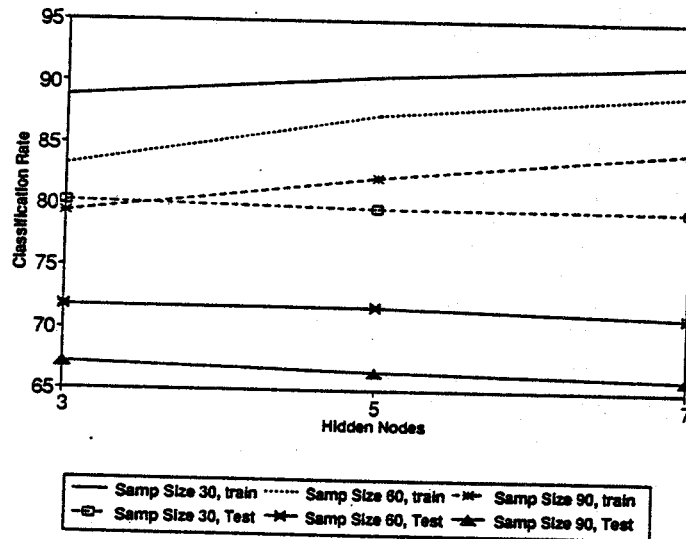


Figure 5: Neural network classification rates by hidden nodes.



While the network can pick up the idiosyncrasies in the training set it may result in a separation curve far from the optimal curve.

Table 3 summarizes the effect of architecture and sample size by problem type. It is seen clearly that for training, both factors are significant and there is no interaction between them. However, for the test set, only sample size is significant in all three problem types. As we have seen previously, as sample size increases, the classification rate in the test set increases also. The effect of architecture on well-behaved problems like P1 and P2 is minimal. It suggests the desirability of having a small network and also the possibility that a network with fewer than three hidden nodes may work as well.

The picture emerging from this experiment is:

1. If the objective is to achieve maximum classification rate in a training sample, use large networks.
2. If the objective is for generalizing the network results back to the population, use small networks with large training samples.

We admit that our explanations are speculative. But there is no available theory in neural networks to offer a satisfactory answer to the phenomena we have observed here. We would also add that our observations, though consistent with our conjectures, are limited to the experimental factors considered.

EXPERIMENT 2

Design

The second experiment is designed to allow direct comparison between the classical discriminant procedures and neural networks. As indicated in the findings of the

Table 2: Average classification rates by factors.

Factor	Levels	Training Set (%)	Test Set (%)	Pairwise Comparisons*	
				Training	Test
Hidden Nodes	H1	83.82	73.15		
	H2	86.66	72.61	(1,2), (1,3), (2,3)	(1,3)
	H3	88.43	72.01		
Problem Type	P1	90.26	79.84		
	P2	86.59	71.31	(1,2), (1,3), (2,3)	(1,2), (1,3), (2,3)
	P3	82.05	66.62		
Sample Size	S1	88.80	70.62		
	S2	85.99	72.81	(1,2), (1,3), (2,3)	(1,2), (1,3), (2,3)
	S3	84.11	74.34		

*Significant differences detected by Tukey's pairwise comparison.

first experiment, the neural architecture with three hidden nodes provides consistently the highest correct classification rate of the test sets and in addition the number of hidden nodes does not interact with the other two experimental factors. Thus, this architecture is to be used in the second experiment. Since both the problem type (P1, P2, P3) and the size of training samples (S1, S2, S3) have significant effects on the correct classification rates of neural networks, these same factors are used in the second 3x3 experiment.

SAS procedure DISCRIM [30] is used to solve all three types of problems. The linear discriminant (LDA) classifier is chosen for P1 as it is the optimal method. The option for not pooling the variance-covariance matrices is used as the quadratic discriminant (QDA) classifier for P2. For P3, both QDA and the NPAR (nonparametric) option with $k=3$ in DISCRIM are used for classification. In addition, the LP model MSD [9] is used. The reason for using so many models for P3 is that there is no known optimal method for this problem and only LDA can be excluded as it is subsumed under QDA.

Results

The results on the training samples are summarized in Tables 4 and 5. As shown in Table 4, the overall classification rate by LDA for problem P1 is 85.20 percent, by QDA for P2 is 79.17 percent, and by nonparametric for P3 is 80.73 percent. Each of these averages was based on 90 random samples from each distribution. These values should be fairly close to the expected rate. This is the reason we postulated the optimal rates to be 85 percent, 79 percent, and 80 percent for problem types P1, P2, and P3, respectively.

Table 4 shows that neural network models are better than LDA for P1 and better than QDA for P2. For P3, neural networks are better than QDA and LP but worse than k -nearest neighbor. The classification rate resulted when a competitive method is paired with that from neural networks. The differences are then tested for significant difference from zero. Table 4 shows that all pairwise comparisons

Table 3: ANOVA results by problem type.

Problem Type	Factor	Training		Test	
		F	p-value	F	p-value
P1	Sample	15.02	.0001	16.45	.0001
	Nodes	7.10	.0010	.65	.5233
	Sample × Nodes	.13	.9706	.30	.8753
P2	Sample	9.19	.0001	23.19	.0001
	Nodes	29.36	.0001	1.02	.3605
	Sample × Nodes	.89	.4674	.12	.9766
P3	Sample	30.47	.0001	17.38	.0001
	Nodes	17.73	.0001	4.95	.0078
	Sample × Nodes	1.28	.2781	.23	.9218

are significant at .05 level except for the difference between *k*-nearest-neighbor and neural networks for P3 and sample sizes 30 (S1) and 60 (S2). However, the advantage of neural networks over the best competitor in each problem type drops when sample size increases.

The analysis of variance results in Table 5 are grouped by problem type and then by procedure. The only factor used is sample size. The results show that for the procedures in SAS DISCRIM - LDA, QDA, and *k*-nearest-neighbor, there is no difference in the classification rate as sample size changes. For linear programming, sample size makes a difference. For neural networks, sample size is a significant factor in both P1 and P3.

Tables 6 and 7 show the results of classification rates using the test samples. Table 6 shows that LDA is better than neural networks for P1, QDA is better for P2, and QDA and LP are better for P3. The analyses of paired differences show that all differences are significantly different from zero except for two occasions in P3: QDA and neural networks in S3, and *k*-nearest-neighbor and neural networks in S1. In general, as sample size increases, the performance of neural networks improves. Indeed, when sample size is 90, neural networks outperformed all other methods for P3. Again, a pattern emerges. The procedures which are better at classifying training sets are not as good at classifying test sets. If we can use a loose term power to denote the ability of a classifier to adapt itself to data, then it is reasonable to say that the nonparametric procedure in SAS DISCRIM is more powerful than neural networks of a fixed architecture, which in turn is more powerful than QDA and LP. (The difference between QDA and LP for P3 are hard to explain.) As the examples show, power comes from the generality of the procedure. A more powerful method is better able to classify a given data set. On the other hand, it is less able to make inferences about populations.

Sample size is a significant factor for LDA in P1, QDA in P2, and neural networks in all three problem types. However, it is not significant for the other methods for P3, as shown in Table 7. The results for LDA and QDA are expected from the Bayesian classification theory. It is seen that as sample size increases, the classification rate converges to our speculated expected rate for either method. The

Table 4: Classification rates of training samples.

	Method	S1 (%)	S2 (%)	S3 (%)	Total (%)
P1	LDA	85.33	84.89	85.37	85.20
	Neural	91.22	88.39	86.85	88.82
P2	QDA	79.22	79.28	79.00	79.17
	Neural	84.22	82.95	82.67	83.28
P3	QDA	72.56	70.78	69.59	70.98
	<i>k</i> -near	80.56*	80.78*	80.85	80.73
	LP	73.56	70.44	69.11	71.04
	Neural	81.89	79.33	76.85	79.36

*Not significantly different from neural at .05 level.

Table 5: ANOVA results for training samples (factor: sample size).

Problem Type	Method	<i>F</i> -value	<i>p</i> -value	Tukey's Comparison
P1	LDA	.08	.9196	
	Neural	5.88	.0040	(1,3)
P2	QDA	.02	.9811	
	Neural	.48	.6226	
P3	QDA	2.41	.0961	
	<i>k</i> -near	.03	.9709	
	LP	4.06	.0207	(1,3)
	Neural	6.08	.0034	(1,3)

overall picture is that for generalizability neural networks should have large samples. This confirms the conclusion from Experiment 1 which is shown in Table 3.

CONCLUSIONS AND IMPLICATIONS

Neural networks have emerged as an important tool for classification. Although there have been many successful applications, few general guidelines are available for building neural models. Our study has shed some light on two key issues—network architecture and size of training samples. If the objective is to classify a given set of objects, then one would want to use a large network. But if the objective is to predict the classification of objects from an unseen population, a theoretical foundation for explaining such behaviors is still lacking.

Neural networks classifiers are better than the traditional methods and the LP model MSD on training samples and slightly worse on the test samples. However, as sample size increases, neural networks improve their performance on the test samples. These statements leave the impression that neural networks may be a poor choice for a real world classification problem. We do not suggest this. Besides the

Table 6: Classification rates of test samples.

	Method	S1 (%)	S2 (%)	S3 (%)	Total (%)
P1	LDA	83.26	84.28	84.12	83.69
	Neural	78.33	80.88	81.17	80.13
P2	QDA	78.39	79.65	80.49	79.51
	Neural	68.92	71.78	74.49	71.73
P3	QDA	68.13	69.38	68.76*	68.76
	<i>k</i> -near	64.69*	65.14	65.03	64.96
	LP	68.40	69.14	68.62	68.72
	Neural	65.93	67.18	69.71	67.61

*Not significantly different from neural at .05 level.

Table 7: ANOVA results for test samples (factor: sample size).

Problem Type	Method	<i>F</i> -value	<i>p</i> -value	Tukey's Comparison
P1	LDA	5.38	.0062	(1,2), (1,3)
	Neural	6.87	.0017	(1,2), (1,3)
P2	QDA	6.53	.0023	(1,3)
	Neural	6.98	.0015	(1,3)
P3	QDA	1.55	.2175	
	<i>k</i> -near	.11	.8911	
	LP	.53	.5896	
	Neural	6.90	.0017	(1,3), (2,3)

difficult and non-quantitative problems such as recognition of finger prints [21] and speech patterns [31], neural networks have been successful with real statistical classification problems such as bond rating [33] and prediction of bank failures [38]. The reason is that real data sets are rarely, if ever, clean in the sense that the variables are drawn from known distributions with known covariance matrices. For such data sets, neural networks are more attractive than the classical methods in two aspects: generality and flexibility. Neural networks are applicable to a much broader class of problems than any of the classical methods and for dirty data sets, neural networks have performed better than the classical methods [38]. In addition, changing a neural network model is easily affected by changing the architecture. So our view is that for real world classification tasks, neural networks should be considered.

Future research will include new factors such as different group proportions in the training samples and attributes with categorical values, and extensions of the current factors by increasing the number of attributes and groups, and increasing the levels of correlation between variables. In another direction, research into the distributions of output values and arc weights will provide better guidelines for

neural network model building (e.g., choice of architecture) and output analysis (e.g., estimation of population parameters). [Received: August 10, 1992. Accepted: May 22, 1993.]

REFERENCES

- [1] Anderson, T. W. *An introduction to multivariate statistical analysis* (2nd Edition). New York: John Wiley, 1984.
- [2] Awh, R. Y., & Waters, D. A discriminant analysis of economic, demographic, and attitudinal characteristics of bank charge-card holders: A case study. *Journal of Finance*, 1974, 29, 973-980.
- [3] Bajgier, S. M., & Hill, A. V. An experimental comparison of statistical and linear programming approaches to the discriminant problem. *Decision Sciences*, 1982, 13, 604-618.
- [4] Capon, N. Credit scoring systems: A critical analysis. *Journal of Marketing*, 1982, 46, 82-91.
- [5] *DARPA neural network study*. Lincoln Laboratory, Massachusetts Institute of Technology, Boston, 1988.
- [6] Denton, J. W., Hung, M. S., & Osyk, B. A. A neural network approach to the classification problem. *Expert Systems with Applications*, 1990, 1, 417-424.
- [7] Duda, R. O., & Hart, P. E. *Pattern classification and scene analysis*. New York: Wiley & Sons, 1973.
- [8] Fisher, R. A. The statistical utilization of multiple measurements. *Annals of Eugenics*, 1938, 8, 376-386.
- [9] Freed, N., & Glover, F. Evaluating alternative linear programming formulations for the two-group discriminant problem. *Decision Sciences*, 1986, 17, 151-162.
- [10] Freed, N., & Glover, F. Linear programming approach to the discriminant problem. *Decision Sciences*, 1981, 12, 68-74.
- [11] Gehrlein, W. V. General mathematical programming formulations for the statistical classification problem. *Operations Research Letters*, 1986, 5, 299-304.
- [12] Gordon, A. D. *Classification*. London: Chapman and Hall, 1981.
- [13] Gorman, R. P., & Sejnowski, T. J. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1988, 1, 75-89.
- [14] Hoskins, J. C., Kaliyur, K. M., & Himmelblau, D. M. Incipient fault detection and diagnosis using artificial neural networks. *Proceedings of the International Joint Conference on Neural Networks*, 1990, Vol. I, 81-86.
- [15] Huang, W. Y., & Lippmann, R. P. Comparisons between neural net and conventional classifiers. *IEEE 1st International Conference on Neural Networks*. San Diego, CA: June 1987, Vol. IV, 485-493.
- [16] Hung, M. S., & Denton, J. W. Training neural networks using a nonlinear programming approach to back propagation. *European Journal of Operational Research*, forthcoming.
- [17] Lang, K. J., & Witbrock, M. J. Learning to tell two spirals apart. *Proceedings of the 1988 Connectionist Models Summer School*, 1988, 52-59.
- [18] Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. Handwritten digit recognition with a back-propagation network. *Advances in Neural Information Processing Systems*, 1990, 2, 396-404.
- [19] Lasdon, L. S., & Waren, A. D. *GRG2 user's guide*. School of Business Administration, University of Texas at Austin, TX, 1986.
- [20] Law, A. V., & Kelton, W. D. *Simulation modeling & analysis* (2nd Edition). New York: McGraw-Hill, 1991.
- [21] Leung, M. T., Engeler, W. E., & Frank, P. Fingerprint processing using backpropagation neural networks. *Proceedings of the International Joint Conference on Neural Networks I*, 1990, 15-20.
- [22] Lippmann, R. An introduction to computing with neural nets. *IEEE ASSP Magazine*, 1987, 4, 2-22.
- [23] Malkoff, D. B. A neural network for real-time signal processing. *Advances in Neural Information Processing Systems*, 1990, 2, 248-257.
- [24] Markowski, E. P., & Markowski, C. A. Some difficulties and improvements in applying linear programming formulations to the discriminant problem. *Decision Sciences*, 1985, 16, 237-247.
- [25] Martin, G. L., & Pittman, J. A. Recognizing hand-printed letters and digits. *Advances in Neural Information Processing Systems*, 1990, 2, 405-414.

- [26] McCulloch, W. S., & Pitts, W. A logical of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 1943, 5, 115-133.
- [27] Pao, Y-H. *Adaptive pattern recognition and neural net implementation*. Reading, MA: Addison-Wesley, 1989.
- [28] Rubin, A. P. A comparison of linear programming and parametric approaches to the two-group discriminant problem. *Decision Sciences*, 1990, 21, 373-386.
- [29] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. Learning internal representations by error propagation. In D. E. Rumelhart & J. L. Williams (Eds), *Parallel distributed processing: Explorations in the microstructure of cognition*. Cambridge, MA: MIT Press, 1986.
- [30] *SAS user's guide: Statistics*, Ver. 5. NC: SAS Institute, 1991.
- [31] Sejnowski, T. J., Yuhas, B. P., Goldstein, M. H., Jr., & Jenkins, R. E. Combining visual and acoustic speech signals with a neural network improves intelligibility. *Advances in Neural Information Processing Systems*, 1990, 2, 232-239.
- [32] Shea, P. M., & Liu, F. Operational experience with a neural network in the detection of explosives in checked airline baggage. *Proceedings of the International Joint Conference on Neural Networks*, 1990, Vol. II, 175-178.
- [33] Singleton, J. C. Neural nets for bond rating improved by multiple hidden layers. *Proceedings of the International Joint Conference on Neural Networks*, 1990, Vol. II, 151-162.
- [34] Smith, C. A. B. Some examples of discrimination. *Annals of Eugenics*, 1946, 13, 272-282.
- [35] Stam, A., & Joachimsthaler, E. A. Solving the classification problem in discriminant analysis via linear and nonlinear programming methods. *Decision Sciences*, 1989, 20, 285-293.
- [36] Subramanian, V., & Hung, M. S. A GRG2-based system for training neural networks: Design and computational experience. *ORSA Journal on Computing*, forthcoming.
- [37] Subramanian, V. Hung, M. S., & Hu, M. Y. An experimental evaluation of neural networks for classification. *Computers and Operations Research*, 1993, 20, 769-782.
- [38] Tam, K. Y., & Kiang, M. Y. Managerial applications of neural networks: The case of bank failure predictions. *Management Science*, 1992, 38, 926-947.
- [39] Walking, R. A. Predicting tender offer success: A logistic analysis. *Journal of Finance and Quantitative Analysis*, 1985, 20, 461-478.
- [40] Wasserman, P. D. *Neural computing—Theory and practice*. New York: Van Nostrand Reinhold, 1989.

Eddy Patuwo is Assistant Professor in the Department of Administrative Sciences at Kent State University. He earned his Ph.D. in IEOR from Virginia Polytechnic Institute and State University. His research interests are in the study of stochastic systems (queueing, inventory, manufacturing) and neural networks.

Michael Y. Hu earned his Ph.D. in management science from the University of Minnesota in 1977 and is currently Associate Professor of Marketing at Kent State University. His research interests include applications of neural networks, applied statistics, marketing research, and international business.

Ming S. Hung is Professor of Operations Research in the Graduate School of Management at Kent State University. His primary research interests include networks and other mathematical programming, neural networks, and communication networks. His publications have appeared in *Operations Research*, *Management Science*, and *Naval Research Logistic Quarterly*.