THE APPLICATION OF ARTIFICIAL NEURAL NETWORKS TO THE CLASSIFICATION OF AUSTRALIAN WHEAT VARIETIES

C.C. Fung, T.A. Vuori, N.R. Belford, W.A. Fakhri and D.G. Myers
Curtin University of Technology
GPO BOX U1987, Perth, Australia

ABSTRACT

This paper reports the results obtained from the application of artificial neural networks to the Australian wheat variety classification problem. A 'HyperSAB' network with a self adaptive acceleration strategy for the error back propagation learning has been developed. This has been applied to six different Australian wheat varieties with 200 samples in each case. The results indicate that the artificial neural network has some potential to be used as an identification tool in this problem.

INTRODUCTION

Traditionally, chemical analysis is the only way to make definitive classifications of wheat varieties. This process however is lengthy and expensive with very low throughput. Alternative methods are keenly sought to improve the productivity and quality assurance process. One of these alternatives is the application of pattern recognition techniques[1]. By utilising image processing and conventional statistical methods, considerable progress has been achieved[2-4]. However, when the number of grain varieties is high, a better solution to the problem is required. This leads to the investigations and the development of the artificial neural network (ANN) approach.

Artificial neural networks have been applied extensively to the problem of pattern recognition[5]. Initial works in this project have utilised the feed forward neural network which is based upon the error back propagation algorithm due to Rumelhart, Hinton and Williams [6]. Although it demonstrated the potential of the ANN approach, the training time can be excessively high. To overcome this problem, some forms of acceleration strategy must be adopted. Examples are the 'Self Adaptive Back Propagation' (SAB) paradigm proposed by Devos and Obans[7], and the 'SuperSAB' approach developed by Tolleneare[8]. However, when these algorithms are applied, problems such as instability and failure to converge are observed. Further research works led to the development and implementation of the 'HyperSAB' algorithm which is genetically linked to the earlier paradigms. In the following sections, the pattern recognition approach to the classification problem and the definitions of the data features are described. It is then followed by the development of the HyperSAB algorithm and application results are presented.

PATTERN RECOGNITION APPROACH

The pattern recognition approach to the classification problem is shown in Figure 1 below.

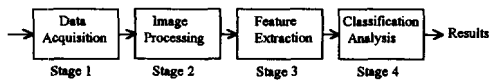


Figure 1: Pattern recognition approach to classification

Descriptions of the hardware and software setups in stages 1 to 3 of the above diagram have been detailed in reference 3. In the paper, statistical methods have been used in stage 4. Two of the features [3,4] extracted from stage 3 are used in the present project. They are defined as follows.

1. Ray parameters

Rays are derived by measuring the distance from the centroid to the edge of the grain, along a line segment inclined at a specified angle to the major axis. The ray parameter set is distinctly defined by a specified angle interval. Each ray parameter is normalised to one of the rays known as the normalising ray. Selection of the normalising ray is arbitrary since the results shown that the choice of this parameter has little effect on the system performance. Illustration of the ray parameters is shown in Figure 2.

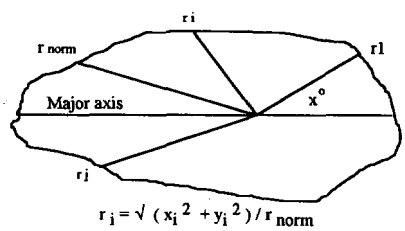
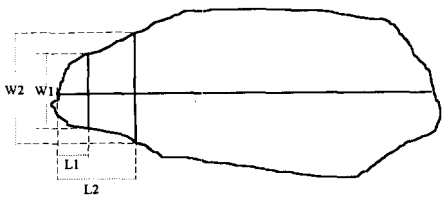


Figure 2: Definition of ray parameters

2. Aspect ratios

Aspect ratios are derived by measuring the lengths of several chords which are perpendicular to the grains major axis and dividing each of these lengths by the distance from the germ end of the grain to the point where the chord intersects the major axis. This is shown in Figure 3 below.



$a_i = L_i / W_i$; i = 1 , 2 ... n

Figure 3: Definition of aspect ratios

HYPERSAB DEVELOPMENT

1. Error back propagation (EBP) network

The feed forward error back propagation network (EBP) as described by Rumelhart et.al. [6] comprises of a number of layers of processing units described as 'neurons'. Neurons are characterised by two variables; an activation function and a threshold. The sigmoidal activation function as shown in equation 1 is commonly used.

$$v_j = \frac{1}{1+e^{(-\beta \phi_j)}} \qquad (1)$$

where v is the activation at neuron j with the a value of potential ϕ and β controls the steepness of the activation function.

The network is comprised of 'synapses' connecting the layers of neurons and the input potential, ϕ for a neuron is defined as

$$\phi_j = \sum_i w_{ij} v_i + \tau_j \qquad (2)$$

where w is the weight of the synapse between neuron-i and neuron-j, v is the activation state of neuron-i, and τ is the threshold of neuron-j.

The original training technique, error back propagation (EBP), is based on the comparison between the output pattern produced by the forward pass and the target values due to a specific given input pattern, p. The absolute differences are collected in a summation function and an error, E can be calculated. Beginning with each weight at the output nodes and working back to the weights at the input layer, a gradient is calculated. This process is described as back propagation of error. The gradients are then summed for each weight over the combinations of all the input output patterns and as a function of the resulting gradient, the weights are then updated.

$$\Delta w_{ij}^n = -\eta \sum_p \frac{\partial E_p}{\partial w_{ij}} \qquad (3)$$

In this weight update rule, the index n refers to the n^{th} iteration in the process and η describes the learning rate, being a discrete step size. This procedure is iterated until error E for all outputs is within a predetermined tolerance, or until a predefined number of iterations is reached. It is effectively a gradient descent performed on the energy surface [6]. The dimension of this surface equals to the number of weights in the network and the energy is the sum of E for all patterns.

Common to all steepest descent methods is the problem of the choice of step size. A large value for η will induce rapid learning, but it will also lead to oscillations and instability. In these circumstances, the network may fail to converge. On the other hand, a small step size will result in slow convergence and it may be trapped in a local minimal. A 'momentum term' has been included by Rumelhart et.al. [6] and it changes the weight update rule as follows.

$$\Delta w_{ij}^n = -\eta \sum_p \frac{\partial E_p}{\partial w_{ij}} + \alpha \Delta w_{ij}^{n-1} \qquad (4)$$

where α is the momentum term in the range from zero to one.

The effect of this is to magnify the learning rate for flat regions of weight space or across local minimal; while in steep regions, movement is focused downwards by damping the oscillations caused by the alternating sign of the gradient.

2. Self adaptive back propagation

Research works in the area of local acceleration strategies proposed a series of heuristics attempting to increase both the rate of convergence and the accuracy of that convergence. These heuristics are summarised as follows [8].

A. Individual step size should be assigned for each weight.
B. The step sizes should be made variable.
C. If the derivative of a weight maintains the same sign for two consecutive steps, the learning rate of that weight should be increased so as to accelerate the descending process.
D. When the derivative of a weight changes sign, the learning rate should be decreased.

It is considered that strategies conforming to these heuristics that are stable will represent an improvement over the original Rumelhart model. In implementation though, several problems are likely to occur with the main focus of attention on the manner in which the learning rate is increased and decreased. Furthermore, each local change will affect the total system. While in general this will not be precipitous, it may occasionally be so such that the effect at best is an increase in the time to convergence and at worst a failure to converge.

Tolleneare [8] sets out a learning rule known as 'SuperSAB' which is based on Devos and Oban's [7] self adapting back propagation (SAB) strategy. The essence of the algorithm is to allow modification of the learning rate, η, according to the value and the sign of the weight derivative.

If the increase factor is η_+ and the decrease factor is η_- for the step size, and η_{start} be the initial value for η_{ij} for all i, j. The SuperSAB algorithm is outlined as follows [8]:

A. Initialise all η_{ij} to η_{start} ;
B. carry out error back propagation with momentum;
C. for each w_{ji}, as long as the w_{ji} derivative does not change sign, set $\eta_{ij}^{n+1} = \eta_+ * \eta_{ij}^n$;
D. if the w_{ji} derivative changes sign:
undo the previous weight update (which caused the change in gradient sign) by using $\Delta w_{ij}^{n+1} = -\Delta w_{ij}^n$ instead of equation (3) ;
set $\eta_{ij}^{n+1} = \eta_- * \eta_{ij}^n$;
put $\Delta w_{ij}^{n+1} = 0$; and this will cause the next invocation of step B ignore the momentum effect due to the previous step.
E. repeat steps B to D until stopping criteria is met.

The implementation and application of this strategy to the wheat classification problem however have experienced the previously mentioned shortcomings. This subsequently leads to the development of the HyperSAB.

3. HyperSAB

When the SuperSAB algorithm is examined, it is considered that the undoing of the previous weight update as described in step D above is unnecessary as this may not be responsible for the change in the gradient sign owing to changes elsewhere in the network. For any value of η, there may be a change in the gradient sign, and this change for the corresponding weight may be significant in optimising the search for the minimum on the complete energy surface. SuperSAB also allows any weight to grow indefinitely if no change in gradient sign is detected and Tolleneare [8] is unsure as to the effect of this on real world data. Experience with the wheat variety classification problem shows a failure to converge as the most likely result of this scenario. It is postulated that these weight growths allow elements in hypervolumes to travel so contrary to the collective descent on the energy surface that an irrevocable instability arises. The most obvious solution to this problem is to place an upper threshold, η_{up}, on η. When η exceeds this threshold, it increases at a rate of η_+. This is defined in the following expression.

$$\begin{array}{ll} \text{for } \eta^{n+1} < \eta_{up} & \eta^{n+1} = \eta_+ * \eta^n, \\ \text{else} & \eta^{n+1} = \eta_{up} \end{array} \qquad (5)$$

A further difficulty with SuperSAB is what action if any should be taken as η approaches zero while under the influence of η_-. The solution to this is to provide a minimum threshold, η_{lo}, for η similar to the upper threshold. For this minimum threshold, it might be put that η is *optimally small* such that it is *meaningfully large*, both in absolute terms for the current iteration, and in its ability to grow in the near future iterations. The above definition of the minimum threshold for η_{ij} might well be applied to η_{start} although there is no requirement that they have to be equal. In addition, the reduction of η is also made larger than η_+ as a way of more rapidly returning to an acceptable step size. This can be defined as shown below.

$$\begin{array}{ll} \text{for } \eta^{n+1} > \eta_{lo}, & \eta^{n+1} = \eta^n/\eta_- \\ \text{else} & \eta^{n+1} = \eta_{lo} \end{array} \qquad (6)$$

While observing the actual changes in the weights during execution of the network, an additional heuristic is developed. When η is equal to η_{lo}, it became apparent that it is necessary to revert the sign of the weight change back to the value obtained during the previous iteration. This enables a continuation of the downward trend without trapping the value of η in the minimum threshold.

The HyperSAB paradigm is now described as follows.

A. An initial η_{start} is assigned for all η_{ij} ;
B. carry out error back propagation with momentum.
C. At the n^{th} iteration, for each weight, the Δw_{ij}^n is calculated and the sign is compared to that of the Δw_{ij}^{n-1} ;
(a) if the sign is unchanged:
η^{n+1} is obtained from expression (5), current Δw_{ij}^n is used to calculate w_{ij}^{n+1} from ($w_{ij}^n + \Delta w_{ij}^n$).
(b) if the sign is changed, set $\Delta w_{ij}^n = 0$, thus $w_{ij}^{n+1} = w_{ij}^n$; and the effect of the momentum term in equation (4) will be eliminated in the next iteration;
η^{n+1} is obtained from expression (6);
(i) if $\eta^{n+1} < \eta_{lo}$, the sign of the Δw_{ij}^n is maintained and to be used in the next iteration;
(ii) if $\eta^{n+1} \geq \eta_{lo}$, the sign of Δw_{ij}^{n-1} is used in the next iteration;
D. repeat steps B to C until stopping criteria is met.

APPLICATION RESULTS

The HyperSAB network has been implemented in C programming language. The computing environment has been a cross platform collaboration of a PC-386 and Hewlett Packard series 700 work station. The algorithm has been applied to 6 different Australian wheat varieties with 200 samples in each case. The network implemented has been tested with a range of 8 to 32 input nodes, and a hidden layer with number of nodes varies from 20 to 80. Due to the diversity of parameters available within the feature sets and the number of independent variables in the neural network framework, it is difficult to fine tune the structure for optimum performance. From the results obtained, they do indicate that a useful wheat classification system can be developed on the HyperSAB framework.

Concerning the parameters used by the network, the values of 5 and 1.5 have been assigned to η_{up} and η_+ respectively. On the other hand, the figures 2.5 and 0.001 are found to be most suitable for η_- and η_{lo} respectively. The momentum term used is 0.001. The training process will stop when the system error is less 0.01.

1. Definition of correctness

To evaluate the performance of the networks, what constitutes a 'correct' output must be defined. Two approaches have been adopted.

A. Average correct classification (acc) percentage. For the six different varieties under investigation, a number of data sets are extracted. They are arranged into a number of groups. The percentage results of the correctly classified data are then averaged over the number of groups.

B. Winner takes all. For any given input set of samples, the output with the highest percentage of correctness is selected to represent the appropriate variety. In practical application this is usually the answer to the question being that whether the sample is a particular variety or not.

2. Comparison of training time

Table 1 compares the number of iterations for the two networks during training. Where the number of dimensions is low, the training time for both is comparable but the EBP network deteriorates at four varieties and it fails to perform within a reasonable time above 5 varieties.

Table 1: Number of iterations during training

	Number of Iterations	
varieties	HyperSAB	EBP
2	104	62
3	282	260
4	860	1294
5	1185	5959

3. Hidden Layer Analysis

Increasing the number of nodes in the hidden layer will cause a corresponding increase in the number of dimensions of the network. However, this also results in a linear rise in the number of floating point calculations. A range of number of nodes from 20 to 80 has been attempted. Results from a data set of 4 varieties of 28 ray parameters are illustrated in Table 2.

With respect to training time, there is a dramatic fall when the number of hidden nodes is increased from 20 to 35. The trend continues until 55 nodes. The average correct classification (acc) also improves until 60 nodes and then falls slightly. Judging from these figures, the number of 60 nodes is assumed to be optimum for the present set of data.

Table 2: Results from 28 ray parameters, 4 varieties

Nodes	20	25	30	35	40	45	50
Time (min)	407	395	135	59	55	56	47
Iterations	20001	15549	4283	1601	1329	1215	919
% acc	43	49	45	51	47	53	49
Nodes	55	60	65	70	75	80	
Time (min)	48	53	56	47	65	53	
Iterations	806	819	826	647	813	630	
% acc	50	57	49	51	54	50	

4. Winner takes all approach

These results demonstrate the performance of the neural network when it operates on a yes or no decision basis. An example of the results from the data set of aspect ratios for two to six wheat varieties is shown in Table 3 below. With the exception of the 5 varieties' classification, all the other results are correct.

Table 3: Results from winner takes all approach

No. of	correct variety wins					
varieties	var1	var2	var3	var4	var5	var6
2	yes	yes				
3	yes	yes	yes			
4	yes	yes	yes	yes		
5	no	yes	yes	no	no	
6	yes	yes	yes	yes	yes	yes

During the course of tests, a problem became apparent that the network could be trained exactly to the training samples but not generalised for the class of data. Hence a less accurately trained network may classify more accurately on the unknown testing data. This implies either the training sample is too small or the selection of more pertinent feature parameters is required.

CONCLUSIONS

Investigation into the application of artificial neural network to the Australian wheat variety classification problem has been carried out and reported in this paper. A feed forward error back propagation network with self adaptive acceleration strategy is developed. Test results show that the developed HyperSAB network is stable and it converges much faster than a network based on the original error back propagation network. It is concluded that HyperSAB has some potential as a general purpose neural network algorithm in wheat variety recognition. However, optimisation of the network structure, feature parameters and sample size require further research.

REFERENCES

1. Neuman, M., Sapirstein, H.D., Shwedyk, E. and Bushuk, W.: 'Discrimination of wheat classes and varieties by digital image analysis of whole grain samples', Journal of Cereal Science, No. 6, (1987), 125-132.
2. Myers, D.G. and Edsall, K.J.: 'The application of image processing techniques to the identification of Australian wheat varieties', Plant Varieties and Seeds, No. 2, (1989), 109-116.
3. Barker, D.A., Vuori, T.A., Hegedus, M.R. and Myers, D.G.: 'The use of ray parameters for the discrimination of Australian wheat varieties', Plant Varieties and Seeds, No. 5, (1992), 35-45.
4. Barker, D.A., Vuori, T.A. and Myers, D.G.: 'The use of slice and aspect ratio parameters for the discrimination of Australian wheat varieties', Plant Varieties and Seeds, No. 5, (1992), 47-52.
5. Pao, Y.H., 'Adaptive pattern recognition and neural networks', Addison Wesley, Reading, Massachesetts, (1989).
6. Rumelhart, D.E., Hinton, G.E. and Williams, R.J.: 'Learning internal representations by error back propagation' in Rumelhart D.E. and McClelland J.L. (Eds): 'Parallel distributed processing: Explorations in the microstructure of cognition', (Ch. 8), Cambridge MA: MIT Press, (1986).
7. Devos, M.R. and Orban, G.A.: "Self Adaptive back propagation", Proceedings NeuroNimes, EZ, Nanterre, France, (1988).
8. Tollenaere, T.: "SuperSAB: Fast Adaptive Back Propagation with Good Scaling Properties", Neural Networks, Vol. 3, (1990), 561-573.