Effects of Learning Parameters and Data Presentation on the Performance of Backpropagation Networks in Milk Yield Prediction

F. Salehi, R. Lacroix, and K.M. Wade

Introduction

In recent years much attention has been given to using backpropagation neural networks to solve real-world problems in the field of agriculture. In building a backpropagation network, various architectures, learning rates, learning rules, momentum and the method of presentation of the input data may be used. Since, in employing neural networks, the user is not expected to understand thoroughly their internal functioning, it is a common practice to use the default configurations, including learning parameter values, provided by commercially available softwares. However, in order to optimize learning, convergence speed, and predictive ability, it is, at times, necessary to adjust some or all of these parameters.

While variations in performance due to modifying the net architectures were frequent, few studies have considered the effect of other factors on the learning of neural nets. Further, there are conflicts about the suggested values and ranges of learning parameters among those who have included them in their reports. This lack of information, along with the existence of conflicts concerning some parameter values, suggests that the effects of different factors on the performance of the neural networks should be investigated. The results of such investigations could lead to application-dependent guidelines for choosing appropriate methods and parameter values which would improve network generalization ability and its learning speed.

Objectives

The objective of this study was to investigate the effect of different values of learning rates in the hidden and output layer, momentum, and epoch size on the performance of a backpropagation network in predicting milk yield. In addition, the effect of a bipolar method of input data presentation was investigated.

Experimental Procedures

Backpropagation neural networks were used to predict milk yield from test-day production data of dairy cattle. The data files consisted of 16 input and 3 output variables (Table 1). The networks used a hyperbolic tangent transfer function, normalized-cumulative delta-rule learning rule, and consisted of three layers, with 10 processing elements (PE) in the hidden layer. The data were provided by the Quebec Dairy Herd Analysis Service (PATLQ), and consisted of individual Holstein records for milk, fat, and protein, for the period 1979-1992. From the principal data set two subsets were constructed; one for training, and one for testing the network. The training and testing data sets contained 8,867 and 31,263 records, respectively.

The modifications of the learning parameters included four levels of learning rate in the hidden and output layers, and momentum, three levels of epoch, and a binary/ bipolar input data. A combination of bipolar input presentation with two epoch sizes was also tested in order to study the combined effect. The modifications were made one at a time, while all preset values of the other parameters were kept constant, as configured by the software. The number of training cycles for all simulations was 150 000. In order to ensure uniform performance, and test the dependence of the results on the initial weights, learning and testing of each network was repeated with ten different sets of initial weights.

The criterion of learning performance, was the root-mean square (RMS) between network outputs and observed values in the training file.

Results and discussion

Figure 1 depicts the distribution of the RMS errors in predicting milk yield for the repetitions corresponding to each of the modified learning parameters. This measure is also shown for a bipolar data presentation and its combination with an epoch of 4. In this illustration each column of points represents the RMS errors computed for 10 repetitions. The extent of variations, as measured by the difference between the maximum and minimum RMS values within each 10 repetitions, was quite similar in all cases except for when a bipolar data presentation was combined with an epoch of 4. The repeated results showed that each network produced consistent predictions, although outliers were detected in some cases. However, the small variations in the repetition results, along with the presence of outliers, indicate that network predictions are affected by the initial random weights selected during the learning stage. Therefore, in practice it may be helpful to reinitialize and run the network several times, in order to ensure consistent performance and avoid biased results.

With regard to the learning rate in the hidden layer the predictions were quite consistent, within each group of ten repetitions. Comparing the results of the 3 modified values of hidden layer learning coefficient to those obtained with the software default value (0.3) revealed that the modified levels did not cause a significant change in network RMS. However, values of 0.6 and 0.9 tended to produce smaller errors, while 0.1 tended to give slightly larger errors than the default value of 0.3. This suggests the possibility of insufficient weight changes due to the small value of the learning coefficient. However, the results obtained from the learning rates of 0.6 and 0.9, were quite similar. This indicates that in both cases the necessary weight adjustments were appropriate.

Increasing the value of the output layer learning rate, within the indicated range, tended to degrade the network performance. Additionally, as the learning rate in the hidden layer was increased the RMS tended to drop, whereas a similar modification in the output layer shifted the RMS upward. A comparison of the results from modifications of the learning

The results from several repetitions

coefficient in the hidden and output layer suggested that the extent of variations in each group of 10 repetitions in the output layer were more pronounced than the hidden layer.

Despite variations within the repetitions the trends from modifying the momentum value suggest that this parameter had little influence on network performance. This parameter is mainly used to speed up the learning process, specially when a small learning rate is used.

The variations of RMS, obtained from different epoch sizes, show consistent results within each set of 10 repetitions. Although one outlier was noticed in repetitions with an epoch of 4, the patterns suggested that the results obtained with epochs of 4 and 64 were significantly different from those of the default epoch size (16). However, the results from an epoch of 30 were similar to those of the networks using default configuration. Comparing the results of the epoch sizes of 4, 30, and 64 to each other revealed considerable differences among the three groups. Excluding the one outlier in the group corresponding to an epoch of 4, the RMS errors for this epoch size were the smallest, while those of an epoch of 64 were the largest among all groups. This behaviour was also observed during the training stage. This indicated that the epoch size inversely affected the learning ability of the network. The loss of network precision due to large epochs was attributed to the effect of averaging the error over a large number of data patterns. This smoothing effect may have prevented the corresponding weight changes from considering the key trends or patterns of less frequently occurring records effectively. However, it must also be noted that the averaging effect prevented global error increases due

of a network revealed that the initial

to particular data records that were completely different from the rest, or perhaps erroneous.

Considering the results from a bipolar data presentation and its combination with two modified epoch sizes, as compared with the software's default set-up, the cluster of RMS values in each group of 10 repetitions showed that a bipolar data presentation resulted in a small effect of initial weights on the results. Compared with the default binary input set-up, networks that had their data mapped in the bipolar range (-1 to +1) produced smaller RMS values, for an equal epoch size. When an epoch four times larger than the default size (16) was selected along with a bipolar input presentation, the RMS values were similar to those of the default network (Figure 1). The RMS values in Figure 1 also revealed that using either a bipolar data presentation or an epoch of 4 had similar effects on the results. Combination of bipolar data and an epoch of 4 did not show any improvement of either individual set of results.

Impact

The results of this study demonstrate that the default values and choices of a software do not always produce the best results. Therefore, it is important to adjust the learning parameter values and method of data presentation, in order to optimize network results. In this study, the most notable influences came from different epoch sizes and input data presentation. Also, the learning rate values in the hidden layer affected the network performance more than those in the output layer. The effect of different values of momentum on the performance of the networks was negligible.

weights could affect the results; thus

a network's performance should be evaluated on the basis of several repetitions. In order to optimize network performance, the results obtained in this study, suggest that users of artificial neural networks should pay attention to the values and sizes of different learning parameters and the method of data presentation.

Table 1. Variables contained in the experimental data set

Data set	
Input variables	
1. Days in milk (d)	11. Lactation number
2. Age at calving*	12. Weight of cow (kg)
3. Season of calving*	13. Logarithm of somatic cell count
4. Test day milk production (kg)	14. Energy fed on test day (mcal)
5. Test day fat production (kg)	15. Protein fed on test day (kg)
6. Test day protein production	16. Dry matter fed on test day (kg)
7. Cumulative milk production (kg)	
8. Cumulative fat production (kg)	**output variables**
9. Cumulative protein production (kg)	1. 305-d milk production (kg)
10. Average 305-d milk production for herd (kg)	2. 305-d fat production (kg)
	3. 305-d protein production

* Binary variables

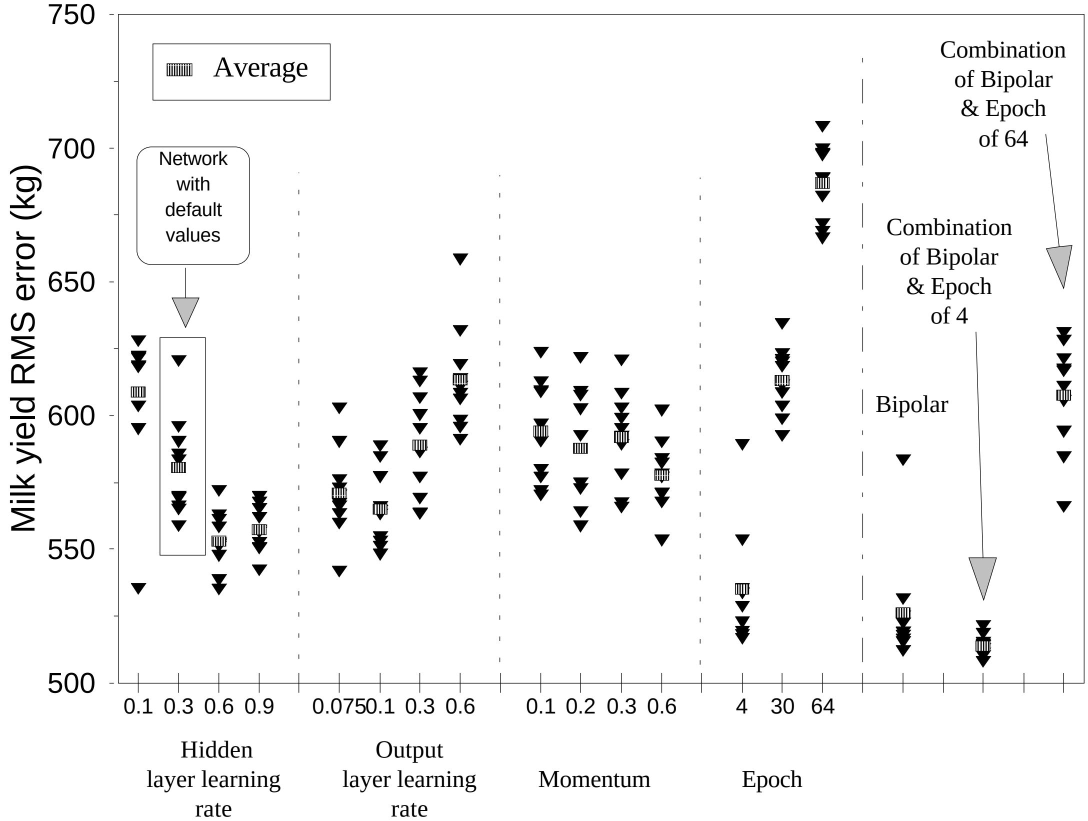

Figure 1. Distribution of the RMS error resulting from different learning parameter values, repetitions, epoch, size, and data presentation method.