Optimum Structure of Feed Forward Neural Networks by SOM Clustering of Neuron Activations

Samarasinghe, S.

Centre for Advanced Computational Solutions, Natural Resources Engineering Group
Lincoln University, Canterbury, New Zealand
Email: sandhya@lincoln.ac.nz

Keywords: *Feed forward networks, hidden neuron activation, correlated activation, clustering, self organising maps*

EXTENDED ABSTRACT

Neural Networks have the capability to approximate nonlinear functions to a high degree of accuracy owing to its nonlinear processing in the hidden layer neurons. However, the optimum network structure that is required for solving a particular problem is still an active area of research. In the past, several network pruning methods based on weight magnitude or sensitivity have been proposed and some example are magnitude based pruning (MBP) (Hagiwara, 1993); optimal brain damage (OBD) and its variants (Cun et al., 1990); and variance nullity (Engelbrecht, 2001). For example, in MBP, weights that are small are removed. How small is 'small' is a subjective decision. In OBD, weights that are not important for input-output mapping are found and removed. This is based on a saliency measure, s_i of a weight, as given in Eq. 1, that is an indication of the cost of setting it to zero. The larger the s_i, the greater the influence of w_i on error. It is computed from the Hessian (H) which is the matrix containing the second derivative of error with respect to a pair of weights in the network.

$$s_i = H_{ii}\, w_i^2 / 2 \qquad (1)$$

Saliency threshold however, is a subjective decision. Variance nullity measure proposes a variance analysis of the sensitivity of the output of the network to perturbation of weights. It is based on a hypothesis test using χ^2 (chi square) distribution to test statistically if a weight should be pruned. Here, null variance threshold must be defined. Teoh et al. (2006) proposes singular value decomposition (SVD) of hidden neuron activation to determine correlated neuron activations in order to select the required number of hidden neurons; however, the method requires heuristic judgment in setting up of a threshold parameter in the criteria for determining the optimum number of neurons. Xian et al. (2005) proposes an approach that utilizes the knowledge of the shape of the target function to determine the optimum number of neurons. This approach is efficient for 2- or 3-dimensional data where the shape of the target function can be ascertained relatively easily. However, the method cannot be applied to high-dimensional data as the target function cannot be visualized for such data. Another approach for optimising network structure is genetic and evolutionary algorithms (Castillo et al., 2000; Yao, 1999) that involve extensive and time consuming search, and in comparison to other network optimisation methods, these provide the least insight into the operation of a network.

In this paper, a new method based on the correlation of the weighted activation of the hidden neurons combined with the Self Organisation Feature Maps is presented for obtaining the optimum network structure efficiently. In an extensive search for internal consistency of hidden neuron activation patterns in a network, it was found that the weighted hidden neuron activations feeding the output neuron(s) displayed remarkably consistent patterns. Specifically, redundant hidden neurons exhibit weighted activation patterns that are highly correlated. Therefore, the paper proposes identifying hidden neurons with weighted activation patterns that are highly correlated and using one neuron to represent a group of correlated neurons. The paper proposes to automate this process in two steps: 1) Map the correlated weighted hidden neuron activation patterns onto a self organising map; and 2) Form clusters of SOM neurons themselves to find the maximum likely number of clusters of correlated activity patterns. The likely number of clusters on the map indicates the required number of hidden neurons to model the data.

The paper highlights the approach using an example and demonstrates its application to solving two problems including a realistic problem of predicting river flows in a catchment in New Zealand.

1. INTRODUCTION

Multi-layer feed forward neural network is the most powerful and most popular neural network for nonlinear regression (Samarasinghe, 2006). A neural network with enough parameters can approximate any nonlinear function to any degree of accuracy due to the collective operation of flexible nonlinear transfer functions in the network. However, finding the adequate complexity of network structure that is the optimum for a particular problem is still an active research problem. Neural networks are still treated as a black box due to lack of transparency in the internal operation of networks. This paper demonstrates that there is internal consistency of networks at the level of output layer processing and shows that it can be used for removing redundancy in a network. Here, the approach is highlighted and its validity is demonstrated through application to solve two problems.

2. OBJECTIVES

The goal of this paper is to demonstrate the possibility of optimising the structure of multilayer perceptron networks using internally consistent and correlated neuron activation patterns at the output layer level to remove redundant neurons. Specifically, it has the following objectives:

1. To demonstrate through an example that redundant neurons in a network show correlated activity patterns which can be used to obtain the optimum structure by clustering these patterns using self organizing maps.

2. To demonstrate the application of the above method to solve two more problems: Sine function approximation; and predicting river flows.

3. BACKGROUND

Feed forward networks have been applied extensively in many fields. In many training cases, a network with larger than required number of neurons are trained and stopped early when the level of required accuracy is achieved (Samarasinghe, 2006). Some pruning methods based on magnitude of weights or sensitivity, such as magnitude based pruning (Hagiwara, 1993), optimal brain damage and its variants (Cun et al., 1990) and variance nullity (Engelbrecht, 2001) have been proposed to prune networks to obtain the optimum network structure. However, they require considerable judgment on the part of the user in setting threshold levels for parameters used as pruning criteria.

Xian et al. (2005) proposes an approach that utilizes the knowledge of the shape of the target function to determine the optimum number of neurons. This approach is efficient for 2- or 3-dimensional data where the shape of the target function can be ascertained relatively easily. However, the method cannot be applied to high-dimensional data as the target function cannot be visualized for such data. Another approach for optimising network structure is genetic and evolutionary algorithms (Castillo et al., 2000; Yao, 1999) that involve extensive and time consuming search, and in comparison to other network optimisation methods, these provide the least insight into the operation of a network.

Teoh et al. [10] proposes singular value decomposition (SVD) of hidden neuron activation to determine correlated neuron activations in order to select the required number of hidden neurons. It is a step toward meaningful investigation into hidden neuron activation space; however, as authors point out, the method requires heuristic judgment in setting up of a threshold parameter in the criteria for determining the optimum number of neurons

4. METHODS

Figure 1 (a) (solid line) shows a one-dimensional nonlinear function used in this paper to demonstrate that redundant neurons in a network form highly correlated activity patterns. This function has the form

$$t = \begin{cases} 0.3\,Sin\,x & If\ \ x < 0 \\ Sin\,x & otherwise \end{cases} \qquad (2)$$

A total of 45 observations were extracted from this function and these were modified further by adding a random noise generated from a Gaussian distribution with 0 mean and standard deviation of 0.25 as depicted by dots in Fig. 1(a).

This data requires 2 neurons to model the regions of inflection. A larger network of 4 hidden neurons, as shown in Fig. 1(b) was used for the purpose of investigation. In this network, the hidden neuron activation functions are logistic, output neuron is linear, the bias and input-hidden weights of neuron j and input i are depicted by a_{0j} and a_{ij}, respectively, and hidden-output weights and the corresponding bias are denoted by b_j and b_0, respectively.

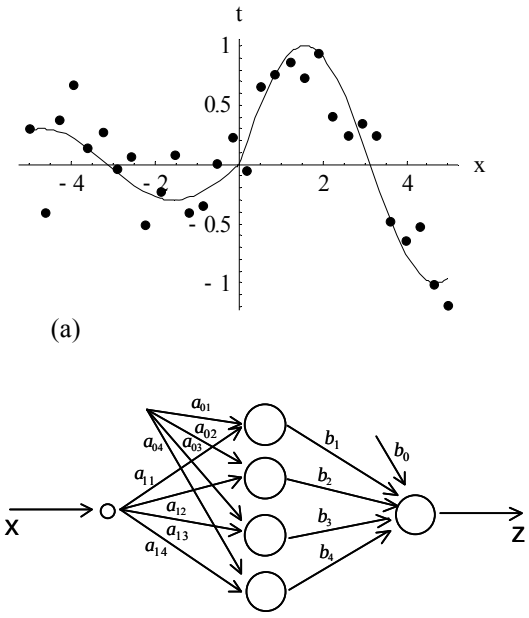
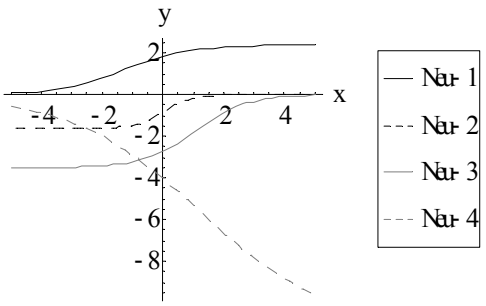


Figure 1. (a) Target data and its generator and (b) 4-neuron network to model the data

Network was trained with second order Levenberg Marquardt method with: 3 random weight initialisations; 3 randomly extracted data sets; and both early stopping and Regularization methods to stop training (Neural Networks for Mathematica (2003)). In all these cases, the internal structure of the networks was critically examined at the hidden layer level as well as the output layer level in search of consistent patterns of neuron activations. This revealed a very consistent and correlated patterns of weighted hidden neuron activations feeding the output layer, which is the contribution of each neuron j to output generation depicted by:

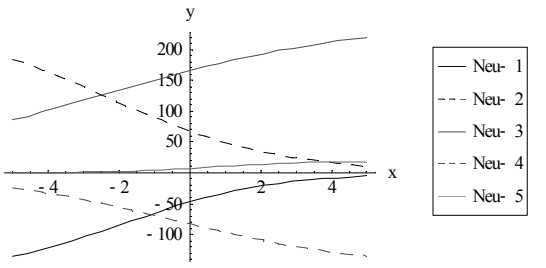
$$y_{weighted_j} = y_j b_j \qquad (3)$$

where y_j is the output of hidden neuron j and b_j is the corresponding weight linking neuron j with the output. This is shown, for one of the tested conditions mentioned above, in Figure 2 which clearly demonstrates that some weighted activations are parallel to each other and are highly correlated with values indicated in the correlation matrix found below the figure. The other conditions displayed similar correlated activity patterns. When trained with 5 neurons, similar display of strong parallelism among groups of correlated weighted hidden neuron activation patterns were found as shown in Figure 3. These correlated neurons point to redundancy where only one neuron would be enough to represent each cluster of correlated neurons, thus revealing optimum number of neurons for the network.



1.	0.938231	0.883288	- 0.929376
0.938231	1.	0.957974	- 0.9482
0.883288	0.957974	1.	- 0.982796
- 0.929376	- 0.9482	- 0.982796	1.

Figure 2. Weighted hidden neuron activations and their correlations for 4- neuron network



1.	- 0.999657	0.996011	- 0.988775	0.957319
0.999657	1.	- 0.996758	0.988397	- 0.9542
0.996011	- 0.996758	1.	- 0.996547	0.969492
- 0.988775	0.988397	- 0.996547	1.	- 0.985844
0.957319	- 0.9542	0.969492	- 0.985844	1.

Figure 3. Weighted hidden neuron activations and their correlations for 5-neuron network

This process can be automated by using self organizing maps (SOM) to cluster correlated neurons using correlation as the distance measure (Samarasinghe, 2006) and then cluster the map neurons using a method such as ward clustering to automatically determine the number of required neurons for the final network. This is illustrated in Figures 4 and 5 for the 4- and 5- neuron networks. In these figures, the top graphs indicate the ward likelihood index (Samarasinghe, 2006) for various cluster sizes. The higher the index, more likely the corresponding number of clusters. Both these figures indicate the highest index for two clusters of correlated neurons, indicating that the optimum network in this case requires two neurons, which agrees with the reality. The

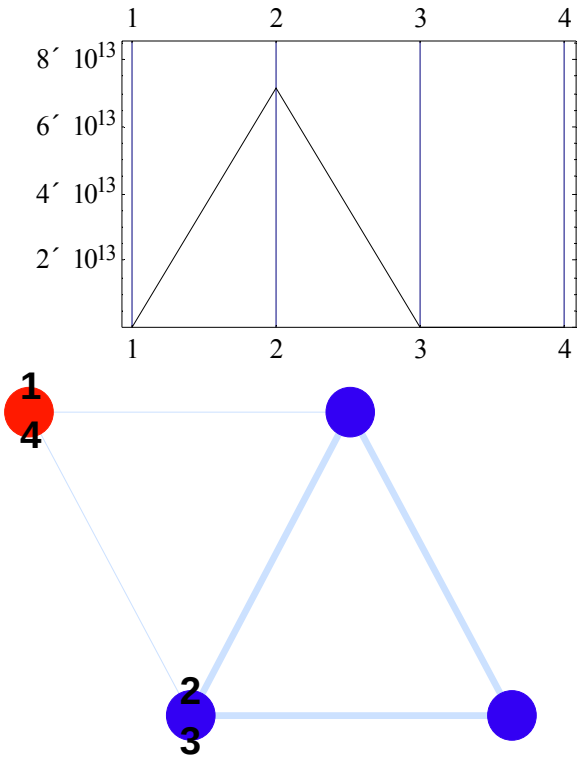


Figure 4 Clustering of Self organising map neurons by Ward clustering method (4- neuron network)

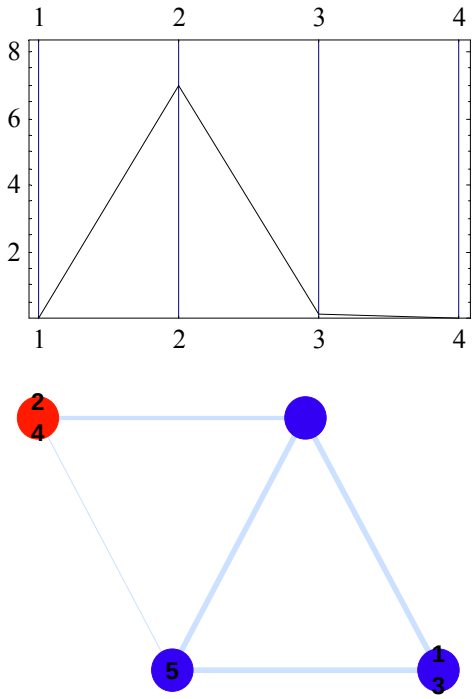


Figure 5 Clustering of SOM (5-neuron net)

bottom figures are the SOMs clustered according to the optimum number of clusters determined by the ward method . Both 4- and 5- neuron networks now display 2 clusters that correspond to the required number of neurons to model the function in Figure 1.

4.1 Application 2: Sine function Approximation

In this paper, the above idea is extended to two larger applications to demonstrate its robustness. First of these is the approximation of the Sine function from the data shown in Figure 6.

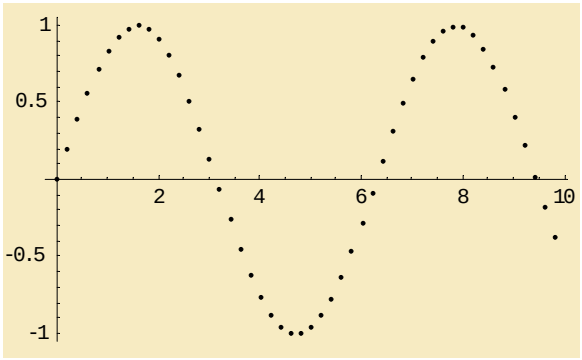


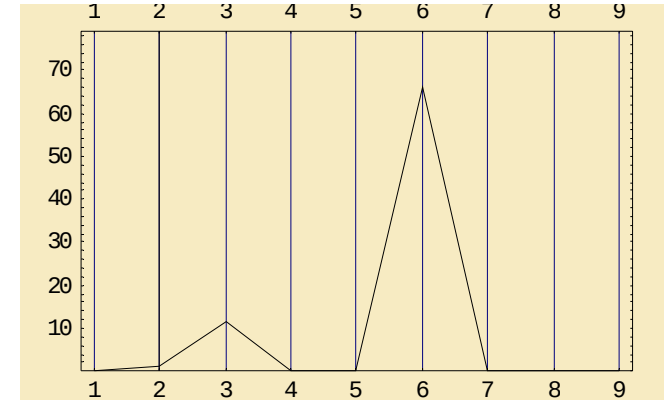
Figure 6 The data from Sine function

The Sine function requires 4 hidden neurons in the optimum model. A network with 10 hidden neurons with logistic activation was selected and Levenberg Marquardt method was used to train it and Regularization to stop training. Due to the large range in the weighted activations, all patterns cannot be shown together graphically but the correlation matrix is given Figure 7 that indicates that some activations are highly correlated and some are weakly or moderately correlated. It also shows that it is not easy to discern the optimum number of clusters when the number of neurons is large.

1.	0.998317	0.991452	0.975579	0.945358	0.895051	0.821693	0.728914	0.627193	0.528909	0.442825
0.998317	1.	0.997349	0.98667	0.962659	0.919372	0.8533	0.767296	0.671181	0.577082	0.493898
0.991452	0.997349	1.	0.995895	0.979799	0.945543	0.888946	0.811872	0.723265	0.634881	0.555745
0.975579	0.98667	0.995895	1.	0.993878	0.971121	0.926743	0.861364	0.78277	0.702161	0.628655
0.945358	0.962659	0.979799	0.993878	1.	0.991534	0.962573	0.912205	0.846715	0.776504	0.710695
0.895051	0.919372	0.945543	0.971121	0.991534	1.	0.989615	0.957682	0.908626	0.851738	0.796012
0.821693	0.8533	0.888946	0.926743	0.962573	0.989615	1.	0.98911	0.959218	0.918207	0.874746
0.728914	0.767296	0.811872	0.861364	0.912205	0.957682	0.98911	1.	0.990374	0.966505	0.936538
0.627193	0.671181	0.723265	0.78277	0.846715	0.908626	0.959218	0.990374	1.	0.992725	0.976047
0.528909	0.577082	0.634881	0.702161	0.776504	0.851738	0.918207	0.966505	0.992725	1.	0.995141
0.442825	0.493898	0.555745	0.628655	0.710695	0.796012	0.874746	0.936538	0.976047	0.995141	1.

Figure 7. Correlation matrix of the 10 weighted hidden neuron activations

Figure 8 (top) shows the Ward likelihood index which indicates 3 and 6 clusters as the most likely with 6 clusters having the highest likelihood. The trained SOM map divided into 6 clusters using the Ward method is shown in bottom Figure 8(bottom).



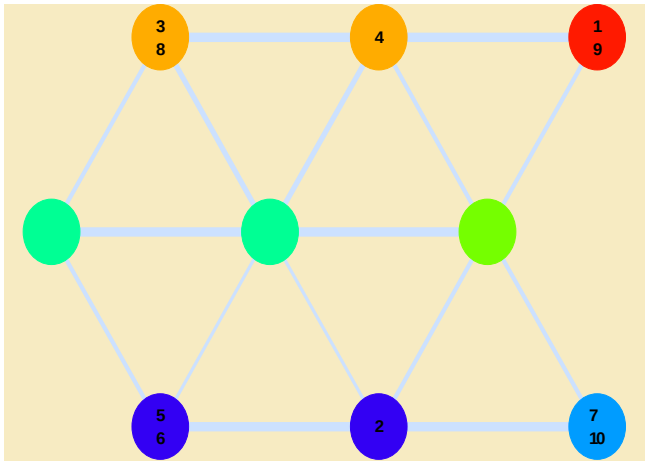


Figure 8 Ward likelihood index against number of clusters (top) and clustered SOM map according to highest likelihood index (bottom)

Figure 8(bottom) reveals that although map has been divided into 6 clusters, effectively there are four clusters represented by yellow, red, dark blue and light blue colours. Neurons in these clusters have labels depicting the weighted activations of the neurons in the original neural network. The other two clusters, indicated by light green and dark green colours, do not represent any of the activation patterns.

Thus, from the map, it can be discerned that the original Sine function requires four neurons to model it effectively. The optimum model output superimposed on data is shown in Figure 9.

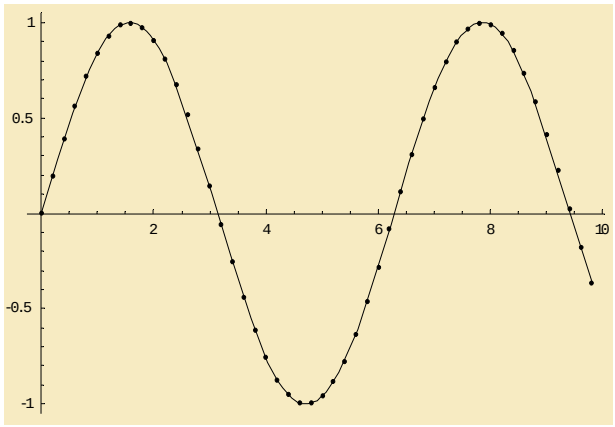


Figure 9. Optimum network output superimposed on data

4.2 Application 3: Predicting river flows in a catchment

Reliable estimates of river flows are important for planning and managing water resources. For this purpose, flow gauges are used to measure river flows. However, in difficult to reach catchments, this is not always possible and some river flows need to be estimated using other methods such as modeling. In this example, 8 river basins in the

Canterbury region in the South Island of New Zealand are selected for a case study. Reason for selecting this case study was that it has been modeled previously by Vieira (2006) using neural networks and therefore, the proposed method can be compared against it.

There were roughly three rainfall gauge stations per catchment. River flow and Rainfall time series for over 5 years were obtained from Environment Canterbury (Ecan, 2006) records. One of the main variables influencing river flow is precipitation whose effect on flow depends on the drainage area. Another factor affecting flow is forests and vegetation that delays run-off or limits it. This is further impacted by soil type- specifically, its permeability. Flow also depends on ground slope and drainage density; the latter is the total channel length to watershed area indicating fraction of the watershed occupied by channels.

The required physical characteristics of the catchments were obtained from a GIS database and these comprised of Drainage area, Average ground slope, Drainage Density, Basin form, and Land use (Open, cultivated, Forest, Wetlands). Basin form is an indication of shape of the basin and was introduced to account for the fact that long narrow forms on average produce lower flows at river mouth. Larger the form lower the flow. The objective of the study was to develop a neural network to estimate monthly average flow in a river from known flows in other rivers in the region, rainfall and basin characteristics. For this purpose, Waipara river was selected as the one requiring estimation. Its measured flows are used to validate the estimates.

Before building the model, a rigorous pre-processing and input selection method was followed. Simple and partial correlation analysis was conducted in three stages, refining the input selection in successive stages. Fourteen input variables including flow and precipitation lags were considered. The final analysis revealed that the current monthly average flow was influenced by current month's precipitation (correlation coeff (CF) = 0.557), previous month's flow (CF = 0.352) and a compound factor computed as (Total catchment area- Forested area) (CF = 0.331) and these inputs were used to build a network.

Combined catchment training data consisted of 1079 records, calibration set for testing model accuracy during training had 269 records. The validation set contained 63 records.
Vieira (2006) tested various model architectures on NeuroShell 2 (1997) and found that the optimum network was a 3- layer network trained with backpropagation with momentum. After this training with selected inputs, a series of different networks were built again with the addition of secondary variables, and watershed form came out as a significant secondary variable. The final best model produced training R^2 of 0.94 and validation R^2 of 0.729. The contributions of inputs to the output were: Precipitation (42.6%), Previous flow (32.7%), compound factor (21%), and form (3%).

Vieira's (2006) model architecture was: 4 inputs, one hidden layer with 70 neurons with logistic activation, output neuron with logistic activation. In the present study the same inputs were used to obtain the optimum network using the proposed approach.

5. OPTIMISATION OF STRUCTURE USING CORRELATED ACTIVATION PATTERNS AND SOM

A 3-layer network with 100 hidden neurons was trained with backpropagation with momentum for predicting the river flow and the weighted hidden neuron activation patterns were projected onto a 100-neuron SOM. The Ward likelihood index for possible clusters of map neurons is shown in Figure 10 which reveals that 2 and 3 clusters have the highest likelihood index followed by 11. This was much smaller than that reported by Vieira (2006) and immediately there was doubt as to the ability of the proposed method in optimising the network structure for complex problems. The SOM structure itself was then carefully reviewed for the spread of activation patterns. It revealed that 100 patterns have been grouped into 59 neurons. The map and the two clusters formed by the Ward method are shown are in Figure 11. A network with 2 hidden neurons were then trained for investigating if this were true and it in deed was found to be the optimum number of neurons

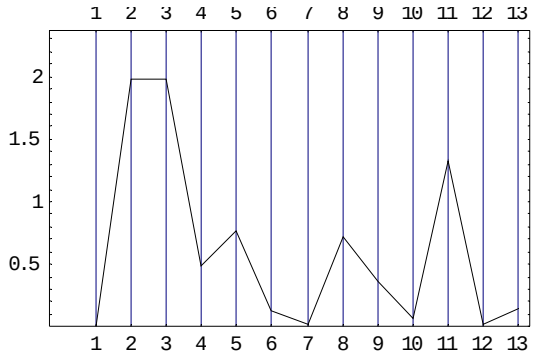


Figure 10 Ward likelihood index for possible SOM neuron clusters

giving training R^2 of 0.88 and validation R^2 of 0.71. Predictions (Figure 12) and contribution of variables were similar to those of Vieria (2006). Results were similar for 59 and 70 neurons.

Thus, the projection of correlated weighted hidden neuron activations onto an SOM followed by clustering of map neurons revealed the required number of neurons with certainty. These results indicate that the proposed approach is useful in optimising network structure for solving even complex realistic problems.

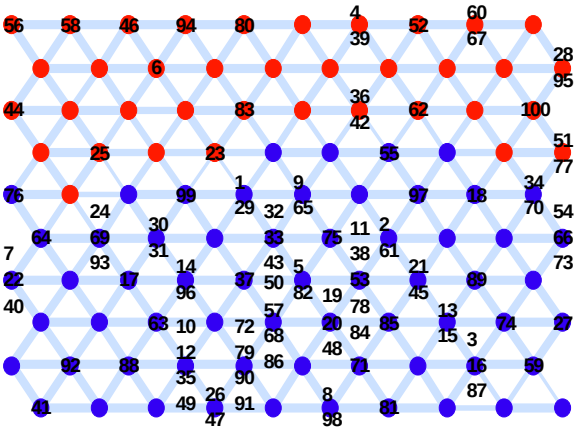


Figure 11. SOM Clustering of weighted hidden neuron activation patterns in a network predicting river flow

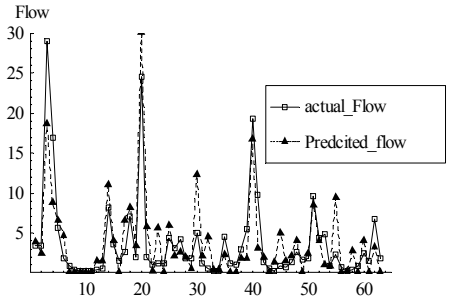


Figure 12. Predicted flow superimposed on actual flow in validation dataset

6. SUMMARY AND CONCLUSIONS

This paper presented the results from a study to optimise the hidden layer neurons in multilayer perceptron network by using the internal consistency of hidden neuron activation patterns. Through three example applications- two nonlinear functions and a realistic river flow prediction- it demonstrated that redundant weighted hidden neuron activations became strongly correlated when there were too many hidden neurons in the network and these patterns were easily mapped onto an SOM. Furthermore, SOM neurons themselves were further clustered by assembling map neurons that are closer together based on Ward clustering and the highest likely number of clusters on the map indicated the optimum number of neurons required in the network. In the river flow prediction, it successfully revealed the required number of 2 neurons from the activation patterns of a 100-neuron network that was trained and further subjected to clustering using SOM and Ward methods. The paper convincingly highlighted the robustness of the proposed approach in optimising the hidden layer through removing the redundant neurons indicated by the correlation of weighted hidden neuron activations feeding the output. Therefore, it proposes a novel approach to structure optimisation that is meaningful and logical.

7. REFERENCES

Castillo, P.A., J. Carpio, J.J. Merelo, V. Rivas, G. Romero, and A. Prieto (2000), Evolving multilayer perceptrons, *Neural Processing Lett.* 12(2), 115-127.

Engelbrecht, A.P. (2001) A new pruning heuristic based on variance analysis of sensitivity information, *IEEE Transactions on Neural Networks*. Vol.12(6), 1386-1399.

Environment Canterbury (Ecan) (2006). NZ

Hagiwara, M. (1993) Removal of hidden units and weights for backpropagation networks, *Proc. Int. Joint Conf. Neural Networks*, l.1, 351-354.

Le Cun, Y., J.S. Denker, and S.A. Solla, (1990) Optimal brain damage, *Advances in Neural Inf. Pro. (2),* pp.598-605.

Machine learning framework for Mathematica. 2002. www.unisoftwareplus.com.

Neural Networks for Mathematica, 2002. Wolfram Research, Inc. USA.

Samarasinghe, S. (2006) Neural Networks for Applied Sciences and Engineering-From Fundamentals to Complex Pattern Recognition. CRC Press. USA.

Teoh, E.J., K.C. Tan, and C. Xiang (2006) Estimating the number of hidden neurons in a feed forward network using the singular value decomposition, *IEEE Trans. on Neural Networks,*. 17(6).

Vieria, J. 2006. Regional analysis of hydrologic events in the Canterbury region- A case study using GIS and Neural Networks, Dissertation, Lincoln University, New Zealand.

Ward, J.H. Jr. (1963) Hierarchical grouping to optimise an objective function, *J. of the American Stat.Assoc.*, 58, 236-244.

Xian, C., S.Q. Ding, and T.H. Lee. (2005), Geometrical interpretation and architecture selection of MLP, *IEEE Trans. on Neural Networks,* 16(1).

Yao, X. (1999) Evolutionary artificial neural networks, *Proc. IEEE*, 87(9),1423-1447.