Volume 44 Issue 4 March 2008 ISSN 0167-9236

ELSEVIER

Decision Support Systems
and Electronic Commerce

www.elsevier.com/locate/dss

Available online at www.sciencedirect.com

ScienceDirect

Decision Support Systems 44 (2008) 899 – 908

Decision Support Systems

www.elsevier.com/locate/dss

Modeling consumer situational choice of long distance communication with neural networks

Michael Y. Hu [a], Murali Shanker [a], G. Peter Zhang [b,*], Ming S. Hung [c]

[a] Kent State University, United States
[b] Georgia State University, United States
[c] Optimal Solutions Technologies, United States

Received 5 December 2006; received in revised form 26 July 2007; accepted 15 October 2007
Available online 22 October 2007

Abstract

This study shows how artificial neural networks can be used to model consumer choice. Our study focuses on two key issues in neural network modeling — model building and feature selection. Using the cross-validation approach, we address these two issues together and specifically examine the effectiveness of a backward feature selection algorithm for consumer situational choices of communication modes. Results indicate that the proposed heuristic for feature selection is robust with respect to validation sample variation. In fact, the feature selection approach produces the same best subset of features as the all-possible-subset approach.

Keywords: Consumer choices; Neural network; Feature selection; Model building; Prediction risk

1. Introduction

Understanding consumer choice is crucial to effective marketing management. Previous studies have shown that consumer choice is a function of consumer demographics, psychographics, the consumption motives and goals [4,20], and the specific consumption situational context [32]. Accurate information on the relative importance of these variables makes it possible for firms to price and promote their products and services more effectively.

Logit models are traditionally used for predicting consumer choices [7,31]. These models are useful for understanding and predicting brand choice behavior and examining the effects of marketing mix and demographic variables on consumers' choice of products. The limitation of this model, however, is the essentially linear form of the utility function that is used to calculate the probability or odds-ratio of making a specific choice. Although nonlinear terms such as interactions could be added into the model, the inclusion of such terms requires knowledge of the underlying structure of the utility function.

Artificial neural networks are a promising modeling tool to overcome the above-mentioned limitation of logit model as well as other linear parametric models used in modeling consumer choices. Neural networks belong to a class of flexible nonparametric, nonlinear regression models that do not impose *a priori* restrictions on the type of relationship to be modeled. Rather, the relationship is

* Corresponding author.
E-mail addresses: mhu@kent.edu (M.Y. Hu), mshanker@kent.edu (M. Shanker), gpzhang@gsu.edu (G.P. Zhang), mhung@optimize.com (M.S. Hung).

0167-9236/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2007.10.009

established through multiple iterations of training on observed sample data. This adaptive learning-from-data property is a powerful approach for pattern recognition and pattern classification.

Neural networks have enjoyed increasing popularity and have been applied to a large number of business problems. For example, Hansen and Nelson [18] use neural networks and traditional time series methods to forecast state tax revenues. Dhar and Chou [13] compare a number of nonlinear methods for predicting earnings surprise and returns. In marketing, we have found applications of neural networks in market share forecasting [2], market response prediction [11,39], customer targeting [26], repeat purchase purchasing modeling in direct marketing [5], market segmentation [24,28,40], and consumer brand choice modeling [6,41].

As West et al. [41] point out in modeling consumer choice with neural networks, marketing researchers typically treat neural network models as a black box. It is well known that neural network models are highly dependent upon the model architecture — number of hidden layers, hidden nodes and arcs. Since the power of neural network models has not been fully appreciated by most marketing researchers, these models have made very limited inroads into the standard toolbox of the researchers. The hesitation on the part of marketing researchers is also due to their limited understanding of how neural networks can be used for feature selection. Most neural network applications in consumer choice models rely on logit models for variable/feature selection before subjecting the reduced set of features to neural network models for prediction purposes. For example, Kim [25] uses the SAS logistic regression procedure with forward variable selection to select the subset of variables that are further fed into a neural network model. This approach may cause potential biases as the selection procedure is based on the predetermined structured model rather than a more general and flexible neural network model.

In this paper, we present a case study on neural network model building for consumer situational choice for long distance communication. Building a successful model that relates important consumer characteristics such as demographic and situational factors to their choice among various modes of communication is a valuable exercise to telecommunication companies. The model can help focus their effort in planning, advertising, and target marketing and thus enhance a company's competitive position. In the process, we highlight the critical issues in building neural networks:

- Model selection: determining an appropriate architecture (number of hidden nodes) for the neural network. Selection of an appropriate model is a nontrivial task. One must balance *model bias* (accuracy) and *model variance* (consistency). A more complex model tends to offer smaller bias but greater variance. Among neural networks, a larger network tends to fit a training data set better but may perform poorly when it is applied to new data. A more detailed discussion will be presented in Section 2.1.
- Feature selection: determining an appropriate feature subset from all candidate features. As all modeling efforts should strive to achieve parsimony, the goal here is to build a model with the fewest number of independent variables yet producing equal or comparable predictive power as larger models. For neural networks, as statistical parameter or model testing is difficult to apply, more computational intensive methods must be employed to determine the variables that should be included in a model.

Although these issues are well known, they are often overlooked in many neural network application studies. In addition, these issues, particularly feature selection, have not been systematically examined in the literature pertaining to marketing applications. Unlike many other applications that separate feature selection and model selection, in this paper, we treat both as integral parts of the model building process. In other words, we propose a disciplined approach in applying neural networks for consumer choice modeling. We believe that the systematic modeling approach proposed in this paper can be used in other marketing applications. Another contribution of this study is to formally validate our heuristic feature selection procedure. We show that the backward selection method is equivalent to the all-possible-subset approach in identifying the best subset of feature variables, and therefore is computationally efficient and effective for practical uses.

The rest of the paper is organized as follows. In the next section, we provide theoretical results of model selection and survey the literature on feature selection. Then in Section 3, we describe our research design and data. Results are presented in Section 4 Finally, Section 5 provides concluding remarks.

2. Model and feature selection

2.1. Model selection

Model selection addresses the issue of what is the appropriate neural network model for a given sample. Theoretically, model selection is based on the trade-off between *model bias* and *model variance* [16]. The bias of a model relates to the predictive accuracy of the model, whereas variance refers to the variability in the predictions. A model with low bias – by having many hidden nodes, for example – tends to have high variance. On the other hand, a model with low variance tends to have high bias.

While ideally we would like to have a model with both low bias and low variance, it is practically difficult to achieve these at the same time for a given data set. A model that is less dependent on the data tends to have low variance but high bias if the pre-specified model is incorrect. On the other hand, a model that fits the data well tends to have low bias but high variance when applied to different data sets. Hence a good predictive model should have an "appropriate" balance between model bias and model variance.

As a model-free approach to data analysis, neural networks often tend to fit the training data well and thus have low bias. But the price to pay is the potential overfitting effect that causes high variance. Dietterich and Kong [14] point out in the machine-learning context that the variance is a more important factor than the bias in poor prediction performance. Friedman [15] finds that for classification, neural networks provide unstable predictions in that small changes in the training sample could cause large variations in the test results. Much attention has been paid to the overfitting problem in the literature with a majority of research devoted to methods of reducing the overfitting effect. For a review of the methods as well as some other related issues, see [8].

Thus, in practical modeling applications, we strive to select the smallest neural network model that learns well the underlying relationships in the data. A well-trained parsimonious model should theoretically generalize better to new data than an overfitted model that memorizes all the details of the training data. One popular way to determine model generalizability is to use the cross-validation approach with data divided into two major portions of in-sample and out-of-sample. The in-sample data are used for parameter estimation and model selection and then the selected model is further tested on the out-of-sample. If performance of out-of-sample is similar to that of in-sample, then the model can be considered to have learned and generalized well. Adya and Collopy [1] further point out that a useful prediction model must be verified on both generalization and stability, which are especially important for powerful model like neural networks. Stability is the consistency of results, during the validation phase, with different samples of data. In fact, both generalizability and stability are issues that are more related to model variance. Thus, our goal in model building and selection is to find a small network that is well trained and has good generalizability and stability.

2.2. Feature selection

Feature selection is an important component of model building. The issue of feature selection is also closely related to the bias-variance or learning-generalization trade-off discussed above. The objective of feature selection is to identify a small number of features that contribute the most to model learning. Therefore, feature selection is an effective approach to dealing with dimensionality reduction that is the key to reduced overfitting effect and improved predictive performance. In general, for predictive model building purposes, the principle of parsimony should always be followed. It is necessary and desirable to have a small number of input features in order to develop a good predictive and less computationally intensive model.

The literature contains numerous studies on feature selection. Statistical feature selection approaches are widely available, but they are not directly applicable to neural networks since most of these statistical methods assume linearity and normality in the correlation structure. On the other hand, neural network-based approaches are mostly heuristic in nature. Some methods are based on ideas borrowed from their statistical counterparts while others are focused on the neural network architecture to support the removal or addition of features.

Since exhaustive search through all possible subsets of feature variables is often computationally prohibitive, most of the feature selection methods use stepwise search algorithms such as forward addition and backward elimination approaches similar to those commonly used in linear statistical modeling. The forward addition approach successively adds one variable at a time, starting with one variable, until no attractive candidate remains. The backward elimination approach starts with all variables in the model and successively eliminates one at a time until only the "good" ones are left. Most of the feature selection algorithms in neural network research are based on the backward sequential method. For example, [9,17,21,34,36–38] describe different backward elimination algorithms. It is important to note that most feature selection algorithms rely on some type of saliency measures that are used to assess a feature's relative importance. Numerous saliency measures are proposed but none of them is universally accepted in the literature. Steppe and Bauer [37] presents a comprehensive survey of various feature saliency measures used in neural networks. They summarize all measures into two categories: measures of relative changes in either neural network outputs or neural network's probability of error and measures of relative size of weight vector emanating from each feature.

Almost all feature selection criteria and search algorithms in neural networks are heuristics, and statistical tests are usually not valid to justify the removal or addition of a feature. Hence their performance may not be consistent and robust in practical applications. Recent comparative studies [12,27,37] on feature saliency measures and search methods suggest that none of the available feature selection approaches is universally the best for all types of problems. Thus, developing and evaluating more effective methods remains an area of interest for researchers in pattern classification and other related areas.

3. Research design

3.1. Data and cross-validation experiment

The American Telephone and Telegraph Company (AT&T) maintained a residential consumer diary panel to study the consumer choice behavior in selecting long distance communication modes over time [30]. The company embarked on a major research effort to understand the effect of situational influences on consumer choices of communication modes. It is envisioned that the usage of long distance phone calling is largely situational since the service is readily available within a household and is relatively inexpensive. A demographically proportional national sample of 3990 heads of households participated in the study over a twelve-month period in early 1980s. The sample was balanced with respect to income, marital status, age, gender, population density and geographic region. Each participant has to record the specifics on a weekly basis of one long distance (50 miles or more) communication situation.

The communication modes being reported are of three types, long distance telephone calling (LD), letter or card writing. Since long distance telephone calling is verbal and the other two are non-verbal, letter and card in this study are combined into one category. The dependent variable, COMMTYPE, is coded as '1' for LD and '0' for 'letter and card'.

In a pre-diary survey, each respondent was asked to provide information on the usage rate of LD (MEANCALL) and written communications (MEANLET) in a typical month. Each diarist also provided information on five communication situation related variables for a specific communication that has taken place in a diary week. The selection of these factors is based on research findings in [22,30]. These input variables will be treated as initial feature variables in our modeling effort. The seven variables are presented as follows:

1. MEANLET: Average number of cards and letters combined in a typical month;
2. MEANCALL: Average number of calls in a typical month;
3. TYCALL: the nature of the communication decision, whether it is 'impulse' (coded as '0') or 'planned' (coded as '1');
4. REASON: Reason for communication, 'ordinary' (coded as '1') or 'emergency' (coded as '0');
5. RECEIVER: Receivers of the communication, 'relatives' (coded as '1') or 'friends' (coded as '0');
6. NUMCALL: Total number of LD calls made and received in a particular week, and
7. NUMLET: Total number of letters/cards sent and received in a particular week.

In the rest of the paper, these variables will be referred as variable 1 through variable 7.

As detailed in the following sections, we propose a backward-elimination procedure for feature selection. An experiment was conducted to evaluate our procedure, and it consisted of training neural networks with all possible combinations of the feature variables and computing the prediction risks of each trained network. Results from the backward elimination procedure were then compared with those from all possible combinations.

A random sample of 3377 communication situations is drawn from the weekly diary database, where 1595 (47.23%) entail LD calls and the remaining 1782 (52.77%) written communications. The entire sample of situations is from a total of 2111 diarists. The maximum number of situations is eight per diarist. Of these 3377 observations, we randomly pick 1535 observations as training data and the remaining 1842 as validation data. To measure the robustness of the backward elimination procedure, the validation sample is randomly subdivided into 3 sets of equal size with set 1 composed of 286 LDs and 328 written cases; set 2 of 278 LDs and 336 written cases, and set 3 of 299 LDs and 315 written cases. The proportion of LD/Written is not necessarily the same in each validation sample. This cross-validation scheme will provide useful information on how sensitive model selection is with respect to the validation samples. Table 1 shows the means and standard deviations of variables used in our study. Across the three validation samples the statistics are generally similar.

Table 1
Variable means and standard deviations

Variables	Training		Validation sets					
			1		2		3	
	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation
COMMTYPE	.48	.013	.47	.020	.45	.020	.49	.020
MEANLET	6.54	7.941	6.75	8.066	6.15	7.201	6.73	7.994
MEANCALL	4.16	4.517	4.26	4.139	4.03	4.247	4.27	4.434
TYCALL	.33	.012	.31	.019	.34	.019	.26	.018
REASON	.84	.009	.83	.015	.87	.014	.85	.014
RECEIVER	.44	.013	.45	.020	.44	.020	.46	.020
NUMCALL	1.92	2.597	1.94	2.623	1.78	2.311	1.83	2.197
NUMLET	3.61	4.677	3.56	4.691	3.64	5.165	3.36	4.077

3.2. Neural networks

Neural networks are computing models for pattern recognition and pattern classification. They learn from examples or experiences and are particularly noted for their flexible nonlinear function mapping ability. A neural network is a layered computing system of interconnected nodes that performs functional mapping from input layer to output layer. It is characterized by the network configuration such as number of layers, connections among the nodes, as well as the linking functions between layers. In this study, we employ feedforward networks with one hidden layer. For more information about the basic ideas and issues of neural networks for classification, readers are referred to [8].

Let $\mathbf{x}=(x_1,x_2,\ldots,x_m)$ be a vector of m input variables, Y be the output from the network, and $\mathbf{w}_1$ and $\mathbf{w}_2$ be the matrices of linking weights from input to hidden layer and from hidden to output layer, respectively. Then a three-layer neural network is a nonlinear model of the form

$$Y=f_2(\mathbf{w}_2 f_1(\mathbf{w}_1\mathbf{x})), \qquad (1)$$

where f_1 and f_2 are the transfer functions for the hidden nodes and output nodes respectively. The most popular choice for f_1 and f_2 is the sigmoid (logistic) function given by

$$f(z)=(1+e^{-z})^{-1} \qquad (2)$$

It has been shown that this type of simple structured network can approximate any type of nonlinear functions. In addition, the output from the neural network will be an unbiased estimate of the posterior probability $P(Y=1|\mathbf{x})$ which plays an important role in the Bayesian classification theory.

In a classification problem, determining the neural network architecture is equivalent to specifying the number of hidden nodes since the number of input and output nodes are usually determined by the problem characteristics. The number of input nodes is equal to the number of predictor variables in the data set. In this study, all networks have one output node, since there is one target variable COMMTYPE, and one hidden layer with h hidden nodes. There are arcs directly connecting each input node to both the output node and the hidden nodes. In addition, each hidden node has a scalar. For the purpose of model selection, the number of hidden nodes h varies from 0 to 7. Typically, the size of a neural network refers to its number of parameters (i.e., the number of arc weights and node biases). Given that we are concentrating on networks of one layer, the size of a network is directly related to the number of hidden nodes.

We use a self-developed neural network program written in C. Neural network training is based on an algorithm [3] which starts with a network with zero hidden nodes and assigns the initial parameter values found in linear regression as the starting values of the arc weights. Then a hidden node is added. The weights of the arcs in the previous network are used as starting weights. The starting weights for the new arcs are determined by a scheme designed to maximize the reduction in the sum of squared errors (SSE). This algorithm employs the second-order (the limited-memory quasi Newton) method to solve the nonlinear optimization problem and is quite robust as shown in the experiments by Ahn [3].

The methods to determine the appropriate network architecture can be summarized as follows.

1. Eliminating arcs whose weights are small or insignificant. For example, [10] constructs an approximate confidence interval for each weight, and if it contains zero, then the arc is eliminated.
2. Eliminating arcs whose *saliency* measure is small. Saliency is typically based on the partial derivative of the SSE with respect to the arc. Methods differ in the

approximation of this derivative. The *optimal brain damage* of [29] defines saliency of arc i as $H_{ii}w_i^2/2$ where H_{ii} is the i-th diagonal element of the *Hessian* matrix, the matrix of second derivatives (of SSE with respect to arc weights), and w_i is the weight of arc i. The *optimal brain surgeon* [19], on the other hand, uses the diagonal element of the inverse of the Hessian matrix.

3. Building networks with different numbers of hidden nodes and then selecting one based on some performance measures in the validation sample. For example, the measure used by Moody and Utans [33] is the *prediction risk* discussed below and it is the mean squared error on the validation set, adjusted by the number of weights. They also compute the prediction risk by using cross-validation, which first divides a data set into k subsets and uses $k-1$ subsets for training and the kth subset for validation. The validation set then rotates to the first subset, and then to the second, etc. in a round-robin fashion. In this study, we use the cross-validation approach in model selection with a measure similar to that of [33]. For other methods, readers are referred to [8].

3.3. Feature selection

In this paper, we use a method for feature selection based on our measure of prediction risk, which is quite similar to that of Moody and Utans [33]. Moody and Utans' variable elimination approach is based on sensitivity analysis under the framework of prediction risk. The prediction risk is defined as the expected performance of a model in predicting new observations and can be estimated by cross-validation method. Since the prediction risk provides a practical way to measure the generalization ability—the core of any feature selection method, we believe this approach is a viable one for feature selection and model building.

Given a trained network of n features and h hidden nodes, denoted as M_n^h, the prediction risk can be estimated as the mean sum of squared errors (SSE) of a validation set V. That is,

$$\begin{aligned}\mathrm{MSE}\left(M_n^h\right) &= \frac{1}{|V|}\mathrm{SSE}\left(M_n^h\right)\\ &= \frac{1}{|V|}\sum_{p=1}^{|V|}\sum_{j=1}^{l}\left(Y_j^p - T_j^p\right)^2\end{aligned} \tag{3}$$

where $|V|$ is the number of patterns in the validation set: $V=(Y,T)$, where T is the matrix of target values, Y the output of the network, and l the number of output nodes of the neural network M_n^h. As the validation sets in our study are all of the same size, we use the sum of square error $\mathrm{SSE}(M_n^h)$ as a measure of prediction risk in our research. The procedure is detailed below:

1. Start with all n features and train a network over a range of hidden nodes; i.e., $h=0, 1, 2,\ldots$
2. Select the optimal hidden nodes $h*$ which yields the smallest sum of squared errors $\mathrm{SSE}(M_n^h)$.
3. Reduce the number of features by 1, and train every possible $(n-1)$ feature network with h^* hidden nodes. Let $\mathrm{SSE}{*}(M_{(n-1)}^{h*})$ indicate the network with the smallest SSE of the $(n-1)$ networks.
4. If $\mathrm{SSE}'(M_{(n-1)}^{h'}) \leq \mathrm{SSE}^*(M_n^{h'})$, then $n=(n-1)$, and go to Step 3; otherwise, go to Step 5.
5. Use the features selected in Step 3, train networks over the range of hidden nodes used in Step 1 and select the optimal hidden nodes $h*$ again.

Although it is possible to search for the best network architecture in terms of the number of hidden nodes at each step of feature selection process, there are several reasons not to do it. First, the fixed hidden node structure allows us to determine unequivocally that the reduction of SSE is attributed to the reduction of feature variables. If different hidden nodes are used in each step of the feature selection process, it is difficult to tell whether the reduction of SSE is due to the change of feature variables or hidden nodes or the combination. Second, with fixed hidden nodes, when the number of features is reduced, the number of model parameters to be estimated also becomes smaller, resulting in more degrees of freedom and increased statistical power. Finally, using one network structure can reduce the time and effort in the modeling process significantly.

4. Results

To evaluate the effectiveness of the feature selection procedure, we consider all possible subsets of the seven potential feature variables identified. With all-possible-subset results, we are able to compare results from our feature selection method to those obtained from the best combination of features. In this AT&T situational choice study, we are able to consider all possible subsets due to the small number of feature variables. It will be very difficult if not impossible to experiment all possible subsets when the number of features is large.

A neural network was set up for each of the 127 possible subsets of the seven input variables. Each network was then trained using 8 different architectures (0

Table 2
Minimum SSE across hidden nodes and number of variables

# of variables	Number of hidden nodes							
	0	1	2	3	4	5	6	7
Validation sample 1								
1	114.68	106.13	106.13	103.87	103.87	115.04	114.74	115.24
2	101.40	84.45	77.78	78.81	79.54	80.27	81.80	80.83
3	98.74	79.82	73.72	74.70	76.30	77.31	77.48	76.72
4	95.45	76.91	70.82	71.54	73.03	73.18	73.74	73.97
5	92.88	74.38	68.68	70.23	69.95	73.18	74.66	75.45
6	92.24	75.37	**68.62**	70.73	72.37	72.88	73.32	75.29
7	92.29	75.51	73.73	74.38	77.65	78.31	80.84	82.72
Validation sample 2								
1	115.19	103.11	103.11	98.27	98.27	110.73	109.94	110.01
2	87.17	80.58	69.54	70.37	70.17	70.86	71.76	72.37
3	86.21	79.44	67.70	68.09	68.66	70.25	70.47	70.85
4	83.27	75.63	64.50	65.06	66.24	67.17	67.31	68.06
5	82.74	74.29	63.19	64.78	64.98	66.51	69.43	70.18
6	82.88	73.63	**61.80**	63.87	64.25	64.63	65.93	66.79
7	83.14	73.67	66.46	67.73	71.31	74.24	74.65	75.46
Validation sample 3								
1	118.07	108.24	108.24	108.17	108.17	111.93	111.89	112.19
2	96.29	84.18	75.00	75.19	75.74	76.64	76.51	76.97
3	94.76	83.90	75.08	74.04	75.62	74.89	75.04	77.15
4	91.91	79.41	**72.06**	72.48	72.74	73.20	74.67	75.80
5	91.26	78.85	73.11	73.23	72.66	75.55	76.11	78.29
6	91.52	79.74	74.03	75.55	76.09	75.21	77.68	77.04
7	91.73	80.57	76.80	76.13	78.08	78.10	78.66	80.14

to 7 hidden nodes). These correspond to a total of 1016 networks. Table 2 shows the minimum SSEs across all hidden nodes and subsets of feature variables for each validation sample. In validation sample 1, among the seven 1-variable networks, variable 6 (not shown) with 4 hidden nodes is tied with variable 6 with 3 hidden nodes with SSE equal to 103.87. Among the 6-variable networks, the network with 2 hidden nodes has the minimum SSE of 68.62. The network with the smallest SSE among all combination of variables and hidden nodes is shown in bold.

Results from validation sample 2 are similar to those from sample 1. Both indicate that the 6-variable network with variables 2, 3, 4, 5, 6 and 7, and 2 hidden nodes has the smallest SSE. Validation set 3 shows a slight difference from the other two samples. The 4-variable (variables 4, 5, 6, and 7) with two hidden nodes has the smallest SSE.

Next, we experiment with the backward elimination procedure. The seven input variables were trained in eight network architectures, hidden nodes from 0 to 7. With validation sample 1, Table 2 shows that the network with two hidden nodes has the smallest SSE of 73.73 for seven variables. With the number of hidden nodes fixed at 2, we then proceeded to examine the SSEs from the seven 6-variable networks. As shown in Table 3, the network with variables 2, 3, 4, 5, 6, and 7 has the smallest SSE, 68.62. Further elimination of variables resulted in an increase in SSE. The set of variables 2, 3, 4, 5, 6, and 7 is then used to train networks with 0 to 7 hidden nodes, and the minimum SSE, shown in bold, corresponds to the network with two hidden nodes (see Table 4). So the recommended feature set, based on validation sample 1, is variable combination of 2, 3, 4, 5, 6, and 7. The best network architecture is the one with two hidden nodes. This is the same "best" selection suggested by the all-subset experiment (Table 2).

With validation sample 2, the backward elimination method ends up with the same "best" selection. The minimum SSE is 61.80. For validation sample 3, the backward elimination method starts with two hidden nodes for all seven variables and ends up with four variables—namely 4, 5, 6, and 7. Table 3 shows that the SSE for this subset is 72.48. The set of four variables is then used to train networks, and the minimum SSE corresponds to the network with two hidden nodes (see Table 4). This is the same as the best selection in the all-subset procedure (Table 2).

Overall results indicate that the feature selection procedure identifies the same "best" models as the all-

Table 3
Backward elimination procedure for all validation samples

Validation sample 1		Validation sample 2		Validation sample 3	
Variables selected	SSE	Variables selected	SSE	Variables selected	SSE
1234567	73.73	1234567	66.46	1234567	76.13
Start with the above 7 variable model					
123456	89.44	123456	84.45	123456	95.78
123457	97.65	123457	89.43	123457	98.32
123467	71.71	123467	68.89	123467	80.13
123567	75.71	123567	68.16	123567	79.51
124567	77.02	124567	67.89	124567	76.97
134567	72.87	134567	64.91	134567	76.12
234567	**68.62**	**234567**	**61.80**	**234567**	**75.55**
Use the best 6 variable model (shown in **bold** *above)*					
23456	90.30	23456	91.57	23456	98.27
23457	97.53	23457	90.08	23457	97.47
23467	71.31	23467	67.68	23467	76.57
23567	71.51	23567	64.73	23567	76.45
24567	75.40	24567	65.36	24567	78.70
34567	**68.68**	**34567**	**63.19**	**34567**	**73.23**
Use the best 5 variable model					
3456	91.50	3456	93.14	3456	97.27
3457	98.14	3457	93.40	3457	100.21
3467	70.98	3467	66.28	3467	75.30
3567	70.87	**3567**	**64.50**	3567	74.90
4567	**70.82**	4567	65.31	**4567**	**72.48**
Use the best 4 variable model					
456	93.66	356	99.02	456	96.93
457	103.02	357	99.02	457	100.36
467	**79.65**	**367**	**67.70**	**467**	**74.04**
567	132.21	567	106.76	567	116.07
Use the best 3 variable model					
46	97.94	36	100.87	46	100.72
47	108.73	37	105.91	47	107.14
67	**77.78**	**67**	**69.54**	**67**	**75.19**
Use the best 2 variable model					
6	**106.13**	**6**	**103.11**	**6**	**108.17**
7	119.37	7	112.39	7	113.74

Table 4
SSE across hidden nodes

Hidden nodes	Validation sample 1	Validation sample 2	Validation sample 3
	Variables: 234567	Variables: 234567	Variables: 4567
0	92.24	82.88	91.91
1	75.37	73.63	79.41
2	**68.62**	**61.80**	**72.06**
3	70.73	63.87	72.48
4	72.37	64.25	72.74
5	72.88	64.63	73.20
6	73.32	65.93	76.39
7	75.29	66.79	76.28

possible-combination approach in all three validation samples. This suggests that the feature selection algorithm based on the prediction risk is quite robust judged from generalization ability for new observations. In addition, we find that networks with 2 or 3 hidden nodes are appropriate for our application.

From a practical standpoint, there seems to be little difference between models of six features and those of four features. In validation samples 1 and 2, the four-variable models end up with only a slight increase in SSE over the six-variable models. For example, in validation sample 1, the four-variable model with variables of 4, 5, 6, and 7 leads to an SSE of 70.82 compared to the smallest SSE of 68.62 for the six-variable model. However, a four-variable network with two hidden nodes has only 14 arcs, whereas a 6-variable network with the same number of hidden nodes has 20 arcs. It may be beneficial to use the four-variable model because of the significant reduction in the size of the network and the number of variables while achieving almost the same level of accuracy.

Tables 5 and 6 report the classification results using both the neural network models and the logistic regression models for the training sample and three validation samples, respectively. The results suggest that both logistic regression and neural network model classify the written communication group (COMMTYPE = 0) more accurately than the LD group (COMMTYPE = 1). In all cases, neural networks outperform the logistic regression judged from both the overall classification rate and the group classification percentages. The results are very robust because neural network models provide similar and consistently better predictions not only in all three validation samples, but also in the training sample.

5. Conclusions

Major advances have been made in the past decade in neural networks for pattern recognition and classification. Applications of neural networks in marketing research are just now emerging. It is our hope that marketing researchers will be able to gain a better appreciation of the technique. Of course, these advances

Table 5
Percent correct classification with the training sample

COMMTYPE	Logistic	Neural network
0	81.80	84.08
1	76.43	81.34
Overall	79.20	82.80

Table 6
Percent correct classification with three validation samples

COMMTYPE	Logistic			Neural network		
	1	2	3	1	2	3
0	80.18	80.65	81.33	82.93	85.12	84.18
1	77.27	78.78	74.92	80.42	81.65	81.94
Overall	78.83	79.80	78.77	81.76	83.55	83.09

are available at a cost. Neural networks are much more computationally intensive than classical statistical methods such as logistic regression. The model selection and feature selection procedures require customized programs. However, as computation cost is getting cheaper each day, these problems become less of an obstacle for modelers. Neural networks are similar to nonparametric statistical techniques in that no distributional assumptions are needed. We believe that neural networks have the advantages over nonparametric techniques because they are data-driven and no model specification is necessary. Therefore, they are particularly useful in social science disciplines for theory development.

Marketers are particularly interested in consumer choice factors because accurate information on the relative importance of these factors makes it possible for firms to better position their products and services in the marketplace. Our study illustrates how a marketer can more effectively select a network architecture and a subset of features.

Most marketing researchers treat neural networks as a black box. They leave the decision in model selection to computer software packages and rely on linear statistical models such as logistic regression for feature selection. Our study presents a rather comprehensive approach to neural network modeling. We believe model development including feature selection is critical in any modeling effort especially in neural network modeling because of its complexity.

Our cross-validation experimental results suggest that the feature selection approach based on the prediction risk idea is very robust. The variables selected correspond precisely to those identified by the all-possible-subset approach. Presumably the all-possible-subset procedure is the most comprehensive and reliable approach for feature selection. Therefore, we have provided credence to the effectiveness of our backward selection algorithm.

Practical managerial implications can be drawn from the results of this study. Across the three validation samples, a consistent pattern emerges. The four-variable model with features REASON, RECEIVER, NUMCALL, and NUMLET seems to be the most suitable model. These variables are directly related to a communication situation found in a weekly diary, and thus supporting previous findings in this area that communication situational variables are useful in predicting consumer choices [23,35]. Marketing efficiency is based primarily on how accurate the marketer can predict or forecast consumer behavior. As shown in this study, neural networks are capable of producing superior performance in terms of classification rates with fewer number of predictor variables. As efficiency increases, marketers will be able to generate more revenue at a lower cost.

References

[1] M. Adya, F. Collopy, How effective are neural networks at forecasting and prediction? A review and evaluation, Journal of Forecasting 17 (1998) 481–495.

[2] D. Agrawal, C. Schorling, Market share forecasting: an empirical comparison of artificial neural networks and multinomial logit model, Journal of Retailing 72 (1996) 383–407.

[3] B.-H. Ahn, Forward additive neural network methods, Ph.D. Dissertation, Kent State University, Kent, OH, USA, 1996.

[4] G.M. Allenby, J.L. Ginter, Using extremes to design products and segment markets, Journal of Marketing Research 32 (4) (1995) 392–403.

[5] B. Baesens, S. Viaene, D.V. den Poel, J. Vanthienen, G. Dedene, Bayesian neural network learning for repeat purchase modelling in direct marketing, European Journal of Operational Research 138 (2002) 191–211.

[6] Y. Bentz, D. Merunka, Neural networks and the multinomial logit for brand choice modeling: a hybrid approach, Journal of Forecasting 19 (2000) 177–200.

[7] J.R. Bettman, J.M. Jones, Formal models of consumer behavior: a conceptual overview, The Journal of Business 45 (4) (1972) 544–562.

[8] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.

[9] G. Castellano, A.M. Fanelli, Variable selection using neural network models Neurocomputing 30 (2000) 1–13.

[10] M. Cottrell, B. Girard, M. Mangeas, C. Muller, Neural modeling for time series: a statistical stepwise method for weight elimination, IEEE Transactions on Neural Networks 6 (1995) 1355–1364.

[11] C.G. Dasgupta, G.S. Dispensa, S. Ghose, Comparing the predictive performance of neural network model with some traditional market response models, International Journal of Forecasting 10 (1994) 235–244.

[12] M. Dash, H. Liu, Consistency-based search in feature selection, Artificial Intelligence 151 (2003) 155–176.

[13] V. Dhar, D. Chou, A comparison of nonlinear methods for predicting earnings surprises and returns, IEEE Transactions on Neural Networks 12 (4) (2001) 907–921.

[14] T.G. Dietterich, E.B. Kong, Machine Learning Bias, Statistical Bias, and Statistical Variance of Decision Tree Algorithms, Technical Report, Dept. of Computer Science, Oregon State University, 1995.

[15] J.H. Friedman, On bias, variance, 0/1-loss, and the curse of the dimensionality, Data Mining and Knowledge Discovery 1 (1997) 55–77.

[16] S. Geman, E. Bienenstock, T. Doursat, Neural networks and the bias/variance dilemma, Neural Computation 5 (1992) 1–58.

[17] L.W. Glorfeld, A methodology for simplification and interpretation of backpropagation-based neural networks models, Expert Systems with Applications 10 (1996) 37–54.

[18] J.V. Hansen, R.D. Nelson, Neural networks and traditional time series methods: a synergistic combination in state economic forecasts, IEEE Transactions on Neural Networks 8 (4) (1997) 863–873.

[19] B. Hassibi, D. Stork, Second order derivatives for network pruning: optimal brain surgeon, in: S. Hanson, J. Cown, C. Giles (Eds.), Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, CA, 1993, pp. 164–171.

[20] S.J. Hoch, B.D. Kim, A.L. Montgomery, P.E. Rossi, Determinants of store-level price elasticity, Journal of Marketing Research 32 (1) (1995) 17–29.

[21] M.Y. Hu, M.S. Hung, M. Shanker, H. Chen, Using neural networks to predict performance of Sino-foreign joint ventures, International Journal of Computational Intelligence and Organizations 1 (3) (1996) 134–143.

[22] M.Y. Hu, M. Shanker, M.S. Hung, Estimation of posterior probabilities of consumer situational choices with neural network classifiers, International Journal of Research in Marketing 6 (1999) 307–317.

[23] M.K. Hui, J.E.G. Bateson, Perceived control and the effects of crowding and consumer choice on the service experience, Journal of Consumer Research 18 (1991) 174–184.

[24] M.Y. Kiang, M.Y. Hu, D.M. Fisher, An extended self-organizing map network for market segmentation—a telecommunication example, Decision Support Systems 42 (2006) 36–47.

[25] Y. Kim, Toward a successful CRM: variable selection, sampling, and ensemble, Decision Support Systems 41 (2) (2006) 542–553.

[26] Y. Kim, W.N. Street, An intelligent system for customer targeting: a data mining approach, Decision Support Systems 37 (2) (2004) 215–228.

[27] M. Kudo, J. Sklansky, Comparison of algorithms that select features for pattern classifiers, Pattern Recognition 33 (2000) 25–41.

[28] K. Kvaal, H. Djupvik, Prediction of customer segments with neural nets, Marketing and Research Today (1996) 247–253.

[29] Y. Le Cun, J. Denker, S. Solla, Optimal brain damage, in: D. Touretzky (Ed.), Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, CA, 1990, pp. 598–605.

[30] E. Lee, M.Y. Hu, R.S. Toh, Are consumer survey results distorted? Systematic impact of behavioral frequency and duration on survey response errors, Journal of Marketing Research 37 (1) (2000) 125–134.

[31] D. McFadden, Conditional logit analysis of qualitative choice, in: P. Zarembka (Ed.), Frontiers in Econometrics, Academic Press, New York, 1973, pp. 105–142.

[32] K.E. Miller, J.L. Ginter, An investigation of situational variation in brand choice behavior and attitude, Journal of Marketing Research 16 (1) (1979) 111–123.

[33] J. Moody, J. Utans, Principled architecture selection for neural networks: application to corporate bond rating prediction, in: D. Touretzky (Ed.), Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, CA, 1992, pp. 683–690.

[34] R. Setiono, H. Liu, Neural-network feature selector, IEEE Transactions on Neural Networks 8 (3) (1997) 654–662.

[35] I. Simonson, R.S. Winer, The influence of purchase quantity and display format on consumer preference for variety, Journal of Consumer Research 19 (1992) 133–138.

[36] J.M. Steppe, K.W. Bauer, Improved feature screening in feedforward neural networks, Neurocomputing 13 (1996) 47–58.

[37] J.M. Steppe, K.W. Bauer, Feature saliency measures, Computers and Mathematical Applications 33 (1997) 109–126.

[38] J.M. Steppe, K.W. Bauer, S.K. Rogers, Integrated feature and architecture selection, IEEE Transactions on Neural Networks 7 (4) (1996) 1007–1014.

[39] M.C. van Wezel, W.R.J. Baets, Predicting market responses with a neural network: the case of fast moving consumer goods, Marketing Intelligence and Planning 13 (7) (1995) 23–30.

[40] A. Vellido, P.J.G. Lisboa, K. Meehan, Segmentation of the online shopping market using neural networks, Expert Systems with Applications 17 (1999) 303–314.

[41] P.M. West, P.L. Brockett, L.L. Golden, A comparative analysis of neural networks and statistical methods for predicting consumer choice, Marketing Science 16 (1997) 370–391.

Michael Hu has a Ph.D. from the University of Minnesota in management science/marketing. Currently he holds the Bridgestone Chair in International Business and is a professor of Marketing at Kent State University. He has published over a hundred and twenty academic articles in the areas of applications of artificial neural networks, international business, and marketing. His research has appeared in Decision Support Systems, Journal of Marketing Research, Marketing Letters, Annals of Operations Research, Decision Sciences, European Journal of Operational Research and among many others. He won the University Distinguished Teaching Award in 1994 and the University Distinguished Scholar Award in 2006.

Murali S. Shanker is an Associate Professor in the department of Management & Information Systems, College of Business, Kent State University. He received his Ph.D. from the Department of Operations and Management Science at the University of Minnesota. His current research is in open source systems, agent-based simulations, developing task-allocation strategies for distributed simulation models, and in artificial neural networks as applied to classification and prediction problems. He has published in journals like Annals of Operations Research, Decision Sciences, Journal of the Operational Research Society, INFORMS Journal on Computing, and IIE Transactions, among others.

G. Peter Zhang is an Associate Professor of Managerial Sciences at Georgia State University. His research interests include neural networks, forecasting, and supply chain management. He currently serves as an associate editor of Neurocomputing and Forecasting Letters and is on the editorial review board of Production and Operations Management and International Journal of E-Business Research. He is the Editor of the book: Neural Networks in Business Forecasting (Hershey, PA: IRM Press, 2004). His research has appeared in Decision Sciences, European Journal of Operational Research, IIE Transactions, IEEE Transactions on Neural Networks, IEEE Transactions on SMC, International Journal of Forecasting, Journal of the Operational Research Society, Neurocomputing, and others.

Ming S. Hung is Professor Emeritus of Operations Research at Kent State University. His main areas of interests are neural networks and mathematical programming. His writings have appeared in Operations Research, Management Science, and European Journal of Operational Research, among others.