DATA MINING OF INPUTS: ANALYSING MAGNITUDE AND FUNCTIONAL MEASURES

Tamás D. Gedeon

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052 AUSTRALIA
tom@cse.unsw.edu.au
Fax +61 2 9385 5995

Abstract

The problem of data encoding and feature selection for training back-propagation neural networks is well known. The basic principles are to avoid encrypting the underlying structure of the data, and to avoid using irrelevant inputs. This is not easy in the real world, where we often receive data which has been processed by at least one previous user. The data may contain too many instances of some class, and too few instances of other classes. Real data sets often include many irrelevant or redundant input fields. This paper examines the use of weight matrix analysis techniques and functional measures using two real (and hence noisy) data sets.

The first part of this paper examines the use of the weight matrix of the trained neural network itself to determine which inputs are significant. A new techniques is introduced, and compared with two other techniques from the literature. We present our experience and results on some satellite data augmented by a terrain model. The task was to predict the forest supra-type based on the available information. A brute force technique eliminating randomly selected inputs was used to validate our approach.

The second part of this paper examines the use of measures to determine the functional contribution of inputs to outputs. Inputs which include minor but unique information to the network are more significant than inputs with higher magnitude contribution but providing redundant information, which is also provided by another input. A comparison is made to sensitivity analysis, where the sensitivity of outputs to input perturbation is used as a measure of the significance of inputs.

This paper presents a novel functional analysis of the weight matrix based on a technique developed for determining the behavioural significance of hidden neurons. This is compared with the application of the same technique to the training and test data available. Finally, a novel aggregation technique is introduced.

Introduction

The networks used in this paper were trained using error-backpropagation (Rumelhart, Hinton and Williams, 1986). All connections are from units in one level to units in the next level, with no lateral, backward or multi-layer connections. Each unit is connected to each unit in the preceding layer by a simple weighted link. The network is trained using a training set of input patterns with desired outputs, using the back-propagation of error measures. The network training is terminated using a test set of patterns which are never seen by the network during training. That is, when the error on the test set is a minimum provides a good measure of the generalisation capabilities of the network. As the test set is not used to design the topology, nor used to select the best from a group of networks, we do not need a third validation set. We note that some authors reverse the naming of the test/validation sets from our description, which we have seen more commonly used. All results quoted in this paper are for the test set. Where multiple networks are mentioned and only one result is given this is invariably for the average of multiple runs, and is not the result of the best run. We have used the basic sigmoid logistic activation function, $y = \left(1 + e^{-x}\right)^{-1}$, though this is not essential to the substance of our results.

The following sections describe the sets of experiments on two collections of data, first a satellite image collection, and second a medical data set.

1.1. Network topology – GIS data

The initial network topology was 16-10-5, being sixteen inputs, ten hidden neurons, and five output neurons. The topology was chosen based on some preliminary experiments using only the data which became the training data for the final network. The raw data for this study comes from an area in the Nullica State Forest on the south coast of New South Wales, Australia. The available information is from a rectangular grid of 179,831 pixels 30 m by 30 m, and is a vector of 16 values (Bustos and Gedeon, 1995). Each pixel has a value for altitude, aspect, slope, geology, topographic position, rainfall, temperature (from a terrain model derived from soil maps, aerial photography and so on), and Landsat TM bands 1 to 7. The outputs are the forest supra-type, being *scrub*, *dry sclerophyll*, *wet-dry sclerophyll*, *wet sclerophyll*, and *rainforest*. For the purpose of training 190 detailed sample plots have been surveyed (Milne, Gedeon and Skidmore, 1995). This data gives us classifications for 190 of the pixels in the field area, and have used 150 for training the neural network and retained 40 for testing.

1.2. Analysis Techniques

Garson (1991) proposed the following measure for the proportional contribution of an input to a particular output:

$$G_{ik} = \frac{\sum_{j=1}^{nh} \frac{w_{ij}}{\sum_{p=1}^{ni} w_{pj}} . w_{jk}}{\sum_{q=1}^{ni} \left(\sum_{j=1}^{nh} \frac{w_{qj}}{\sum_{p=1}^{ni} w_{pj}} . w_{qj} \right)} \tag{1}$$

A disadvantage of this approach is that during the summation process, positive and negative weights can cancel their contribution which leads to inconsistent results.

Wong, Gedeon and Taggart (1995) used the following measure for the contribution of an input to a neuron in the hidden layer:

$$P_{ij} = \frac{|w_{ij}|}{\sum_{p=1}^{ni} |w_{pj}|} \tag{2}$$

Milne (1995) commented that the sign of the contribution is lost, and proposed the following measure:

$$M_{ik} = \frac{\sum_{j=1}^{nh} \frac{w_{ij}}{\sum_{p=1}^{ni} |w_{pj}|} . w_{jk}}{\sum_{q=1}^{ni} \left(\sum_{j=1}^{nh} \left| \frac{w_{qj}}{\sum_{p=1}^{ni} |w_{pj}|} . w_{qj} \right| \right)} \tag{3}$$

The measure introduced here is an extension of our technique (Wong, Gedeon and Taggart, 1995). We can define a measure P_{jk} for the contribution of a hidden neuron to an output neuron similar to the measure P_{ij} used above:

$$P_{jk} = \frac{|w_{jk}|}{\sum_{r=1}^{nh} |w_{rk}|} \tag{4}$$

The contribution of an input neuron to an output neuron is then:

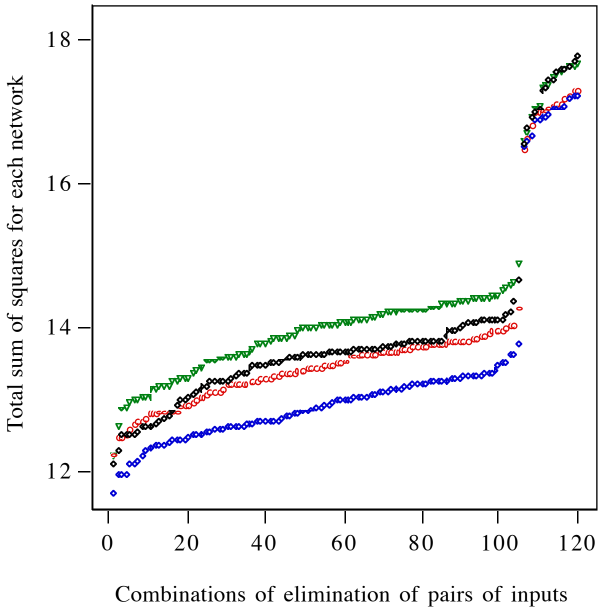
$$Q_{ik} = \sum_{r=1}^{nh} \left(P_{ir} \times P_{rk} \right) \qquad (5)$$

The benefit of this approach is that the magnitude of the contribution is disentangled from the sign of the contribution. The magnitude of contributions is significant in indicating whether an input is important, while the sign of contribution is largely irrelevant in the decision to remove or retain an input, and is recoverable in any case from the raw data by simple statistical methods.

Each of the above techniques could be extended to networks with larger numbers of hidden layers than the topology used in this experiment.

1.3. Brute force analysis

The brute force approach is to eliminate inputs and to compare the results with the predictions. Eliminating only 1 input produced inconsistent results, hence 2 inputs were eliminated. With 16 inputs, there are 120 ways to chose 2 inputs to remove. Four networks with the same topology (14-10-5) were trained for each of the 120 possibilities.



The above graph shows the results on the best total sum of squares (tss) value on the test set for each of the 480 networks run. The initial weights for each of the 4 runs was generated for the full 16 input network, and the appropriate weights excluded when 2 inputs were eliminated. Thus, each run had largely the same initial (random) starting weights. This as some small effect on all of the descendant networks, as shown by the consistent (minor) overall differences between the curves.

The tss values are sorted into increasing order, as there is no meaningful 1 dimensional scale on which to represent the removal of pairs of inputs. The discontinuity demonstrates the point at which there was some significant degradation of the neural network prediction, which correlated well with the tss values. The right hand part of the graph and the leftmost part of the longer curve were used to determine the most and least significant inputs, respectively, by calculating the average rank for the tss values for each combination of inputs.

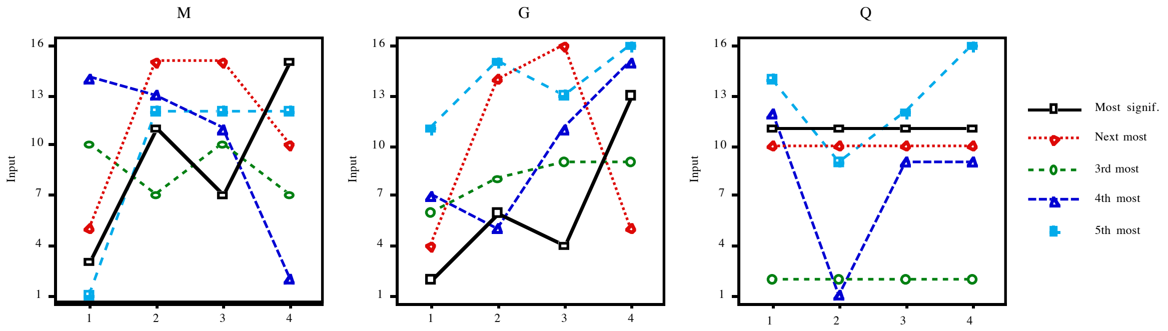
1.4. Comparison of results

The contribution of inputs to each of the 5 outputs were averaged to determine the significance of inputs to the entire task the network is solving. For clarity, the comparisons are made between the brute force technique and the calculated measures on the top and bottom thirds of the ordering.

model	Most significant					…	Least significant				
B	5	1	10	14	11	…	7	8	13	9	6
Q	11	10	2	12	14	…	13	5	4	8	7
G	2	4	6	7	11	…	9	13	14	1	15
M	11	15	7	13	12	…	8	10	2	4	5

From the above table it is clear that the model Q introduced in this paper is ${}^3/_5$ths in accord with the brute force method B for both the most and least significant inputs, while both of the other methods are only ${}^1/_5$ths in accord on either end of the significance scale.

The following diagrams show the changes in the 5 most significant inputs on overtraining.



Clearly model Q is most robust during overtraining, then M, with model G being inconsistent.

1.5. Discussion

A measure for determining the contributions different inputs make to the outputs was introduced, which was validated by a brute force input ranking technique eliminating all combinations of pairs of inputs. The measure was 60% in accord with the brute force ranking, while the comparison measures were only 20% in accord.

The measure introduced here is also more stable during training, which suggests that there is closer coupling to the network behaviour over time than with the other measures. Note that this stability is not due to the use of a different network, but is a result of the choice of formula solely.

2.1. Network topology – Medical data

The initial network topology was 12-7-1, being twelve inputs, seven hidden neurons, and one output neurons. The data for this study was acquired from a novel eye gaze detector developed at Westmead Hospital. The major advantage of eye gaze data is that when we know where the eye is looking, we know the contents of the major input channel to the brain.

For example, the point of first fixation for schizophrenic versus normal controls on a neutral affect face produces results which statistically separate the two cases. This work extends this classification process to reliably classify the individual cases based on multiple responses to a wire frame drawing, a neutral affect face, a happy face and a sad face. This initial trial used 10 schizophrenic and 10 normal individuals, with 4 responses of 10 seconds duration recorded at 50 Hz (Gedeon et al, 1996).

The detector uses infra-red to detect the difference between the angle of reflection from the front of the eye and the retina to determine where on screen the subject is looking. The data used in this paper makes use only of the summary statistics of the entire data stream, with respect to fixations of gaze of 200 msec or longer.

The twelve inputs are: x and y co-ordinates; overall distance, horizontal, and vertical distance to previous fixation point; distance to previous fixation point relative to scan distance; pupil area; pupil area relative to pre and post-stimulus pupil areas; dwell time; and relative dwell time compared to the average dwell time; and finally, which image in being looked at.

The single output classifies by values above/below 0.5 whether the particular patterns belongs to a normal control or schizophrenic patient. Note the this problem is particularly hard, as the network needs to determine a classification based on the current eye gaze location and the difference from the previous one.

Previous work has investigated the use of related soft computing techniques, being vector quantisation and simulated annealing in the classification of schizophrenic versus medicated schizophrenic patients versus normal controls (Haig, Gordon, et al, 1995).

2.2. Magnitude measures of contributions

Section 1.2 above has introduced the measures for the contributions of input to outputs. Henceforth, they will be referred to as *magnitude* measures as they use the weight magnitude information stored in the static weight matrix after training to determine significance.

Each of the techniques could be extended to networks with larger numbers of hidden layers than the topology used in this experiment.

2.3. Functional measures

The technique of distinctiveness analysis (Gedeon and Harris, 1991) uses hidden neuron activations over a training set to determine similarity using the angle between the multi-dimensional vectors thus formed. The formulae used are below.

$$\mathrm{angle}(i, j) = \tan^{-1}\left(\sqrt{\frac{\sum_{p}^{\mathrm{pats}} \mathrm{sact}(p, i)^2 * \sum_{p}^{\mathrm{pats}} \mathrm{sact}(p, j)^2}{\sum_{p}^{\mathrm{pats}} (\mathrm{sact}(p, i) * \mathrm{sact}(p, j))^2} - 1}\right) \tag{6}$$

$$\text{where} \qquad \mathrm{sact}(p, h) = \mathrm{activation}(p, h) - 0.5 \tag{7}$$

The technique has been extended for examining the functionality of hidden neurons using the weight matrix (Gedeon, 1996a), and is adapted here to determine the functional differences between inputs as represented by the pattern of input to hidden weights.

For this purpose (model *W* below), equation (7) becomes:

$$\text{where} \qquad \mathrm{sact}(p, h) = \mathrm{norm}(\mathrm{weight}(h)) - 0.5 \tag{8}$$

For comparison purposes, the pattern of values of inputs in the labelled set available (being both training and test sets) is analysed in an analogous fashion, That is, the values for a particular input in all of the instances in the labelled set are used to construct a multi-dimensional vector.

For this purpose (model *I* below), equation (7) becomes:

$$\text{where} \qquad \mathrm{sact}(p, h) = \mathrm{pattern}(h) - 0.5 \tag{9}$$

This 1,334 dimensional vector is then compared to the vectors for the other 11 inputs. Note that this is essentially a first-order correlation measure, and does not incorporate the possible higher order features that a neural network could learn and incorporate into the internal representation encoded in its weight matrix. Hence we would expect this measure to be less reliable than the distinctiveness approach applied to the weight matrix.

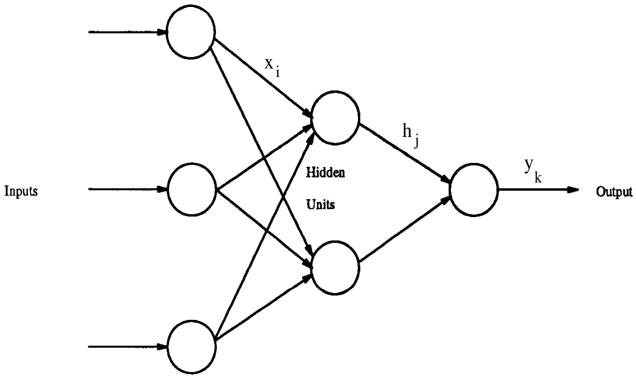
The distinctiveness analysis technique was initially developed for pruning hidden neurons, and provides ranking in which pairs of similar neurons are listed together. During pruning, most often only single neurons are removed at a time, in the process of fine-tuning the generalisation of a trained network. For eliminating inputs, however, we would wish to remove larger numbers of inputs at one time, as the elimination of a single input produces relatively little savings on the time taken to train a network. As a compromise, in this paper two inputs are removed together. Since we wish to remove more than one input, some aggregate measure is required, which is provided here by the average angle to all other inputs.

In the following section the results for both of the above forms of functional analysis, together with the aggregated rankings are provided, listing the order of significance of inputs.

To maintain comparability with the previously used magnitude measures, the three measures discussed earlier are used to provide a joint ranked list based on the individual lists, which can be assumed to be representative of such magnitude ranking techniques and less affected by the computational peculiarities of the specific measures. Thus, the input which is most important in all three lists is clearly the most important by magnitude techniques and so on.

2.4. Sensitivity analysis

Sensitivity analysis is a simple method of finding the effect an input has on the output of the network. The relationship of an input neuron *i* and an output neuron *k* is found by determining the impact that a small change in *i* has on *k*. If drastic change occurs *i* is considered to be one of the key factors in producing the current activation value of *k*.



Structure of a three layered ANN with one output

Given the above (simplified) network structure, we find the rate of change of an output neuron y_k with respect to an input neuron x_i by calculating the derivative dy_k/dx_i using the chain rule of differentiation.

$$\frac{dy_k}{dx_i} = \frac{dy_k}{dU_{k2}} \cdot \frac{dU_{k2}}{dh_j} \cdot \frac{dh_j}{dU_{j1}} \cdot \frac{dU_{j1}}{dx_i} \tag{10}$$

$$= f'(U_{k2}).f'(U_{j1}).\sum_j w_{jk}.w_{ij} \tag{11}$$

This method has also been used with the assumption that the product $f'(U_{k2}).f'(U_{j1})$ is constant for all k and j (Hora, Enbutsu and Baba, 1991). The influence of x_i on y_k could thus statically be determined from the weight matrix of the trained network. Unfortunately in all the domains we have tried, this assumption does not hold (Gedeon and Turner, 1993, Gedeon 1996a) and thus we must continue with the computationally more expensive approach of calculation the effect of perturbations.

We experimented with a range of increasingly computationally expensive methods to determine the magnitude of change in value of outputs to particular inputs. These were:

1) Perturbation of single inputs on a single pattern, all elements 0.5.
2) Perturbation of single inputs on a single pattern, all elements the actual average value for that input as calculated from the training patterns.
3) Perturbation of single inputs on all training patterns, deltas accumulated for each input.
4) Perturbation of pairs of inputs on all training patterns, deltas accumulated for each input involved in each perturbation event.
5) Perturbation of triplets of inputs on all training patterns, deltas accumulated as above.
6) Perturbation of quadruplets of inputs on all training patterns, deltas accumulated as above.
7) Perturbation of quintuplets of inputs on all training patterns, deltas accumulated as above.

model	Most significant						Least significant					
0.5	9	3	8	10	1	6	2	4	11	7	5	12
Ave.	9	3	8	2	10	6	1	7	4	11	12	5
Δ1	9	3	1	7	10	6	8	11	4	12	2	5
Δ2	9	3	7	10	8	1	6	11	4	12	5	2
Δ3	9	3	7	10	1	8	5	2	12	4	11	6
Δ4	9	3	7	10	1	8	5	2	12	11	4	6
Δ5	9	3	7	10	1	8	5	2	12	11	4	6

Clearly, the order is converging. Beyond the perturbation of quadruplets of inputs, no change is detected. This implies that the interactions between inputs is limited to four at a time. As is unsurprising, the values of the most significant inputs are clear very early, while determining which are the least significant inputs is quite expensive, and happens late. In the subsequent section, the results quoted for the significance analysis technique will be that of the Δ4/5 final version.

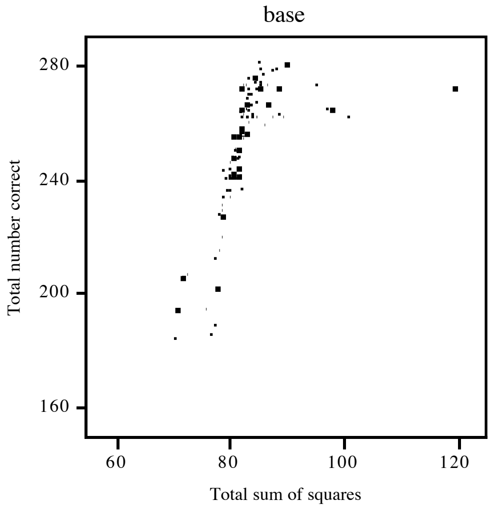
2.5. Results and comparisons

In the following table, model *I* is the distinctiveness of inputs over the labelled set of patterns, of which *C* is the aggregated form, *W* is the weight distinctiveness, of which *U* is the aggregated form.

The combined ranking for the three magnitude measures is given in the last column.

model	Most significant						Least significant					
I	11	10	3	8	4	1	6	5	2	9	7	12
C	3	10	8	11	7	1	2	9	6	12	5	4
W	9	8	6	3	11	4	2	5	10	7	12	1
U	6	4	3	2	7	5	1	10	9	8	12	11
Mag.	4	5	11	6	10	1	2	8	3	7	12	9
Sens.	9	3	7	10	1	8	5	2	12	11	4	6

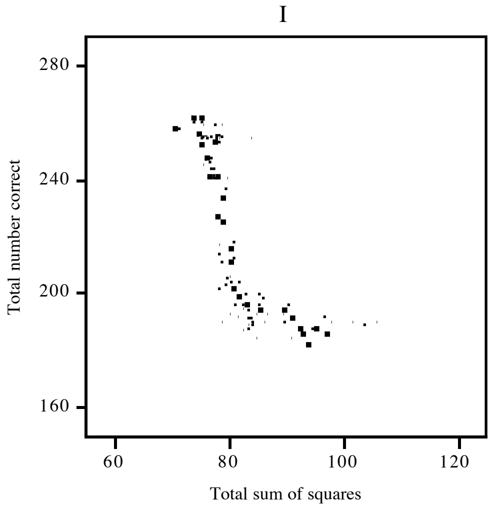
The following diagram shows the base case, using all of the inputs.



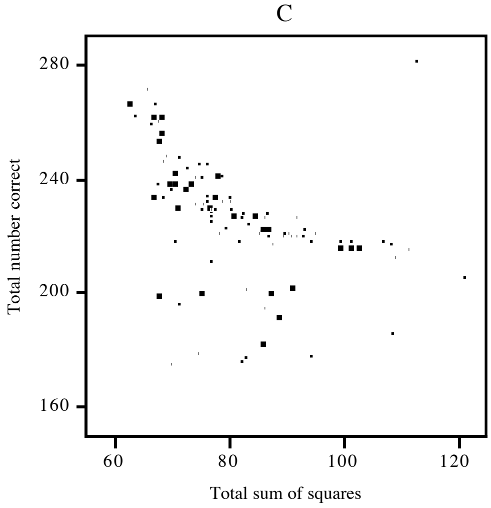
The anti-correlation of the total sum of squares (tss) value and the number of patterns correctly classified demonstrates the degree of difficulty of the classification problem for the network.

There are a number of inputs which are providing irrelevant information, and the network was trained using sum squared error measure, when the network provides a better result in terms of low tss, the number of correctly classified patterns is reduced.

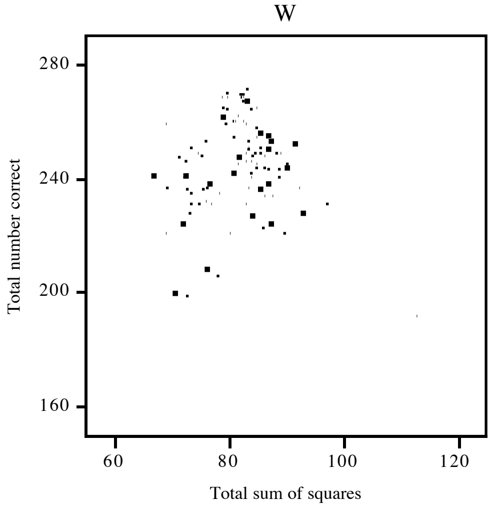
The following diagrams show the results on using the above table to eliminate some pairs of inputs. These are the top two inputs from the table, being the least significant inputs.



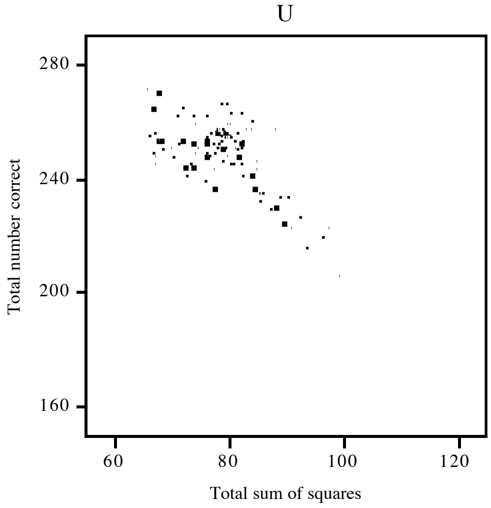
There is now a better correlation between tss and total correct, however overall the number correct has decreased indicating some significant information has also been lost.



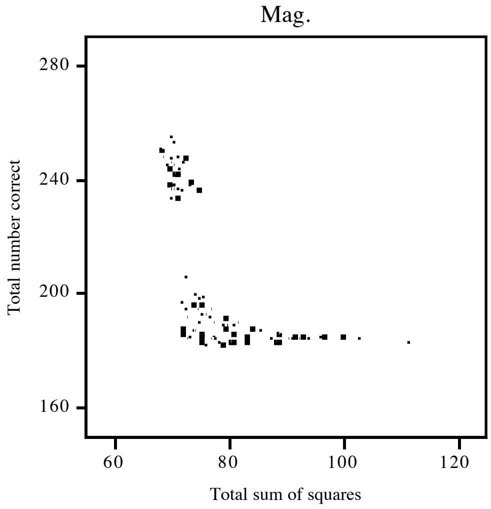
The correlation is much better, and less significant information has been lost. This demonstrates that the aggregated measure is a better predictor of significance, at least for this functional measure. The *I* and *C* measures depend only on the labelled pattern set, and input elimination using these measures has reduced the number of correct classifications overall.



This diagram has shown that the anti-correlation has been removed, and a slight correlation introduced, similar to *I*. Note that the overall number correct is significantly improved in *W* over *I*, which indicates the former is a better indicator of significance.



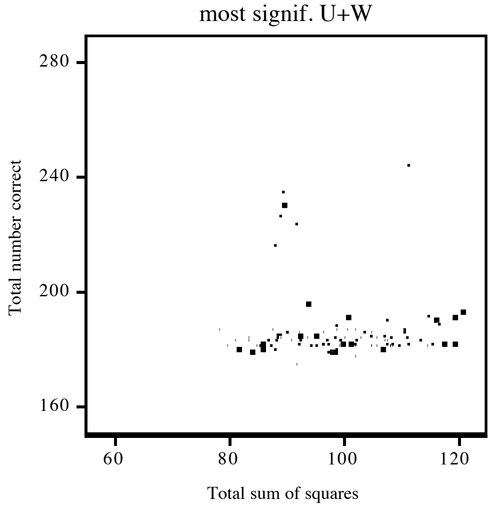
The above diagram demonstrates the correlation we hoped to attain, with a higher overall total correct classifications than the original base case. This again demonstrates the advantage of the aggregated measure. The diagram also demonstrates that the network learnt weights provide a better indication of the significance of inputs than provided by simple statistical properties of the labelled set.



The correlation is slightly improved, but there is now a discontinuity in the quality of results produced, and the number correct are overall significantly lower. This indicates that at least one of the inputs removed was significant, notwithstanding the measure. This significant input is very probably input 9, as the other input (12) is in the least significant pair in three of the functional measures used above.

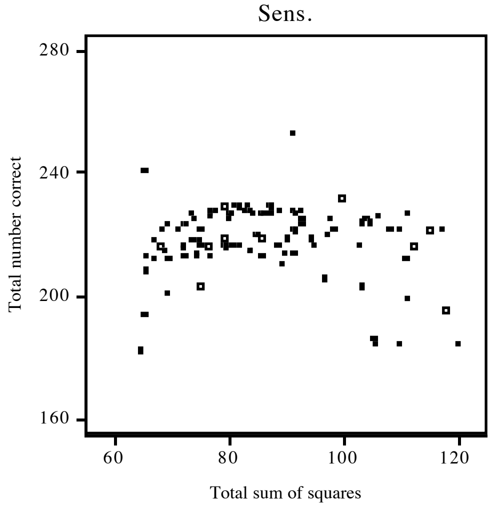
To test this, a further pair of inputs were eliminated, using the most significant inputs as determined by the *U*, and *W* measures.

The observed discontinuity possibly indicates that there is now a ‘local minimum’ which is easier to find than the best minimum that the network is otherwise able to find given its inputs.



The above diagram demonstrates that the effect of removing the two most significant inputs has a catastrophic effect on network performance. The lowest tss value is now higher, and the total number correct is no longer related to the tss value. This suggests that the number correct is now due to chance.

A further experiment was done using the sensitivity analysis measure described above to determine the pair of significant inputs to remove. These were input 9, and 3.



The diagram demonstrates that the two inputs removed are significant, in that the overall performance was reduced, and the correlation we would desire between total sum of squares and the number of patterns correctly classified is almost gone. That the inputs eliminated were not the most significant is clear from comparison with the preceding diagram, in that the overall level of correctness is still higher using the sensitivity analysis technique for selection.

3. Conclusion

In this paper a number of functional measures for determining the significance of inputs were introduced. They were contrasted with with the traditional magnitude based input significance measures in both qualitatively by means of dicussion, and quantitatively, by experimentation.

The experimental work demonstrated that the functional measures, particularly based on the analysis of the network and not just the data, produce better indicators of the significance of particular inputs, as shown exhaustively by the elimination of pairs of inputs judged to be least significant by each measure.

The effect of eliminating the two most significant inputs was to destroy network performance which serves as extra validation of the utility of functional measures introduced. The use of sensitivity analysis was also examined, demonstrating that this technique works better than the magnitude based techniques, but not as well as the functional measures. This fits with our hypothesis that as we move from the techniques based on static properties (magnitude techniques) through semi-dynamic properties (pattern perturbation for sensitivity analysis) to dynamic properties (functional behaviour over pattern presentations) we increase the reliability of the determination of significance of inputs.

References

Bustos, RA and Gedeon, TD "Decrypting Neural Network Data: A GIS Case Study," *Proceedings International Conference on Artificial Neural Networks and Genetic Algorithms (ICANNGA), 4 pages,* Alès, 1995.

Garson, GD "Interpreting Neural Network Connection Weights," *AI Expert*, pp. 47-51, April, 1991.

Gedeon, TD "Indicators of Hidden Neuron Functionality: Static versus Dynamic Assessment," invited paper, *Australasian Journal of Intelligent Information Systems*, vol., 10 pages, June, 1996a.

Gedeon, TD "Indicators of Input Contributions: Analysing the Weight matrix," *Proceedings ANZIIS'96 International Conference*, 4 pages, Adelaide, 1996b.

Gedeon, TD and Harris, D "Network Reduction Techniques," *Proceedings International Conference on Neural Networks Methodologies and Applications*, AMSE, vol. 1, pp. 119-126, San Diego, 1991.

Gedeon, TD, Li,K, Gordon, E, Manor, B and Latimer, C "Neural Network Classification of Schizophrenia using Eye Gaze Data," *Proceedings International Panel Conference on Soft and Intelligent Computing*, 5 pages, Budapest, 1996.

Gedeon, TD and Turner, H "Explaining student grades predicted by a neural network," *Proceedings International Joint Conference on Neural Networks*, pp. 609-612, Nagoya, 1993.

Haig, AR, Gordon, E, Rogers, G and Anderson, J "Classification of single-trial ERP sub-types: application of globally optimal vector quantization using simulated annealing," *Evoked Potentials*, 31 pages, 1995.

Hora, N, Enbutsu, I and Baba,K "Fuzzy rule extraction from a multilayer neural net," *Proc. IEEE*, vol. 2, pp. 461-465, 1991.

Milne, LK "Feature Selection Using Neural Networks with Contribution Measures," *Proceedings Australian Conference on Artificial Intelligence AI'95*, Canberra, 1995.

Milne, LK, Gedeon, TD and Skidmore, AK "Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood," *Proceedings Australian Conference on Neural Networks*, pp. 160-163, Sydney, 1995.

Rumelhart, DE, Hinton, GE, Williams, RJ, "Learning internal representations by error propagation," in Rumelhart, DE, McClelland, *Parallel distributed processing*, vol. 1, MIT Press, 1986.

Wong, PM, Gedeon, TD and Taggart, IJ "An Improved Technique in Porosity Prediction: A Neural Network Approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, n. 4, pp. 971-980, 1995.