Neural networks: fool's gold or layman's delight?

- a neural network model is used to predict the turnover of potential retail store locations

David Coates, Neil Doherty, Alan French and Malcolm Kirkup

This article addresses the application of neural networks to the problem of predicting store performance. The article considers the predictive power of neural networks, but concentrates on the issues associated with using them, in particular, whether neural networks can be successfully applied by managers with a rudimentary knowledge of personal computing.

-oo0oo-

This article describes the experience gained from using neural networks to tackle the practical problem of predicting the performance of individual stores belonging to an established retail clothing chain. The evaluation considers the two following issues:

1 Is the predictive power of neural networks useful?

2 How easy is it to use neural networks?

Many research articles concentrate on the first issue, pointing to the excellent results which can be achieved with neural networks, but neglect to discuss the actual processes involved in developing them.

Neural networks have been claimed to be viable alternatives to other approaches in a predictive sense, but how do they measure up in terms of usability? Therefore, this article evaluates the processes, *warts-and-all*, that were used to derive the results. The authors had no previous knowledge of using neural networks, and wanted to see whether a relatively simplistic, naive approach to generating store performance forecasts would work.

Neural networks

The potential of using neural networks for Operational Research problems has been described by Dodd (1992). Neural networks are composed of large numbers of processing elements, arranged in a highly inter-connected network. Each individual processing element takes a number of weighted inputs, and produces one output. On receipt of a set of input signals, a processing element firstly sums the inputs, and then applies a transfer function which determines how the inputs are transformed into an output. In simple terms the transfer function determines whether the output signal will excite or inhibit all of the processing elements to which it is subsequently connected. The architecture of a neural network consists of layers of processing elements. Every neural network has an input layer, an output layer and one or more hidden layers connecting the input layer to the output layer. An example of a neural network with five inputs, one output and one hidden layer containing three processing elements is shown in Figure 1.

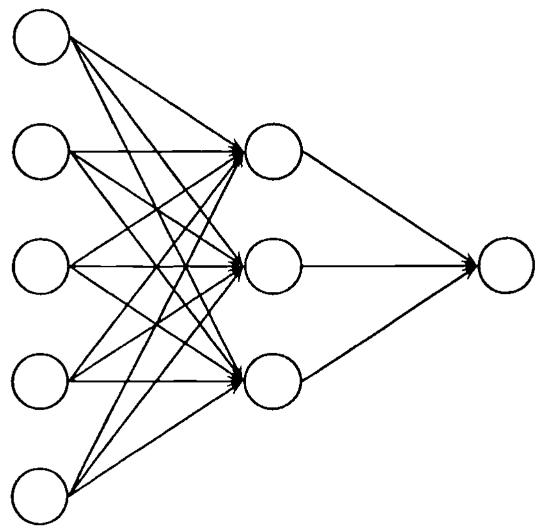


Figure 1: An example of a neural network

Neural networks have a generalised capability to learn, and consequently they need to be trained to solve specific problems. The training is conducted by applying a training set of inputs with their corresponding known outputs to the network. The network learns by applying learning rules which automatically adjust the weightings of the connections between the processing elements so as to decrease the difference between the network output and the known output. Having trained the network it is necessary to test it against a validation data set. The network outputs for the validation set can be compared with the actual output values to assess the effectiveness of the network's predictive capabilities. It should be noted that the ultimate performance of the trained network can only be guaranteed if it is supplied with good quality data for both training and validation. When training a neural network, it is important to be aware of the danger of *overtraining*. This is where the network can derive very close relationships between inputs and outputs in the training set, but loses its ability to generalise, and hence predict. Intuitively, neural networks are a particularly appealing OR tool for the non-expert to use, since being 'black box', they require no prior explicit knowledge of the relationships between variables.

The retail case study

It has long been recognised that a retail store's 'location' is a key determinant of its likely market share and profitability. In an age of intensifying retail competition, and similarity of product offerings, location has become still more important as a key source of differential advantage in retail marketing strategy.

For retailers, site selection decisions are critical. Given the long-term nature of leasehold or freehold agreements, the scale of leasing and construction costs, and increasing retail competition, retailers cannot afford the considerable financial and marketing penalties involved in premature closures and locational mistakes. A big element in site selection decisions is predicting the performance of proposed stores in various possible locations.

Given the limitations inherent in some traditional approaches to store performance forecasting, there is a clear need to identify and evaluate further alternative techniques.

The data used in this study were supplied by a specialist retail chain selling fashion clothing to a relatively broad target market. The chain has over 200 outlets spread throughout the UK, ranging in size from 800 to 9,000 sq ft.

The neural network software

As was noted earlier, the authors had no previous experience of using neural networks, therefore the first task was to purchase a suitable neural network package. Many alternative packages were considered by reviewing the literature and experimenting with demonstration packages, but ultimately the package chosen was NeuDesk. NeuDesk was chosen primarily because being a Windows based package, users who are familiar with the Windows environment will find it relatively straightforward to use. Furthermore, the package has spreadsheet style tables to facilitate easy data entry and comes with a wide range of training algorithms, which can be augmented with new algorithms as and when they become available.

Experimentation methodology

The experimentation methodology used to train a neural network for the task of predicting store performance is described in this section and is illustrated in Figure 2.

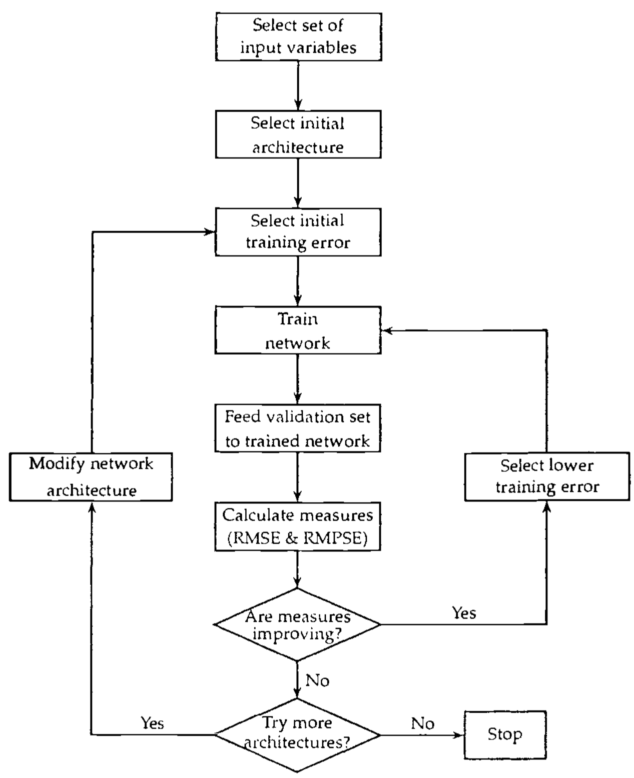


Figure 2: Demonstrating the training/validation process

The first phase was to determine what data should be included in the study. A subset of stores (38 in total) was chosen for analysis - all based in the South

West of England. This geographical region provides a relatively isolated and self-contained market area (with minimum loss of consumer expenditure to stores not included in the analysis) while also providing a representative sample of stores. The data was then amended further by separation into the two distinct data sets which would be used to train and validate the network (29 and 9 store records respectively). The choice of training and validation data sets was made to ensure that the two data sets were representative and that no outliers were present in the validation data.

The dependent, or output, variable chosen for this study was annual sales turnover, since this has been used widely in previous location analysis models. The independent, or input, variables chosen were population size, demographics, store size, competition and quality of location (pitch), all of which have been used in previous studies. All of these except population size and demographics were supplied by the retailer. Population size and demographic data were generated by gaining access to the CCN MOSAIC database. This database supplied by CCN Marketing contains a wide variety of geodemographic information for the whole of the United Kingdom, incorporating a host of census and financial variables to produce 'lifestyle' classifications of the residential population. The elements of this information utilised in this study were the 12 main lifestyle groups within the CCN MOSAIC classification, such as Suburban semis; Council flats; and Stylish singles, and the total population for the given catchment area. Consequently, the 17 input variables, as highlighted in Table 1, were utilised in this study.

Table 1: Variables utilised in study

Code	Description	Data type
TUR	turnover	integer (000s)
FLO	floor size	integer (sq. ft.)
MUL	number of comparison multiples	integer
PIT	pitch	1 to 5
COM	presence/absence of major competitor	0 or 1
L01 to L12	population in 12 MOSAIC lifestyle groups	integer
TOT	total population	integer

Having determined what base data was required for the range of experiments, it was assembled into the following four Lotus spreadsheets:

1. Training inputs: the 17 inputs for the 28 records of training data;
2. Training outputs: the single output for the 28 records of training data;
3. Validation inputs: the 17 inputs for the 9 records of validation data;
4. Validation outputs: the single output for the 9 records of validation data.

Each of these data files could then be read into NeuDesk's own spreadsheet style tables for training and validation data, as and when required.

The NeuDesk package comes equipped with four basic training algorithms, standard back propagation, stochastic back propagation, Weigend's weight elimination and quick propagation. Any of these algorithms could have been used for the neural network experimentation. In order to ascertain the most appropriate algorithm, a small pilot study was carried out. A small number of network architectures were selected, and each of the algorithms was used to train these networks. It was established that standard back propagation gave marginally better results than the other three, and this was consequently used for all remaining experiments.

A wide range of experiments was conducted with NeuDesk to identify the best network for forecasting the performance of stores. A series of network architectures could be evaluated by changing the number of nodes in the input layer, adjusting the number of nodes in each hidden layer, and also experimenting with the number of hidden layers.

In practice, the strategy adopted was to start with a small subset of the input variables and conduct a range of experiments on these variables starting with a single hidden layer containing one processing element, and then incrementally increasing the sophistication of the hidden layer configuration. The adjustments ranged from adding an extra processing element to a hidden layer, to adding an additional hidden layer.

Having determined a specific network architecture, the data in the training and validation input and output spreadsheets was loaded into NeuDesk. This network architecture was then easily created by deleting the columns of input data in both the training and validation sets that were not required, and then specifying the requisite number and composition of the hidden layers.

Training for a given architecture was terminated when the predictive power of the network began to decline. This was measured using the root mean square error (RMSE), and the root mean square percentage error (RMPSE). These two error measures were adopted because they are common measures which are widely used both in neural networks and regression modelling research. For a given architecture, the training stopped when these two measures began to increase on the validation set, indicating that the network was becoming overtrained.

Unfortunately, having trained the network and performed the validation the predicted turnover values had to be transferred back to Lotus to calculate the overall error values for the run, because NeuDesk currently does not provide calculation facilities. In practice, this was a simple task involving the standard Windows Copy and Paste facilities.

When a wide selection of experiments, ranging from the very simple to highly sophisticated had been reviewed for a given subset of the input variables, the next step was to modify the input set, by increasing the number of input variables, and then repeating the whole process. Though time consuming, this approach was highly systematic and ensured the detection of the best network. When small architectures, for example 2 or 3 input variables were used, the run-time for experiments took in the region of 5 to 10 minutes to complete using a PC with an 80386 processor running at 20MHz. The larger experiments, for example input sets with 15 variables and 25 processing elements in one hidden layer, running on the same machine were taking about an hour to complete. Fortunately, once started, NeuDesk requires no intervention from the user during the training phase.

Results of the experimentation

A wide range of input sets were tried, ranging from, for example, using just pitch (PIT) and the total population (TOT) to using all 17 possible inputs. It was discovered that the pitch and presence of a major competitor added nothing to the predictive power of the networks. The best result was found when the individual MOSAIC categories (L01 ... L12) were included along with the total population (TOT), the floor size (FLO) and the number of comparison multiples (MUL) giving 15 input variables. In real terms, the average absolute error when using the best performing network to predict the sales turnover for a given store, is in the order of ±£27,000, which equates to a percentage error of only 7.5%. The RMSE and RMPSE for this network were £35,700 and 9.9% respectively. The corresponding architecture had a single hidden layer with 25 processing elements.

From the experimentation a number of interesting points were noted. Firstly, for the data sets being used, the inclusion of more than one hidden layer failed to improve the predictive power of the network. It was also noted that the number of experiments necessary to find the best network architecture increased significantly as the number of input variables rose. The best network architecture for smaller input sets was found after only four or five architectures were tried, whereas for the largest input set (17 variables), the best network architectures did not become clear until 30 processing elements has been tried in the hidden layer.

Having successfully trained the neural network so that it could predict the sales turnover of a set of stores with an acceptably low error, the next stage was to evaluate the performance of the neural network by comparing it with an alternative technique. The traditional way of performing store location analysis modelling is through the use of regression (Jones and Simmons, 1990). Consequently a regression model was developed using the same training and validation sets as had been used for the neural network.

A wide range of regression models, using both the original linear values and log-log transformations, were developed and appraised, but ultimately the best regression model with an RMSE of £42,400 and an RMPSE of 10.2%, could not match the best neural network.

Utilising the model

The ultimate rationale for developing the neural network model was to be able to predict the turnover for any specific potential store location. This involved identifying a number of possible sites not currently serviced by the retailer, and assembling the corresponding MOSAIC variables and number of multiples. Predictions for turnover could then be made based on these variables, which are fixed for a given location, and a wide range of potential floor sizes. Table 2 shows the results of a set of experiments conducted for Salisbury which proved to be the most attractive new site. It should be noted that Table 2 ignores all the fixed variables, and therefore, only shows the floor sizes tested, and the resultant predicted turnovers. The range of floor sizes tested was limited to the range of values present in the training and validation input data sets. It is interesting to note that the relationship between floor size and turnover is non-linear. This feature reinforces the justification for using neural networks which are capable of modelling complex non-linear relationships between input and output variables.

Table 2: Predicted turnover figures for Salisbury

Floor size (sq. ft.)	Predicted Turnover (000s)
1000	339
1400	405
1800	475
2200	545
2600	611
3000	671
3400	722
3800	766
4200	801
4600	830

Implications

It has been shown that neural networks can be successfully applied to the problem of forecasting store performance, and that they compare favourably with the traditional regression modelling approach.

The methodology used with the neural networks is relatively straightforward, although time consuming, and potentially easy for an inexperienced analyst to understand and apply, providing they adopt a thorough and systematic approach. Furthermore, the NeuDesk software is user-friendly, can be readily used with little training, and makes it easy to accommodate modifications to the network architecture, training algorithms and error levels.

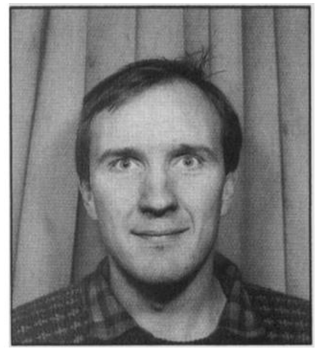
This case study indicates that neural networks deserve a place in the OR toolkit. Once the relevant data sets had been created, the task of finding and training a network architecture to learn the problem was straightforward and systematic. A major problem was the relatively exhaustive way in which the search for the best network architecture was conducted. Work is currently continuing to look for better experimentation strategies, and it is envisaged that subsequent experimentation could be carried out far more efficiently, saving much time.

For the interested reader

Coates D S, Doherty N F, French A P and Kirkup A H (1995) Neural networks for store performance forecasting: an empirical comparison with regression techniques, *International Review of Retail, Distribution and Consumer Research,* (in press, publication due soon).

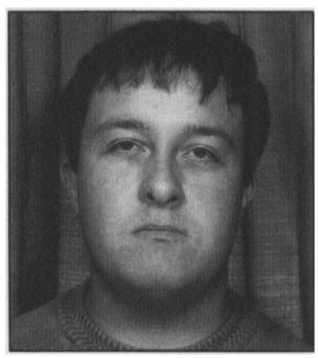
Dodd N (1992) 'An introduction to neural networks', in: Mortimer M E (ed) *Operational Research Tutorial Papers,* Operational Research Society, pp 47-66.

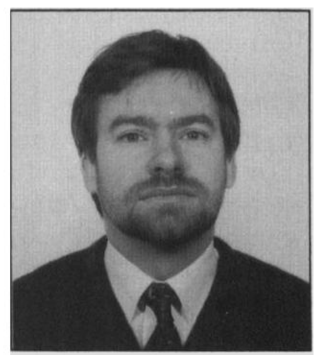
Jones K and Simmons J (1990) *The retail environment,* London: Routledge.



DAVE COATES (left) is a lecturer in statistics at Loughborough University Business School. His main research interests include applied statistics, time series analysis and total quality management. He has previously worked as a systems analyst for T I Chesterfield plc, and as a senior lecturer in statistics at the University of Plymouth.

NEIL DOHERTY (right) is a lecturer in management information systems at Loughborough University Business School. He has 10 years experience in the oil and aerospace industries. His research interests include the management applications of intelligent knowledge based systems, organisational issues in systems development and success factors in systems development.





ALAN FRENCH (left) is a lecturer in quantitative methods at Loughborough University Business School. Previously he worked for the Central Electricity Generating Board as a systems programmer. Research interests include management applications of neural networks, genetic algorithms and integer programming.

MALCOLM KIRKUP (right) is a lecturer in retailing at Loughborough University Business School. Formerly marketing planning manager with Sears plc, and lecturer in marketing at Cranfield University and Aston Business School. His research interests include retail location assessment and the application of computer models in retail location analysis.