Using Neural Networks To Predict the Onset of Diabetes Mellitus†

Murali S. Shanker

Department of Administrative Sciences, Kent State University, Kent, Ohio 44242

Received June 26, 1995⊗

Classification is an important decision making tool, especially in the medical sciences. Unfortunately, while several classification procedures exist, many of the current methods fail to provide adequate results. In recent years, artificial neural networks have been suggested as an alternative tool for classification. Here, we use neural networks to predict the onset of diabetes mellitus in Pima Indian women. The modeling capabilities of neural networks are compared to traditional methods like logistic regression and to a specific method called ADAP, which has been used to predict diabetes. The results indicate that neural networks are indeed a viable approach to classification. Furthermore, we attempt to provide a basis upon which neural networks can be used for variable selection in statistical modeling.

1. INTRODUCTION

Classification has emerged as an important decision making tool. It has been used in a variety of applications including credit scoring,[2] prediction of events like credit card usage[1] and tender offer outcomes,[31] and as a tool in medical diagnosis.[4,18,28] Unfortunately, while several classification procedures exist,[24] many of the current methods fail to provide adequate results.

In recent years, artificial neural networks (ANNs) have been suggested as an alternative tool for classification.[6,8] The idea of neural computing grew out of a desire to capture pattern recognition capabilities of a biological brain. McCulloch and Pitts[20] developed the first model of a physiological brain called McCulloch−Pitts neuron, which became the basis for almost all artificial neural networks where nodes are likened to neurons and arcs to dendrites or axons. Now, ANNs have been developed for recognition of such ill-defined objects as handwritten characters,[15,19] finger prints,[16] and double spirals.[13] They also have been developed for detection of faults in a chemical process,[7] explosives in airline baggage,[27] and prediction of bank failures.[30] ANNs, unlike traditional classifiers like linear discriminant analysis and quadratic discriminant analysis, are nonparametric and are able to adjust the form of the discrimination to fit the data. As ANNs can approximate, arbitrarily closely, any mapping function,[17] they might prove to be useful classification tools.

In this paper, we evaluate the effectiveness of ANNs classifiers in forecasting the onset of non-insulin-dependent diabetes mellitus among the Pima Indian female population near Phoenix, AZ.[10,11,28] The dataset used is that considered by ref 28, where they use a model called ADAP in predicting the onset of diabetes mellitus. Here, we first use ANNs to model the relationship between the onset of diabetes mellitus and various risk factors for diabetes among Pima Indian women. We then empirically compare the performance of ANNs with logistic regression and ADAP. Comparisons are made on the ability to identify significant factors and overall prediction of diabetes.

The next section describes the dataset used. Section 3 describes logistic regression and neural networks and their application to classification problems. Results of such empirical application are in section 4, and conclusions are in section 5.

2. DATA

2.1. Pima Indian Population. The population for this study is the Pima Indian female population near Phoenix, AZ. This population has been under continuous study since 1965 by the National Institute of Diabetes and Digestive and Kidney Diseases because of its high incidence rate of diabetes.[10,11] Each community resident over 5 years of age was asked to undergo a standardized examination every two years, which included an oral glucose tolerance test. Diabetes was diagnosed according to the World Health Organization criteria;[34] that is, if the 2 h post-load plasma glucose was at least 200 mg/dL (11.1 mmol/L) at any survey examination, or if the Indian Health Service Hospital serving the community found a glucose concentration of at least 200 mg/dL during the course of routine medical care.[10] This database provides a well validated data resource for exploring the prediction of the date of onset of diabetes.[22,28]

2.2. Variable Selection. Eight variables were chosen for predicting the onset of diabetes in Pima Indian women. These variables, described below, were considered as they have been found to be significant risk factors for diabetes among Pima Indians and other populations.[28] The variables chosen are

- number of times pregnant (PREGNANT)
- plasma glucose concentration at 2 h in an oral glucose tolerance test (GTT)
- diastolic blood pressure (mmHg) (BP)
- triceps skin fold thickness (mm) (SKIN)
- 2-h serum insulin (μU/mL) (INSULIN)
- body mass index (weight in Kg/(height in m)^2) (BMI)
- diabetes pedigree function (DPF)
- age (years)

The diabetes pedigree function (DPF) was developed by Smith *et al.*[28] to provide a synthesis of the diabetes mellitus history in relatives and the genetic relationship of those relatives to the subject. The DPF uses information from parents, grandparents, siblings, aunts and uncles, and first

† Key words: Neural networks, medical diagnosis, classification.
‡ Phone: (216) 672-2750; e-mail: mshanker@scorpio.kent.edu.
⊗ Abstract published in *Advance ACS Abstracts,* January 1, 1996.

0095-2338/96/1636-0035$12.00/0

cousins. It provides a measure of the expected genetic influence of affected and unaffected relatives on the subject's eventual diabetes risk. See ref 28 for further details.

2.3. Case Selection. Diabetes is defined as a plasma glucose concentration greater than 200 mg/dL 2 h following the ingestion of 75 g of a carbohydrate solution.[34] Cases selected for this study met the following criteria:

•The subject was female and older than 21 years.

•Only one examination, one that revealed a nondiabetic GTT and met either of the two following criteria, was selected per subject:

1. Diabetes was diagnosed between one and five years after the examination, or

2. Diagnosis five or more years later failed to reveal diabetes

•Cases where diabetes was diagnosed within a year of the examination were dropped from the model as they were considered potentially easier to predict.

Based on the above criteria, 768 cases were selected for analysis, of which 268 cases were diagnosed with diabetes. As the dependent variable in this case is binary valued (1 if diabetes is diagnosed, 0 otherwise), the problem evolves into one of classification.

The next section presents a brief review of logistic regression and neural networks and their application to classification problems.

3. CLASSIFICATION METHODS

3.1. Logistic Regression. The logistic regression model, or logit model, may be applied when the data consist of a binary response and a set of explanatory variables. Specifically, let the response, **Y**, of an individual take on one of two values, denoted for convenience by 0 and 1 (for example, here, $\mathbf{Y} = 1$ indicates that diabetes is present, otherwise $\mathbf{Y} = 0$). Suppose **X** is a vector of explanatory variables, and $\theta = P(\mathbf{Y} = 1|\mathbf{X})$ is the response probability to be modeled. The linear logistic model then has the form

$$\text{logit}(\theta) = \log\left(\frac{\theta}{1-\theta}\right) = \alpha + \beta'\mathbf{X}$$

where α is the intercept parameter, and β is the vector of slope parameters. Therefore, the logit is the logarithm of the *odds* of success, the ratio of the probability of success (θ) to the probability of failure ($1 - \theta$). The maximum likelihood estimates of the parameters of the logistic regression model can then be estimated using an iteratively reweighted least squares algorithm.[21] Once the parameters are estimated, it is possible to calculate the predicted probability of an individual having diabetes ($\mathbf{Y} = 1$) as follows

$$\theta = \frac{\exp(\alpha + \beta'X)}{1 + \exp(\alpha + \beta'X)}$$

A response is then classified based on the value of θ and a predetermined critical probability value.

The logistic regression model shares a common feature with a more general class of models first proposed by Nelder and Wedderburn[23] in that a function $g = g(\mu)$ of the mean of a response variable is assumed to be linearly related to the explanatory variables. Since the mean μ implicitly depends on the stochastic behavior of the response, and the explanatory variables are assumed fixed, the function g provides the link between the random (stochastic) component and the systematic (deterministic) component of the response variable **Y**. For this reason, g is referred to as a link function.[23]

While detection of outliers and other diagnostics have widespread use in linear regression, there is not yet an accepted body of methods for logistic regression. Pregibon,[25] Landwehr, Pregibon and Shoemaker,[12] and Cook and Weisberg[3] have presented some approximate diagnostics roughly equivalent to many of the methods for linear regression. Jennings[9] discusses the application of two such approximate diagnostics to logistic regression.

The next section discusses neural networks and its application to classification problems.

3.2. Artificial Neural Networks. 3.2.1. Neural Networks for Classification. An artificial neural network (ANN) is a system of interconnected units called nodes, and is typically characterized by the network architecture (layers, and connections or links among the nodes) and its node functions.

Let $G = (N,A)$ denote a neural network where N is the node set and A the arc set containing only directed arcs. G is assumed to be acyclic in that it contains no directed circuit. The node set N is partitioned into three subsets: N_I, N_O, and N_H. N_I is the set of input nodes, N_O is that of output nodes, and N_H that of hidden nodes. In a popular form called the multilayer perceptron, all input nodes are in one layer, the output nodes in another layer, and the hidden nodes are distributed into several layers in between. The knowledge learned by a network is stored in the arcs and nodes in the form of arc weights and node values called biases. We will use the term k-layered network to mean a layered network with $k - 2$ hidden layers.

When a pattern is presented to the network, the variables of the pattern activate some of the neurons (nodes). Let a_i^p represent the activation value at node i corresponding to pattern p

$$a_i^p = \begin{cases} x_i^p & \text{if } i \in N_I \\ F(y_i^p) & \text{if } i \in N_H \bigcup N_O \end{cases}$$

where x_i^p, $i = 1, \ldots, n$ are the variables of pattern p. For a hidden or output node i, y_i^p is the input into the node and F is called the activation function. The input, representing the strength of stimuli reaching a neuron, is defined as a weighted sum of incoming signals

$$y_i^p = \sum_k w_{ki} a_k^p$$

where w_{ki} is weight of arc (k,i). In some models, a variable called bias is added to each node. The activation function is used to activate a neuron when the incoming stimuli are strong enough. Today, it is typically a squashing function that normalizes the input signals so that the activation value is between 0 and 1. The most popular choice for F is the logistic function,[5,33] and it is given by

$$F(y) = (1 + e^{-y})^{-1}$$

Figure 1

Table 1.

hid
0
1
2
3
4
5
logist

Then, variable activatic and acc total is t It in tur until ev Figure network input n Connect

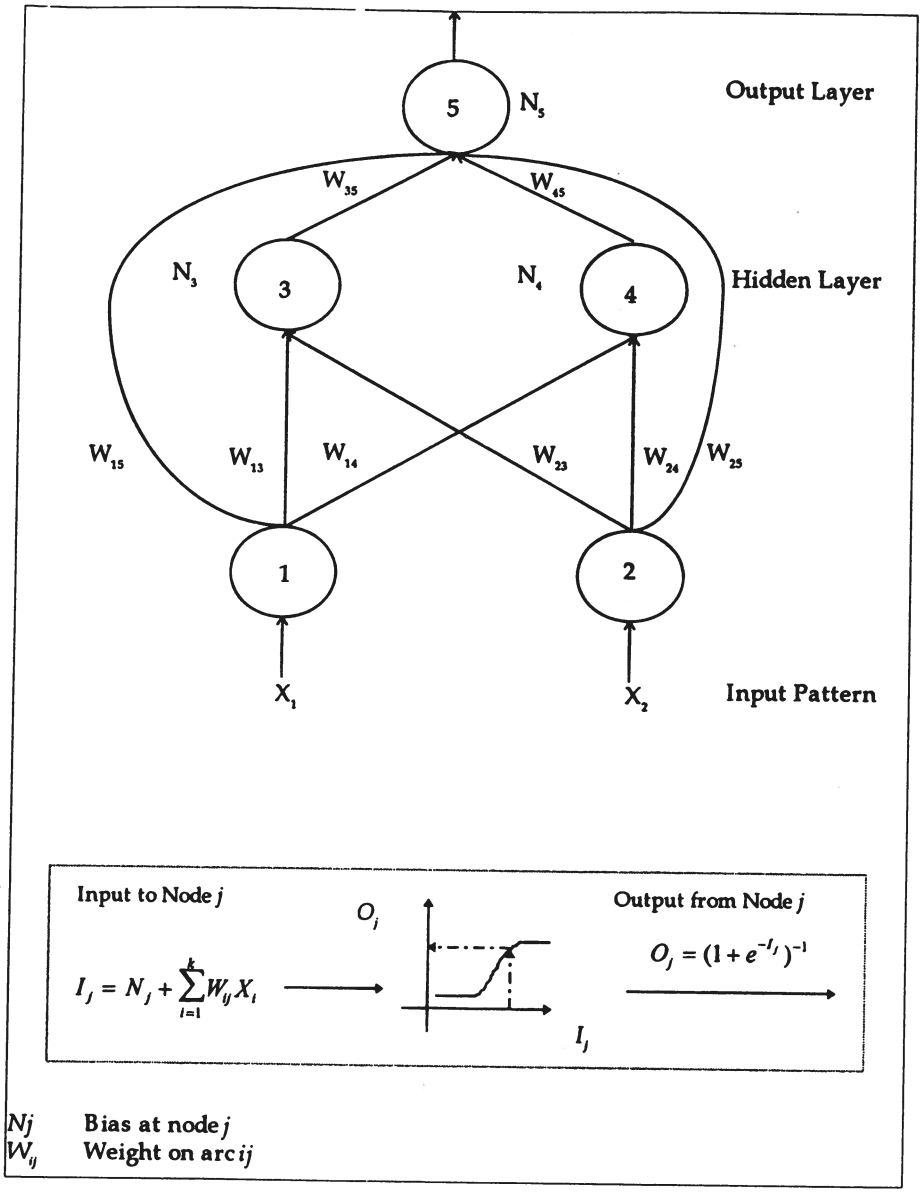

Figure 1. A neural network with two hidden nodes.

Table 1. Classification Percentage for Training and Test Samples by Hidden Nodes

	training results				test results		
hidden nodes	group 1	group 2	overall	MSE	group 1	group 2	overall
0	88.89	56.57	77.78	0.1578	90.16	58.57	78.65
1	87.30	58.59	77.43	0.1555	90.16	65.71	81.25
2	88.89	61.11	79.34	0.1557	90.16	57.14	78.13
3	90.74	64.65	81.77	0.1465	86.89	48.57	72.92
4	91.80	67.17	83.33	0.1376	81.97	48.57	69.79
5	89.42	69.70	82.64	0.1344	84.43	51.43	72.40
logistic regression	88.89	56.06	77.60		92.62	55.71	79.17

Then, the neural computing process is as follows: The variables of a pattern are entered into the input nodes. The activation values of the input nodes are weighted (with w_{ki}'s) and accumulated at each node in the first hidden layer. The total is then squashed (by F) into the node's activation value. It in turn becomes an input into the nodes in the next layer, until eventually the output activation values are computed. Figure 1 shows the basic topology of the type of neural network used in our study. The network consists of two input nodes, two hidden nodes, and one output node. Connections exist from the input nodes to the hidden nodes and also directly to the output node. Node biases exist at all hidden and output nodes, and the activation function used is the above-mentioned logistic function.

Before the network can be used for classifying a pattern, the arc weights must be determined. The process for determining these weights is called training. A training sample is used to find the weights that provide the best fit for the patterns in the sample. Each pattern has a target value t_i^p for output node i. For a two-group classification problem, only one output node is needed, and the target can be $t^p = 0$ for group 1 and 1 for group 2. In order to measure

Table 2. Classification Results Using Neural Networks with One Hidden Node

variables excluded from the model	training classification percentage	test classification percentage	no. of variables	*F*-ratio
A: Variables Dropped Include				
none	77.43	81.25	8	
PREGNANT	78.65	78.65	7	9.23
GTT	73.61	72.40	7	44.88
BP	77.43	80.73	7	8.42
SKIN	78.99	78.65	7	1.50
INSULIN	77.43	83.85	7	8.47
BMI	75.87	78.13	7	14.56
DPF	79.51	76.56	7	7.43
AGE	78.65	78.65	7	2.33
B: Variables Dropped Include SKIN and				
PREGNANT	78.30	79.17	6	4.86
GTT	73.09	66.67	6	23.31
BP	77.08	78.13	6	3.53
INSULIN	78.99	79.17	6	0.64
BMI	77.08	78.13	6	8.07
DPF	76.39	77.08	6	7.72
AGE	78.99	77.60	6	1.76
C: Variables Dropped Include SKIN, INSULIN, and				
PREGNANT	78.47	80.73	5	3.58
GTT	71.70	69.27	5	17.24
BP	77.08	77.60	5	1.35
BMI	76.91	80.21	5	4.60
DPF	78.13	78.65	5	1.45
AGE	77.78	78.13	5	2.83
D: Variables Dropped Include SKIN, INSULIN, BP, and				
PREGNANT	77.26	78.65	4	1.14
GTT	71.35	68.23	4	13.45
BMI	75.87	79.69	4	4.19
DPF	76.91	77.60	4	2.76
AGE	77.26	78.65	4	3.80
E: Variables Dropped Include SKIN, INSULIN, BP, PREGNANT, and				
GTT	73.44	69.79	3	10.95
BMI	76.04	80.73	3	3.48
DPF	77.08	80.21	3	2.17
AGE	77.26	79.69	3	4.07
F: Variables Dropped Include SKIN, INSULIN, BP, PREGNANT, DPF, and				
GTT	69.44	69.27	2	10.74
BMI	75.87	79.17	2	4.00
AGE	76.56	78.13	2	3.93

Table 3. Classification Results Using Logistic Regression

variables excluded from the model	training classification percentage	test classification percentage	no. of variables in the model	wald χ-square for variable excluded from the model
A: Variables Dropped Include				
none	77.60	79.17	8	
SKIN	77.78	79.17	7	0.0078
B: Variables Dropped Include SKIN and				
AGE	78.13	79.69	6	0.5422
C: Variables Dropped Include SKIN, AGE, and				
INSULIN	77.95	78.65	5	1.3194
D: Variables Dropped Include SKIN, AGE, INSULIN, and				
BP	76.91	80.21	4	3.6934

the best fit, a function of errors must be defined. Let E^p represent a measure of the error for pattern p

$$E^p = \sum_{i \in N_O} |a_i^p - t_i^p|^l$$

where l is a nonnegative real number. A popular choice is the least squares problem where $l = 2$. The objective is to minimize $\sum_p E^p$, where the sum is taken over the patterns in the training sample.

The neural network training system used in this research was developed by Subramanian and Hung[29] and is based on a well-known nonlinear optimizer called GRG2.[14] GRG2 is a widely distributed system, available even in popular spreadsheet programs like Microsoft Excel, and has been shown to be particularly effective for highly nonlinear problems like those in neural network training. All mathematical optimizers use strictly descend methods, which means they all converge to a local minimum. To guarantee descend, a gradient is computed after all the training patterns have been evaluated. So training epoch, in the terminology of the back-propagation algorithms, is defined for the entire training set, rather than for each training pattern. Since our algorithm is not related to back-propagation, parameters like learning rate and momentum are not necessary (see ref 29 for further details).

For classification, the output node with the maximum activation value is used to determine the class of the pattern. For example, in a neural network classifier with a single output node for two group classification, the pattern is classified as group 1 ($t^p = 0$) if the output value is less than 0.5, into group 2 otherwise. Under proper assumptions, it can be shown that the least square estimator, as our neural networks are, yields the optimal Bayesian classifier.[26,32] That is, neural network output estimates posterior probabilities. This allows one to relate neural networks to traditional statistical classifiers.

As with logistic regression, there is not yet an accepted body of works for outlier detection in neural networks. As neural network training involves least-squares estimation, the influence of outlying observations exist. But unlike linear regression, in neural networks points closer to the separation function appear to have greater influence on the separation function than points farther away.

Perhaps the critical concerns one might have in applying neural networks are problems associated with the interpretation of weights and its inability to perform statistical testing. Traditional statistical procedures rely on distributional assumptions in order to perform testing. The behavior of the error distribution in neural networks is not well understood. This study uses the Pima Indian diabetes dataset to illustrate neural networks for variable selection.

3.2.2. Analysis. Artificial neural networks are used to study the relationship between performance and the explanatory variables. As in regression, we take the approach of "parsimony" in building our neural network model to explain the onset of diabetes. As such, two interrelated questions need to be answered:

•What is the appropriate neural network architecture for this data set?

•What combination of variables will provide the best explanation for the onset of diabetes?

Network architecture refers to the number of layers, the number of nodes in each layer, and the number of arcs and nodes they connect. Other network design decisions include the choice of activation functions and whether to include biases or not. Patuwo, Hu, and Hung[24] find that networks with one hidden layer is sufficient for most problems. As such, all networks considered in this research have one hidden layer. Also, node biases occur at all hidden and output nodes, and the activation function used is the logistic

Table 4. Final Classification Results: Neural Networks vs Logistic Regression

method	training results			test results			no. of variables
	group 1	group 2	overall	group 1	group 2	overall	
A: Results Based on Final Model Chosen by Neural Network (GTT, BMI, and AGE)							
neural network	88.62	55.05	77.08	87.70	67.14	80.21	3
logistic regression	87.30	53.03	75.52	90.98	61.43	80.21	3
B: Results Based on Final Model Chosen by Logistic Regression (PREGNANT, GTT, BMI, and DPF)							
neural network	89.42	58.59	78.82	89.34	60.00	78.65	4
logistic regression	88.89	54.04	76.91	90.98	61.43	80.21	4

function. Unlike previous studies where neural network connections existed only between adjacent layers, for example, in a three-layered network between the input layer and the hidden layer but not between the input and the output layer, here we consider networks that can have direct connections between any two layers in the network. Thus, in a three-layered network, direct connections can also exist between the input and the output layer (see Figure 1). These types of networks are a superset of the networks considered previously and therefore provide greater flexibility in modeling different functional forms.

A related question is the choice and the number of variables to use for explaining the onset of diabetes. Here, we use a backward-elimination, stepwise approach for determining the subset of predictor variables to use in our neural network model. Starting with the full list of variables, at each step in our variable selection, we eliminate the variable with the smallest F ratio of all the variables in the model. We stop our process when the F-ratio for all the variables is greater than some predetermined number, say *F-OUT*.

The F-ratio used here is similar to that used in regression analysis. In the basic structure, a reduced model is compared to a full model, where the reduced model is obtained by dropping some variables from the full model. In neural networks, this is equivalent to considering fewer input nodes, where variables associated with the discarded input nodes are exempt from further consideration. The F-ratio is

$$F = \frac{\dfrac{(\mathrm{SSE}_R - \mathrm{SSE}_F)}{(\mathrm{df}_R - \mathrm{df}_F)}}{\dfrac{\mathrm{SSE}_F}{\mathrm{df}_F}}$$

where SSE_R and SSE_F represent the objective value ($= \sum_{p=1}^{n} \sum_{i \in N_O} (a_i^p - t_i^p)^2$, where the summation is over all input patterns and output nodes) for the reduced and full networks, respectively, and df_R and df_F represent the degrees of freedom of the respective models. The degrees of freedom, in general, is defined as $df =$ (number of input patterns − number of parameters estimated). For example, consider a neural network with eight input nodes, two hidden nodes, and one output node. Using Figure 1 as a reference, the number of arc weights estimated is 26. There are node biases at each hidden and output node. Therefore, the total number of parameters estimated for this neural network model is 29 (26 + 3). Our input dataset contains 768 observations, giving $df = 768 - 29 = 739$.

Clearly, $df_R \geq df_F$, since the full model would estimate more parameters than the reduced model, if the number of hidden and output nodes remain the same for both models. But, $(\mathrm{SSE}_R - \mathrm{SSE}_F)$ is not necessarily greater than or equal to zero. Unlike regression analysis, neural network training is an unconstrained, nonlinear optimization problem. As such, the global minimum for a given problem cannot be guaranteed. Therefore, it is possible that for a particular problem instance $\mathrm{SSE}_R \leq \mathrm{SSE}_F$. To ensure that the objective value (SSE) is a nonincreasing function as the number of variables in the model increase, each neural network model is trained with a large number of starting weights. Among the solutions that are determined from these various starting weights, the solution with the minimum objective value is chosen for that particular model. This strategy is repeated for all neural network models considered. While this strategy does not ensure a global minimum, it does increase the confidence in the results. For our problems, the above strategy did provide nonincreasing objective values as the number of variables in the model increased.

The validity of the F-test in regression is under the assumption of normality of residuals. As indicated by Richard and Lippmann,[26] the predicted values of neural networks approximate the posterior probabilities when the network architecture and sample size are large. The posterior probability in our study refers to the probability that a pattern belongs to group 2 (the subject developed diabetes). The output of neural networks is not normally distributed. Thus the F-ratio in our study serves only as a guide for variable selection and will not be used for significance testing.

The next section discusses the results.

4. RESULTS

As in ref 28, the entire sample of 768 was first randomly divided into a training and a test sample, with 576 cases used for training, and 192 for test. The training sample had 378 subjects without diabetes, and 198 subjects with diabetes. For the test set, 122 subjects did not develop diabetes, while 70 did.

In this study, the number of hidden nodes was varied from 0 to 5. Using the mean square error (the objective value adjusted for degrees of freedom) as a criterion, the final architecture chosen had one hidden node. As shown in Table 1, the mean square error (MSE) for the network with one hidden node is less than that of the neighboring hidden nodes, i.e., 0 and 2. While the MSE for three or more hidden nodes is smaller than that of one hidden node, our objective is to use the smallest model that will adequately predict the onset of diabetes, as such, we choose the network with one hidden node for further analysis.

This network produces a training classification percentage of 77.43 and a test classification of 81.25 (Table 1). For logistic regression, these classification percentages are 77.60 and 79.17, respectively (Table 1). In contrast, the test classification achieved by the ADAP model[28] is 76%. The authors[28] do not mention the classification on the training set.

We undertake two separate procedures to compare neural networks and logistic regression in terms of their modeling capability. The first approach relies on using the *F*-ratio in the training sample as the criterion for deleting variable(s) from the neural network model. Table 2A shows that SKIN is the prime candidate for deletion, with an *F*-ratio of 1.50. With SKIN deleted, the network with one hidden node is rerun. The *F*-ratio for INSULIN is the smallest at 0.64 (Table 2B). At the third stage, BP with an *F*-ratio of 1.35 is dropped from the model (Table 2C). The next variable to be deleted is PREGNANT with an *F*-ratio of 1.14 (Table 2D). For a predetermined *F-OUT* of, say, 3.00, the elimination process stops with DPF being dropped from the model (Table 2E). In the final model (GTT, BMI, and AGE), each variable has an *F*-ratio greater than 3.00 (Table 2F). Using these variables, the training and test classification percentages for the neural network model is 77.08 and 80.21, respectively (Table 4A). Using logistic regression with the same set of variables (Table 4A), the training and test classification percentages are 75.52 and 80.21, respectively. Thus, neural network provides better training results and test results comparable to logistic regression.

In the second comparison procedure, we apply the backward elimination procedure onto logistic regression for selection of variables. We sequentially deleted variable(s) that are the least statistically significant (at the 0.05 level) in the training sample. With all variables in the model, SKIN had a χ-square statistic of 0.0078 and is only significant at 0.9298 (Table 3A). Logistic regression was rerun with the remaining variables in the model. At this second phase, AGE was dropped from the model (p-value = 0.4615; Table 3B). INSULIN and BP were deleted at the third and fourth stages, respectively (Tables 3C and 3D). With the deletion of SKIN, AGE, INSULIN, and BP, the remaining variables were all statistically significant at the 0.05 level. As indicated in Table 4B, logistic regression with PREGNANT, GTT, BMI, and DPF has an overall training classification rate of 76.91 and a test classification of 80.21. Neural networks using the variables selected by logistic regression produced a training rate of 78.82 and a test rate of 78.65 (Table 4B). Neural networks provide better training results, but a slightly lower test classification rate than logistic regression.

As shown in Tables 2–4, both approaches to variable selection result in similar variables being retained in the model. In fact, if *F*-OUT is set to 2.00, the variables selected by neural networks are GTT, BMI, DPF, and AGE, while the backward elimination procedure in logistic regression retains GTT, BMI, DPF, and PREGNANT. Thus, the only difference is that neural network retains AGE, while logistic regression retains PREGNANT. With either set of variables, both neural network and logistic regression provide better results than ADAP.

5. CONCLUSIONS

The use of neural networks for classification is just now growing. For this particular application in predicting the onset of diabetes, we believe the results are interesting and will lead to further research on how the technique can be used for statistical testing purposes. Some previous research has indicated the performance of neural classifiers depends very much on the domain of applications. As indicated by Lippmann,[17] neural network users should first experiment on what architecture to use before finalizing on the results. Skipping this critical step will render neural classifiers less effective. In addition, prior applications of neural networks failed to recognize the sensitivity of the technique with respect to initial starting weights. Classification rates can be very different depending on the initial arc weights used.

Our study shows that neural networks are indeed appropriate for predicting the onset of diabetes. The chance probabilities of classification in training and test samples are around 55%, while the classification rates achieved by neural networks are around 78% in training and 81% in the test sample.

This study also proposes a viable approach to using neural networks as a modeling tool. The *F*-ratio is a good indicator of the variance in the dependent variable explained by the predictor variables adjusted for degrees of freedom. The results appear to support this for variable selection in neural networks. Classification results using the reduced model (Table 4) are comparable to that using all variables (Table 1). But, since the normality assumption is violated in neural networks, statistical testing is not appropriate at this time. Further research is needed in this direction.

REFERENCES AND NOTES

(1) Awh, R. Y.; Waters, D. A Discriminant Analysis of Economic, Demographic, and Attitudinal Characteristics of Bank Charge-Card Holders: A Case Study. *J. Finance* **1982**, *29*, 973–980.

(2) Capon, N. Credit Scoring Systems: A Critical Analysis. *J. Marketing* **1982**, *46*, 82–91.

(3) Cook, R. D.; Weisberg, S. *Residuals and Influence in Regression*; Chapman and Hall: New York, 1982.

(4) Crooks, J.; Murray, I. P. C. *et al.* Statistical Methods Applied to the Clinical Diagnosis of Thyrotoxicosis. *Q. J. Med.* **1959**, *28*, 211.

(5) DARPA *Neural Networks Study*; Lincoln Laboratory: MIT. 1988.

(6) Denton, J. W.; Hung, M. S. *et al.* A Neural Network Approach to the Classification Problem. *Expert Systems Applications* **1990**, *1*, 417–424.

(7) Hoskins, J. C.; Kaliyur, K. M. *et al.* Incipient Fault Detection and Diagnosis Using Artificial Neural Networks. *Proc. Int. Joint Conference Neural Networks I*, **1990**, 485–493.

(8) Huang, W. Y.; Lippmann, R. P. Comparisons Between Neural Net and Conventional Classifiers. *IEEE 1st International Conference on Neural Networks* **1987**, 485–493.

(9) Jennings, D. E. Outliers and Residual Distributions in Logistic Regression. *J. Am. Statistical Assoc.* **1986**, *81* (396), 987–990.

(10) Knowler, W. C.; Bennett, P. H.; Hamman, R. F.; Miller, M. Diabetes Incidence and Prevalence in Pima Indians: A 19-Fold Greater Incidence than in Rochester, Minnesota. *American J. Epidemiology* **1978**, *108* (6), 497–505.

(11) Knowler, W. C.; Pettitt, D. J.; Savage, P. J.; Bennett, P. H. Diabetes Incidence in Pima Indians: Contributions of Obesity and Parental Diabetes. *Am. J. Epidemiology*, **1981**, *113* (2), 144–156.

(12) Landwehr, J.; Pregibon, D.; Shoemaker, A. Graphical Methods for Assessing Logistic Regression Models (with discussion). *J. Am. Statistical Assoc.* **1984**, *79*, 61–83.

(13) Lang, K. J.; Witbrock, M. J. Learning to Tell Two Spirals Apart. *Proceedings 1988 Connectionists Models Summer School* **1988**, 52–59.

(14) Lasdon, L. S.; Waren, A. D. *GRG2 User's Guide*; School of Business Administration: University of Texas at Austin, Austin, TX. 1986.

(15) Le Cun, Y.; Boser, B.; et al. Handwritten Digit Recognition with a Back-Propagation Network *Adv. Neural Inf. Processing Systems* **1990**, 2, 396–404.

(16) Leung, M. T.; *et al.* Fingerprint Processing Using Backpropagation Neural Networks. *Proc. Int. Joint Conference Neural Networks I* **1990**, 15–20.

(17) Lippmann, R. P. An Introduction to computing with neural nets. *IEEE ASSP Magazine* **1987**, *4*, 2–22.

(18) Mangasarian, O. L.; Setiono, R.; Wolberg, W. H. Pattern Recognition Via Linear Programming: Theory and Application to Medical Diagnosis. *Proc. Workshop Large-Scale Numerical Optimization* **1989**, 22–30.

(19) Martin, G. L.; Pittman, J. A. Recognizing Hand-Printed Letters and Digits. *Adv. Neural Inf. Processing Systems* **1990**, 2, 405–414.

(20) McC
Ner
133.

(21) McC
Hall

(22) Mur
Dat
Info
CA.

(23) Nelc
Roy

(24) Patu
Net

(25) Preg
198

(26) Rich
Bay
461

(27) She
in th
Int.

(20) McCulloch, W. S.; Pitts, W. A Logical of the Ideas Immanent in Nervous Activity. *Bulletin Mathematical Biophysics* **1943**, *5*, 115−133.
(21) McCullagh, P.; Nelder, J. *Generalized Linear Models*; Chapman Hall: London, 1983.
(22) Murphy, P. M.; Aha, D. W. *UCI Repository of Machine Learning Databases (Machine-Readable Data Depository*; Department of Information and Computer Science: University of California, Irvine, CA.
(23) Nelder, J. A.; Wedderburn, R. W. M. Generalized Linear Models. *J. Royal Statistical Society, Ser. A* **1972**, *135*, 370−384.
(24) Patuwo, E.; Hu, M. Y. *et al.* Two-Group Classification Using Neural Networks. *Decision Sciences* **1993**, *24*(4), 825−845.
(25) Pregibon, D. Logistic Regression Diagnostics. *The Annals of Statistics* **1981**, *9*, 705−724.
(26) Richard, M. D.; Lippmann, R. Neural Network Classifiers Estimate Bayesian A Posterior Probabilities. *Neural Computation* **1991**, *3*, 461−483.
(27) Shea, P. M.; Liu, F. Operational Experience with a Neural Network in the Detection of Explosives in Checked Airline Baggage. *Proc. Int. Joint Conference Neural Networks II* **1990**, 175−178.
(28) Smith, J. W. *et al.* Using the ADAP Learning Algorithm to Forecast the Onset of Diabetes Mellitus. *Proc. Twelfth Annual Symposium Comput. Applications Medical Care* **1988**, 261−265.
(29) Subramanian, V.; Hung, M. S. *A GRG2-based System for Training Neural Networks*: *Design and Computational Experience*. ORSA J. Computing, **1993**, *5*(4), 386−394.
(30) Tam, K. Y.; Kiang, M. Y. Managerial Application of Neural Networks: The Case of Bank Failure Predictions. *Management Sci.* **1992**, *38*(7), 926−947.
(31) Walking, R. A. Predicing Tender Offer Success: A Logistic Analysis. *J. Finance Quantitative Analysis* **1985**, *20*, 461−478.
(32) Wan, E. A. Neural Network Classification: A Bayesian Interpretation. *IEEE Transactions Neural Networks* **1990**, *1*(4), 303−305.
(33) Wasserman, P. D. *Neural computing*: *Theory and Practice*; Van Nostrand Reinhold. 1989.
(34) *WHO Technical Report Series*, No. 727, (Report of a WHO Study Group) 1985.

CI950063E