural network-based procedure is superior to that of the other procedures regardless of the values of LAMBDA. Overall, when LAMBDA = 1 and the group separation is low, e.g., MEAN = (.5, .5, .5, .5), the gap between the hit rates of the neural network procedure and other techniques is small but it widens considerably as the values of LAMBDA increase. On the other hand, when the groups are well separated (e.g., MEAM=(1, 1, 1, 1)), the hit rates of FLDA, QDA, LDA and OSD are consistently lower than those for the neural network procedure.

This simulation study indicates that the neural network-based classification procedures are superior to the commonly-used discriminant techniques such as : FLDA, QDA, LDA and OSD. Of all the situations considered, only in one case its predictive ability lags behind that of FLDA - when the groups have multivariate normal distributions with identical dispersion matrices. As an aggregate, however, the NN classifiers show considerable advantage over the competitors. Consequently, they provide a viable and an excellent alternative to the existing repertoire of classification procedures.

(References and figures are available upon request from the first author)