Rough Neural Networks

Pawan Lingras

Department of Computer Science, Algoma University College
Sault Ste. Marie, Ontario, Canada, P6A 2G4.
e-mail: lingras@tbird.auc.on.ca

Abstract

This paper describes rough neural networks which consists of a combination of rough neurons and conventional neurons. Rough neurons use pairs of upper and lower bounds as values for input and output. In some practical situations, it is preferable to develop prediction models that use ranges as values for input and/or output variables. A need to provide tolerance ranges is an example of such a situation. Inability to record precise values of the variables is another situation where ranges of values must be used. In the example used in this study, a number of input values are associated with a single value of the output variable. Hence, it seems appropriate to represent the input values as ranges. The predictions obtained using rough neural networks are significantly better than the conventional neural network model.

1. Introduction

The concept of upper and lower bound has been used in a variety of applications in artificial intelligence (Shafer 1976; Pawlak 1982). In particular, theory of rough sets (Pawlak, 1992, 1984) has demonstrated the usefulness of upper and lower bounds in rule generation. Further developments in rough set theory (Polkowski, 1994; Wong, 1994; Yao, et al, 1994), have shown that the general concept of upper and lower bounds provide a wider framework that may be useful for different types of applications. This paper uses rough patterns for predictions using neural networks. Each value in a rough pattern is a pair of upper and lower bound. Conventional neural network models generally use a precise input pattern in their estimations. The conventional neural network models need to be modified to accommodate rough patterns. Rough neurons proposed in this paper provide an ability to use rough patterns. Each rough neuron stores the upper and lower bounds of the input and output values. Depending upon the nature of the application, two rough neurons in the network can be connected to each other using either two or four connections. A rough neuron can also be connected to a conventional neuron using two connections. A rough neural network consists of a combination of rough and conventional neurons connected each other. The paper outlines procedures for feedforward and backpropagation in a rough neural network.

The paper also compares two different rough neural network models with a conventional neural network model for prediction of the design hourly traffic volume (DHV) for a highway section. The prediction is based on traffic volumes recorded over a short period of time. The input to the network consists of traffic volumes for each day of the week, i.e. Sunday, Monday, Tuesday, ..., Saturday, over the given time period. There are several Mondays in the data collection period. Hence, the traffic volume for a Monday cannot be a single value but must be a set of values. The conventional neural network alternative uses the average of all the values for each Monday. Similar argument applies to the rest of the days of the week. The use of average values tends to ignore some of the available information. The rough neural network models use rough input pattern consisting of upper and lower bounds of daily traffic volumes.

2. Overview

This section briefly reviews some of the essential concepts of neural networks. A brief description of highway data collection and analysis program is also provided.

2.1 Conventional Neural Networks

Neural networks are good at recognizing patterns, generalizing, and predicting trends (White, 1989). Researchers have proposed different types of neural networks for solving a variety of problems (Hecht-Nielsen, 1990). In its most general form, a neural network consists of several layers of neurons. Each neuron receives input from other neurons and external environment and produces output.

The conventional neural network used in this study is based on multi-layered, feed-forward, and backpropagation design for supervised learning. This network consists of one input layer, one output layer and one hidden layer of neurons. The input layer neurons accept input from the external environment. The output from input layer neurons is fed to the hidden layer neurons. The hidden layer neurons feed their output to the output layer neurons which send their output to the external environment. Neurons from each layer feed the output only to the next layer and hence the network is called *feed forward*. The input and output of a neuron are governed by certain mathematical equations. Output from a neuron is calculated using a transfer function.

Two stages in the development of the neural network model are training and testing. During the training stage, the network uses an inductive learning principle to learn from a set of examples called the training set. The learning process used in this study is called *supervised learning*. In supervised learning, the desired output is known for output layer neurons for the examples in the training set. The network attempts to adjust weights of connections between neurons to produce the desired output. During this process, the error in the output is propagated back from one layer to the previous layer by adjusting weights of the connections. This is called the *backpropagation* method for propagating the error. This study uses one of the most popular learning equations called the generalized delta rule for supervised learning.

In the testing stage, the network is tested for another set of examples for which the output from the output layer neurons is known. After the neural net model is tested successfully, it is used for predictions.

2.2 Estimation of Traffic Parameters

Highway agencies collect traffic volume data from various seasonal and permanent traffic counters over a number of years (Garber and Hoel, 1988). Since the installation of a permanent traffic counter (PTC) on every road section is not economically feasible, highway agencies routinely use sample traffic counts (DeGarmo and Sullivan, 1985). The sample traffic counts are obtained using seasonal traffic counters (STCs). The data obtained from seasonal traffic counts is used to estimate important traffic parameters for overall highway network (Sharma and Allipuram, 1993). The present study deals with the estimation of an important traffic parameter called the *design hourly volume*. A highway section is designed to service a certain amount of traffic volume with a reasonable level of service. The volume that is used in such a design is called the Design Hourly Volume (DHV). Different agencies use different criteria for determining the DHV. However, most agencies use the highest hourly volumes in the calculation of the DHV. The design hourly volume is calculated by sorting all the hourly traffic volumes in a given year to identify the highest volume hours that are likely to experience traffic congestion. The 30th highest hourly volume is one of the most commonly used design hourly volumes.

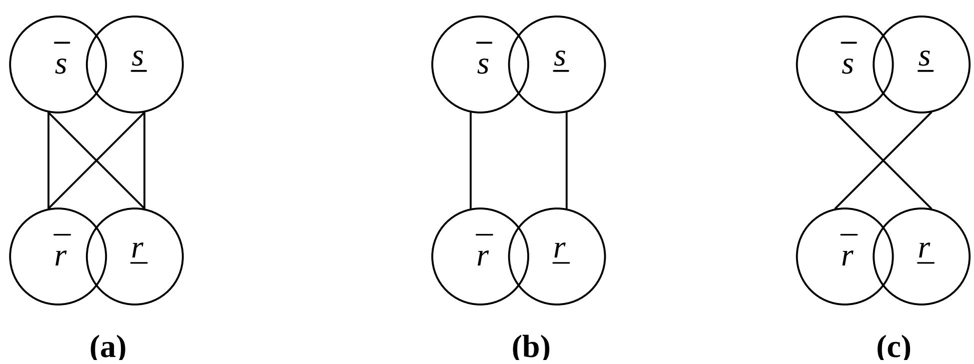

Fig.1 Three Different Types of Connections Between Two Rough Neurons

3. Rough Neural Network

A rough neural network consists of conventional neurons and rough neurons connected to each other. A rough neuron *r* in rough neural networks can be viewed as a pair of neurons, one for the upper bound called $\overline{r}$ and the other for the lower bound called $\underline{r}$. A rough neuron is connected to another rough neuron through two or four connections. Fig. 1 depicts three types of connections between rough neurons. The overlap between the upper and lower neurons indicates that upper and lower neurons exchange information. Two rough neurons in Fig. 1(a) are *fully connected*. A rough neuron *r* is said to be *fully connected* to another rough neuron *s*, if $\overline{r}$ and $\underline{r}$ are connected to both $\overline{s}$ and $\underline{s}$. If a rough neuron *r* is fully connected to *s*, then there are four connections from *r* to *s*. In Fig. 1(b) and 1(c), there only two connections from *r* to *s*. If the rough neuron *r* *excites* the activity of *s* (i.e. increase in the output of *r* will result in the increase in the output of *s*), then *r* will be connected to *s* as shown in Fig. 1(b). On the other hand, if *r* *inhibits* the activity of *s* (i.e. increase in the output of *r* corresponds to the decrease in the output of *s*), then *r* will be connected to *s* as shown in Fig. 1(c).

This paper uses multi-layered, feed-forward, and backpropagation design outlined in section 2.1 to describe the methodology of rough neural networks. Rough neural networks used in this study consist of one input layer, one output layer and one hidden layer of rough/conventional neurons. The input layer neurons accept input from the external environment. The outputs from input layer neurons are fed to the hidden layer neurons. The hidden layer neurons feed their output to the output layer neurons which send their output to the external environment. The output of a rough neuron is a pair of upper and lower bounds, while the output of a conventional neuron is a single value.

The input of a conventional, lower, or upper neuron is calculated using the weighted sum as:

$$input_i = \sum_{\text{there is a connection from } j \text{ to } i} w_{ji} \times output_j \,, \tag{1}$$

where i and j are either the conventional neurons or upper/lower neurons of a rough neuron. The outputs of a rough neuron r is calculated using a transfer function as:

$$output_{\overline{r}} = \max\left(transfer(input_{\overline{r}}), transfer(input_{\underline{r}})\right), \tag{2}$$

$$output_{\underline{r}} = \min\left(transfer(input_{\overline{r}}), transfer(input_{\underline{r}})\right). \tag{3}$$

The output of a conventional neuron i is simply calculated as

$$output_i = transfer(input_i). \tag{4}$$

This study uses the sigmoid transfer function given as:

$$transfer(u) = \frac{1}{1+e^{-gain \times u}}, \tag{5}$$

where *gain* is a system parameter determined by the system designer to specify the slope of the sigmoid function around input value of zero. There are several other functions for determining the output from a neuron. The sigmoid transfer function is chosen because it produces a continuous value in the 0 to 1 range.

If two rough neurons are partially connected, then the excitatory or inhibitory nature of the connection is determined dynamically by polling the connection weights. The network designer can make initial assumptions about the excitatory or inhibitory nature of the connections. If a partial connection from a rough neuron r to another rough neuron s is assumed to be excitatory and $w_{\overline{r}\overline{s}} < 0$ and $w_{\underline{r}\underline{s}} < 0$, then the connection from rough neuron r and s is changed from excitatory to inhibitory by assigning $w_{\underline{r}\overline{s}} = w_{\underline{r}\underline{s}}$ and $w_{\overline{r}\underline{s}} = w_{\overline{r}\overline{s}}$. The links $(\underline{r}, \underline{s})$ and $(\overline{r}, \overline{s})$ are disabled while links $(\underline{r}, \overline{s})$ and $(\overline{r}, \underline{s})$ are enabled. On the other hand, if the neuron r is assumed to have an inhibitory partial connection to s and $w_{\overline{r}\underline{s}} > 0$ and $w_{\underline{r}\overline{s}} > 0$, then the connection between rough neuron r and

s is changed from inihibitory to excitatory by assigning $w_{\underline{r}\underline{s}} = w_{\underline{r}\bar{s}}$ and $= w_{\bar{r}\bar{s}} = w_{\bar{r}\underline{s}}$. The links $(\underline{r}, \bar{s})$ and $(\bar{r}, \underline{s})$ are disabled while links $(\underline{r}, \underline{s})$ and $(\bar{r}, \bar{s})$ are enabled.

The training and testing stage in the development of a rough neural networks is similar to the conventional neural network discussed in section 2.1. During the training stage the network uses inductive learning principle to learn from a set of examples called the training set. In supervised learning, the desired output from output layer neurons for the examples in the training set is known. The network attempts to adjust weights of connections between neurons to produce the desired output. During this process, the error in the output is propagated back from one layer to the previous layer for adjusting weights of the connections.

The weights of the connections are modified iteratively. The network is presented with the training set repeatedly and is allowed to change weights after one (or more) iteration(s). The weights are modified using a learning equation. This study uses the generalized delta rule for modifying the weights of the connections using the following equation:

$$w_{ji}^{new} = w_{ji}^{old} + \alpha \times output_j \times error_i \times transfer'(input_i) \quad (6)$$

where $transfer'(input_i)$ is the derivative of the transfer function evaluated at $input_i$ and α is the learning parameter which represents the speed of learning. For the sigmoid transfer function used in this study,

$$transfer'(input_i) = input_i \times (1 - input_i) \quad (7)$$

The error in eq. (6) is calculated as:

$$error_i = desired\_output_i - output_i. \quad (8)$$

As mentioned before, in the testing stage, the network is tested for another set of examples for which the output from the output layer neurons is known. After the neural net model is tested successfully, it is used for predictions.

4. Conventional and Rough Neural Network Models for Traffic Estimations

One of the objective of this study is to demonstrate the usefulness of rough neural networks over the conventional neural networks. The neural networks have shown to be more effective than the existing methods for estimation of traffic parameters such as the Design Hourly Volume (DHV) for a highway section (Lingras and Adamo, 1995). This section outlines the nature of the experiment and the rough and conventional neural network models used in the study to predict the DHV.

Traffic volume data used in the study consisted of five year traffic volumes collected at various permanent traffic counter (PTC) sites in the province of Alberta, Canada. The PTC sites collect data for every hour in a given year. The annual hourly patterns were divided into training and test sets. The hourly volumes for the entire year were sorted and the thirtieth highest hourly volume was used as the DHV. For all the objects in training and test sets, the DHV is known. The objective of the experiment is to estimate the DHV based on daily volume patterns over a week in the months of July and December. Months of July and December were chosen because these two months generally have significantly different travel patterns.

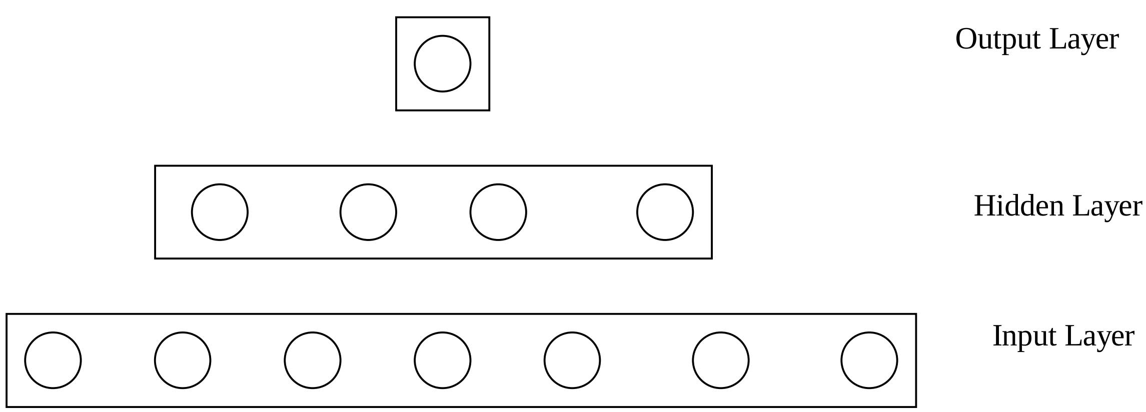

Fig. 2. The conventional Neural Network Model Used in the Estimation of DHV

Fig. 2 shows the conventional neural network model used for the estimation. The conventional model has seven input neurons, four hidden layer neurons and one output neuron. Neurons in the input layer are fully connected to neurons in the hidden layer. Neurons in the hidden layer are fully connected to the neuron in the output layer. The input to the conventional neural network model consists of average weekly pattern, i.e. average daily volumes on Sundays, Mondays, Tuesdays, ..., Saturdays for an object. The output is the DHV for the object.

The first rough neural network model shown in Fig. 3 has seven rough input neurons, and eight hidden layer conventional neurons and one output neuron. Rough neurons in the input layer are fully connected to conventional neurons in the hidden layer. Conventional neurons in the hidden layer are fully connected to the conventional neuron in the output layer. Since the hidden and output layer neurons are conventional neurons, this network can be easily implemented using existing neural network packages such as Stuttgart Neural Network Simulator (SNNS) (Zell, *et al.*).

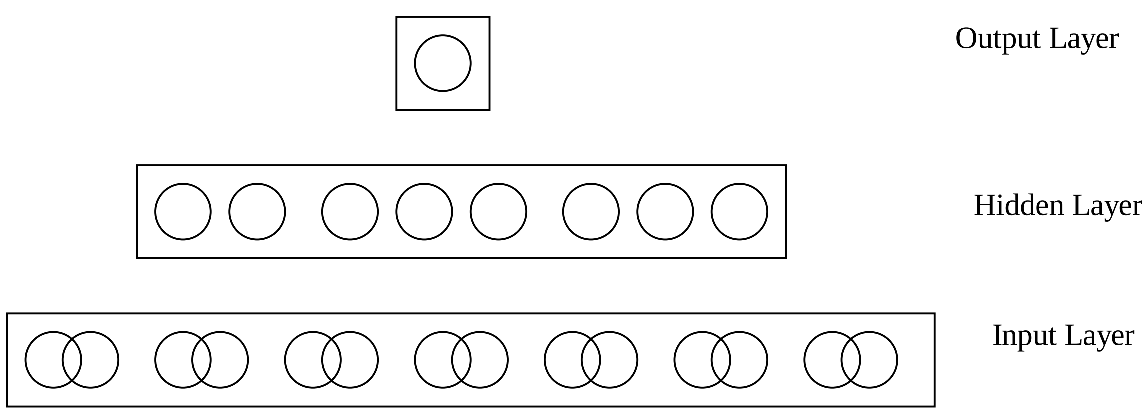

Fig. 3. The Rough Neural Network Model Hidden Layer Conventional Neurons

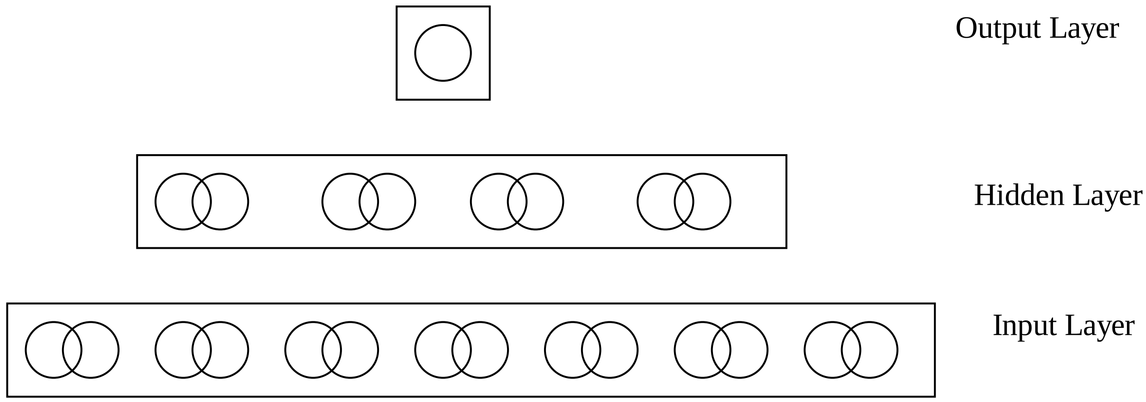

Fig. 4. The Rough Neural Network Model With Hidden Layer Rough Neurons

The second rough neural network model shown in Fig. 4 has seven rough input neurons, and four hidden layer rough neurons and one output neuron. Rough neurons in the input layer are fully connected to rough neurons in the hidden layer. Rough neurons in the hidden layer are fully connected to the conventional neuron in the output layer. The rough network shown in Fig. 4 can also be implemented using SNNS. However it was necessary to add two activation functions to implement eq. (2) and eq. (3).

The input to both the rough neural network models consists of rough weekly pattern, i.e. upper and lower bounds of daily volumes on Sundays, Mondays, Tuesdays, ..., Saturdays for an object. The output is the DHV for the object. Since the output is a unique value, the output layer for both the models used a conventional neuron.

Both the networks were trained using 211 objects from the training set and tested using 53 objects in the test set. Errors in estimation for the test set may originate from two sources. One of the sources of errors is the sampling process. The number of patterns in training and test sets might be very small, or the samples may not provide a good representation of the universe. The other source of error is the estimation method itself. In order to get an indication of the errors from these two different sources, the conventional and rough neural networks were tested for training set as well as the test set. Testing the models using the training set indicates how well the training method works by itself.

The values of estimated and actual values of DHV are compared using the following percent difference measure.

$$\Delta = \frac{|\text{estimated} - \text{actual}|}{\text{actual}} \times 100$$

where

Δ = percent error

actual = actual DHV

estimated = estimated DHV

The maximum and average errors for each set are used to compare the results of estimation. The average error provides a measure of the overall accuracy, while the maximum error describes the worst case.

5. Result and Analysis

Table 1 shows the errors for the three models. Rough neural networks clearly outperform the conventional neural networks. The reduction in errors from the conventional network to the rough neural networks for training set is almost 50% for average errors, while 75% for maximum errors. The reduction in errors for test set is 25% for average errors and 10% for maximum errors. However, sampling errors may explain somewhat lower performance gain.

Model	Train Set		Test Set	
	Maximum	Average	Maximum	Average
Conventional	46.2%	9.6%	28.1%	9.7%
Rough 1	17.5%	5.5%	24.9%	8.1%
Rough 2	13.7%	5.8%	23.0%	8.0%

Table 1. Percetage Errors for Conventional and Rough Set Models

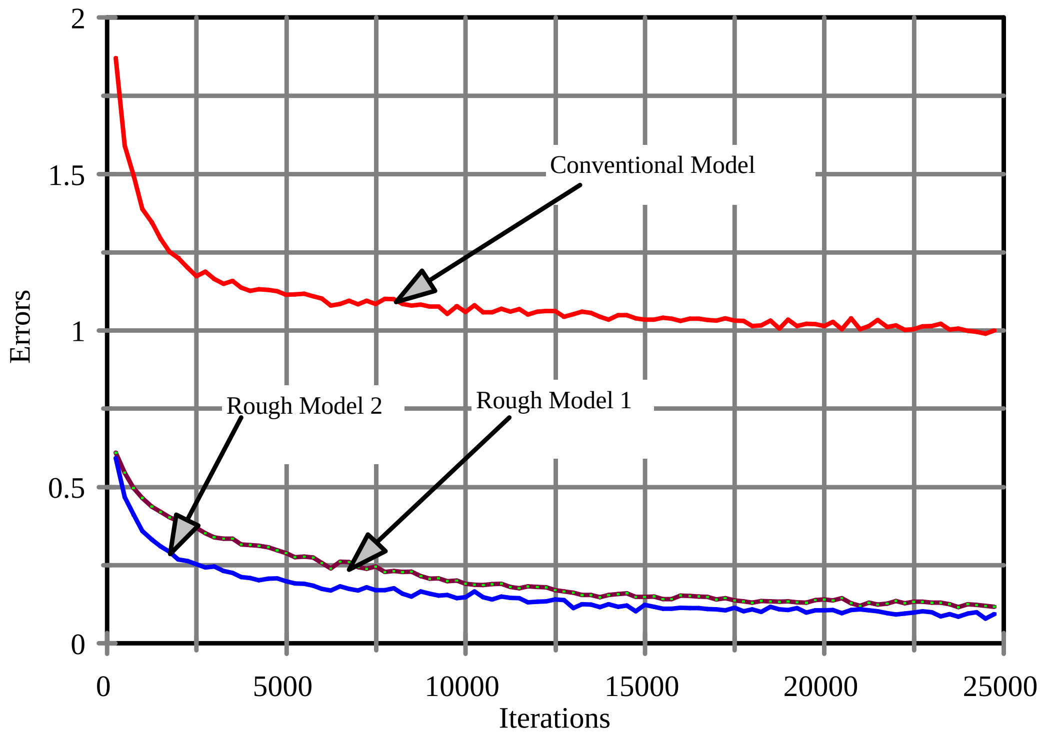

Fig. 5 Sum of Squares of Errors for Conventional and Rough Neural Networks

The first rough neural network with conventional neurons in the hidden layer results in somewhat higher errors than the second rough neural network which uses rough neurons in the hidden layer. This indicates that use of rough neurons in place of conventional neurons where ever possible will improve the performance of the neural networks. Another interesting observation from the experiment is the change in the sum of squared errors during the training process. Fig. 5 shows reduction of errors during the training process for all the three networks. The reduction in errors is more dramatic for rough neural networks than the conventional network. The errors for second rough neural network are consistently lower than the first neural network confirming earlier observation regarding the use of hidden layer rough neurons.

6. Summary and Conclusion

This study proposed rough neural networks for estimating rough output patterns from rough input patterns. A rough pattern uses upper and lower bounds of the values as opposed to precise values. The rough neural networks use a combination of rough and conventional neurons. A rough neuron can be viewed as a pair of neurons. One neuron corresponds to the upper bound and the other corresponds to the lower bound. Upper and lower neuron exchange information with each other during the calculation of their outputs. The paper discussed different types of connections to and from rough neurons and the corresponding feedforward mechanisms.

The study compared the effectiveness of two different rough neural network models with a conventional neural network model for estimation of the design hourly traffic volume on a highway section. The problem lends itself very well for rough neural network modeling. The errors in estimation from rough neural network models are significantly lower than the conventional neural network model. Moreover, the addition of rough neurons in hidden layer seems to improve the prediction performance.

References

DeGarmo, E.P., Sullivan, W.G. and Canada, J.R. 1984. *Engineering Economy*, Macmillan Publishing Co., New York, N.Y., pp. 264-266.

Garber, N.J. and Hoel, L.A. 1988. *Traffic and Highway Engineering*, West Publishing Co., New York, N.Y., pp. 97-118.

Hecht-Nielsen, R. *Neurocomputing*, Addison-Wesley Publishing, Don Mills, Ontario.

Lingras, P.J. and Adamo, M. (1995). Estimation of AADT Volume Using Neural Networks, *Computing in Civil and Building Engineering*, Pahl & Werner (Eds.), 1355-1362.

Pawlak, Z. (1982). Rough sets, *International Journal of Information and Computer Sciences*, 11, pp. 145-172.

Pawlak, Z. (1984). Rough classification, *International Journal of Man-Machine Studies*, 20, pp. 469-483.

Polkowski, L., Skowron, A. and Zytkow, J. (1994). Rough Foundations for Rough Sets, *Conference Proceeding of Third International Workshop on Rough Sets and Soft Computing*, November 10-12, San Jose, California, pp. 142-149.

Shafer, G. (1976). *A Mathematical Theory of Evidence*, Princeton University Press, Princeton, New Jersey.

Sharma S.C. and Allipuram, R.R. 1993. Duration and Frequency of Seasonal Traffic Counts, Journal of Transportation Engineering, American Society of Civil Engineers, 116, 3, pp. 344-359.

Sharma S.C. and Werner, A. 1991. Improved Method of Grouping Provincewide Permanent Traffic Counters, Transportation Research Record 815, Transportation Research Board, Washington D.C., pp. 13-18.

White, H. 1989. Neural Network Learning and Statistics, *AI Expert*, 4, 12, 48-52 .

Wong, S.K.M. (1994). Rough Sets and Extended Models, *invited paper in the Third International Workshop on Rough Sets and Soft Computing*, November 10-12, San Jose, California.

Yao, Y.Y., Li X., Lin, T.Y. and Liu, Q. (1994). Representation and Classification of Rough Set Models, *Conference Proceeding of Third International Workshop on Rough Sets and Soft Computing*, November 10-12, San Jose, California, pp. 630-637.

Zell, *et al.* (1995). *Stuttgart Neural Network Simulator: User Manual Version 4.0*, University of Stuttgart, Institute of Parallel and Distributed High Performance Systems, Report No. 6/95.