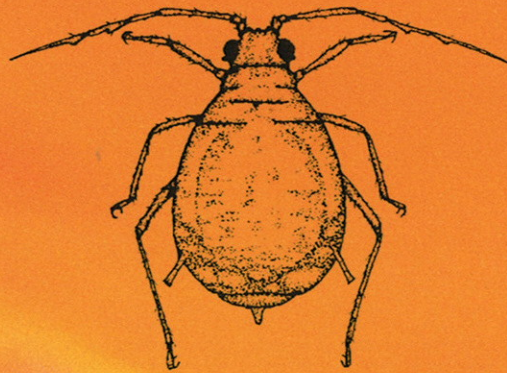


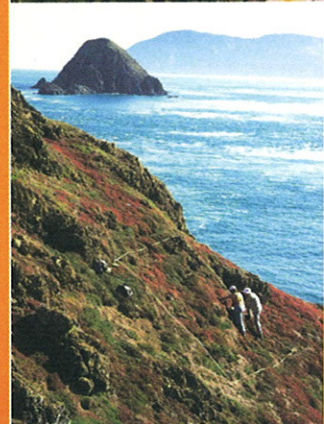
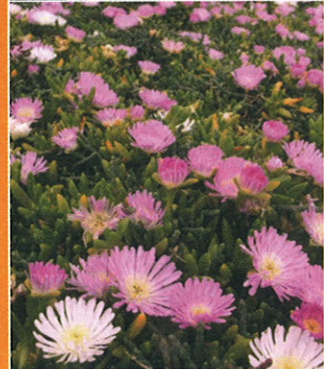
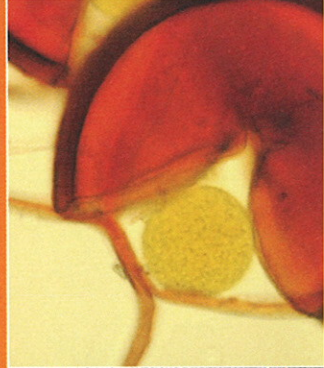
Bio-Protection & Ecology Division

Using Multi-Layer Perceptrons to Model the Lincoln Aphid Data Set

Michael J. Watts and S.P. Worner



Bio-Protection & Ecology Technical Report No. 1



CHRISTCHURCH
NEW ZEALAND
www.lincoln.ac.nz



Lincoln
University
Te Whare Wānaka o Aoraki

ISSN: 1177-7435 ISBN: 0-86476-176-7 978-0-86476-176-7

Using Multi-Layer Perceptrons to Model the Lincoln Aphid Data Set

Michael J. Watts and S.P. Worner

wattsm2,worner@lincoln.ac.nz

Chapter 1

Introduction

Backpropagation of errors (BP) trained Multi-Layer Perceptrons (MLP) (Rumelhart et al., 1986) have previously been applied to the problem of predicting pest insect numbers (Worner et al., 2003), especially aphids (Lankin, 2002). This document is the initial report on a systematic approach to the application of MLP to the aphid prediction problem.

The aims of this initial work are three-fold:

1. To investigate the effectiveness of a particular representation of the data.
2. To identify the approximate optimal topology for MLP applied to this problem.
3. To identify the approximate optimal training parameters for MLP applied to this problem.

Therefore, a methodical and rigorous approach has been taken, backed up by a sound statistical analysis of the results.

Chapter 2

Method

2.1 Description of the Data Set

The original source of the data is Crop and Food at Lincoln, Canterbury, New Zealand.

The raw data set consists of weekly measurements at the sampling station of several climate variables and weekly counts of the aphids caught in a suction trap. The aphid count as performed at the end of each week, while the climate variables were measured continuously throughout that week.

The following climate variables are present in the data set:

- average rainfall in millimetres
- cumulative rainfall in millimetres
- wind run in kilometres per day
- maximum air temperature in degrees Celsius
- minimum air temperature in degrees Celsius
- mean air temperature in degrees Celsius
- d-days is the cumulative temperature for the week
- grass temperature in degrees Celsius
- soil temperature at 100 centimetres below ground, in degrees Celsius
- Penman potential evaporation in millimetres
- potential deficit of rainfall, as accumulated excess of Penman over rainfall
- vapour pressure in hecto Pascals
- solar radiation in megajoules per square metre

2.2 Analysis of the Data

A simple analysis of the data was performed, and consisted of visualising the time-series plots of each of the variables in the data set. The plots generated are presented below. Figures 2.1 and 2.2 present the plots of the average and cumulative rainfall variables, respectively. While there are spikes present in both sets of data, they are not as significant as those found in other variables, such as the aphid count.

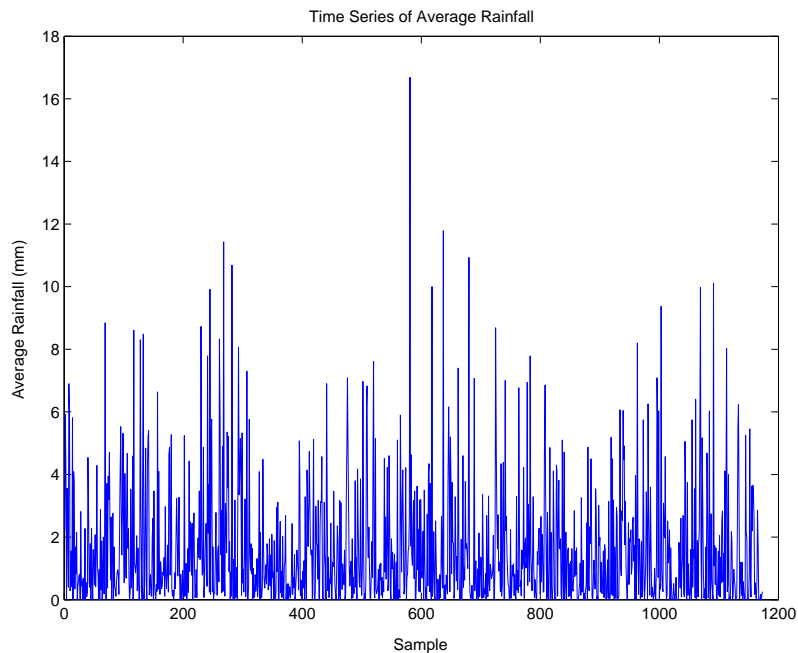


Figure 2.1: Time series plot of average rainfall (mm).

The variables wind run, maximum, minimum, and mean air temperature, and d-days (Figures 2.3, 2.4, 2.5, 2.6 2.7) all follow the same cyclic pattern, without any spikes or anomalous measurements being apparent, as do the grass and soil temperature variables, displayed in Figures 2.8 and 2.9.

The plot of the Penman potential evaporation (Figure 2.10) has some spikes, this however does not seem to be due to errors in the data set and so does not require correction.

The plot of the potential rainfall deficit (Figure 2.11) shows a large rise in the tail that is very much of proportion with the rest of the plot. However, this is an upward trend and does not appear to be due to errors in the data set: rather, it is due to a climatic trend. This large tail, however, has the potential to disrupt the modelling of the data set and must be dealt with.

Figure 2.12 plots the vapour pressure variable. Again, there are some small spikes, but these are not large enough to warrant concern or correction.

Plotting the solar radiation variable in Figure 2.13 reveals some errors in the data. While it is obvious that the solar radiation measure can never yield negative values, the plot clearly shows that there are such present in the data. These values were corrected, which yielded the plot in Figure 2.14.

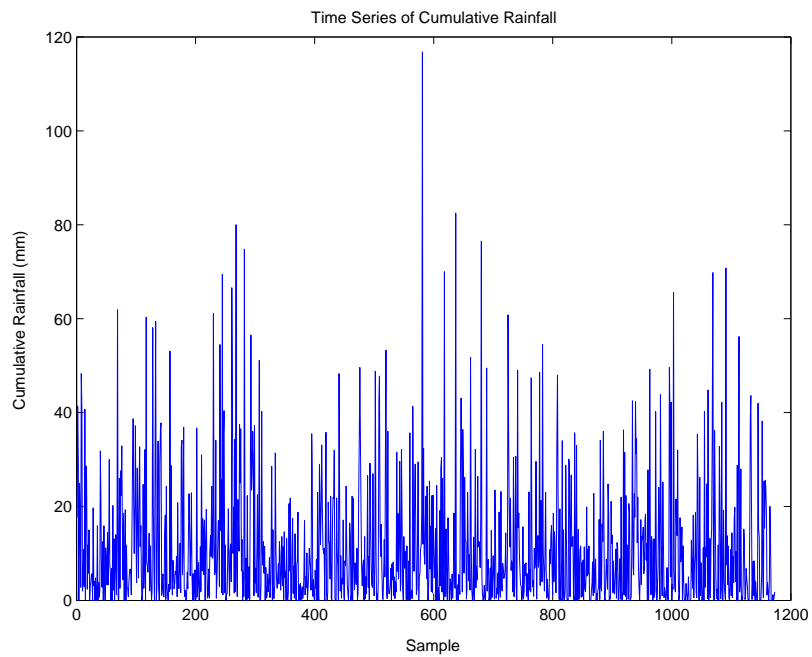


Figure 2.2: Time series plot of cumulative rainfall (mm).

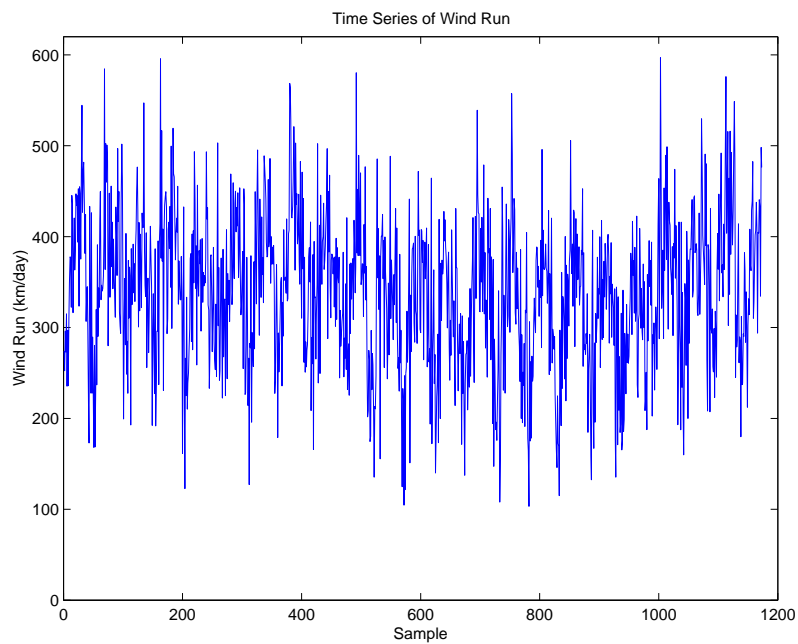


Figure 2.3: Time series plot of wind run (km/day).

The final plot is of the aphid catch (Figure 2.15). This clearly shows that there are several spikes in the number of aphids caught, which are out of proportion with the rest

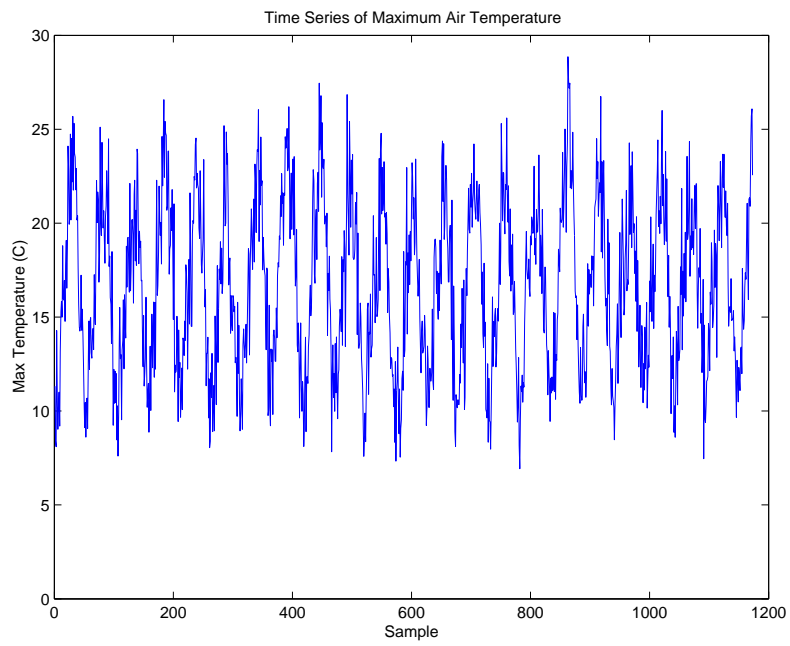


Figure 2.4: Time series plot of maximum air temperature (C).

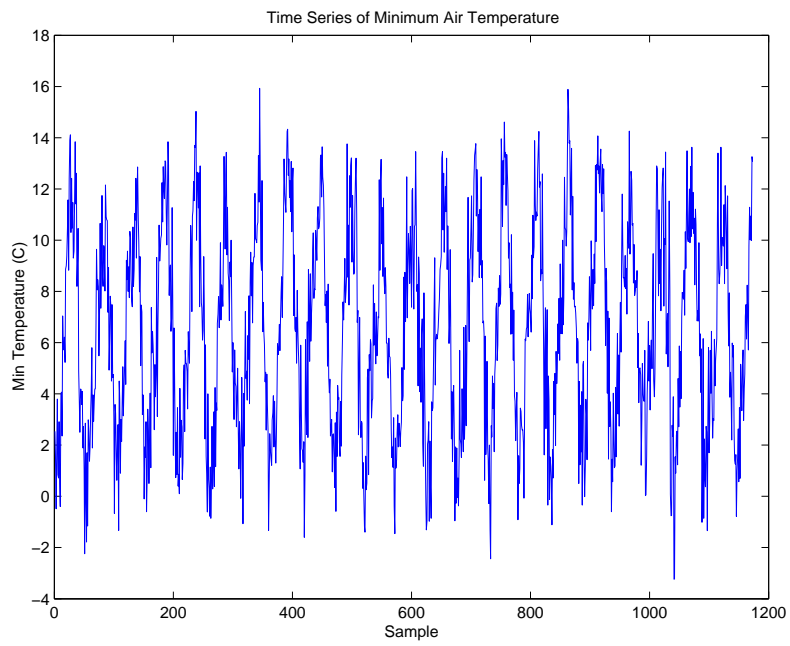


Figure 2.5: Time series plot of minimum air temperature (C).

of the data set. This disproportionality requires attention in the processing of the data.

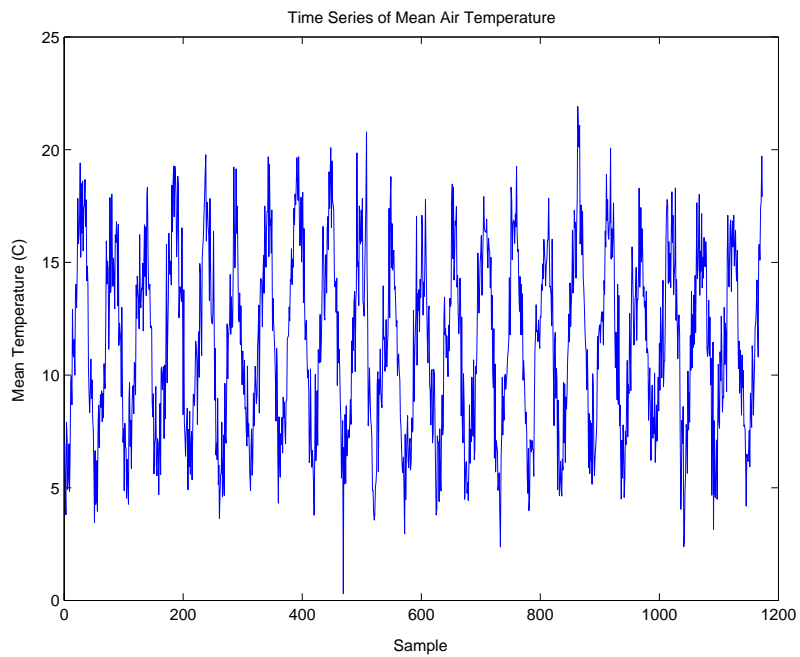


Figure 2.6: Time series plot of mean air temperature (C).

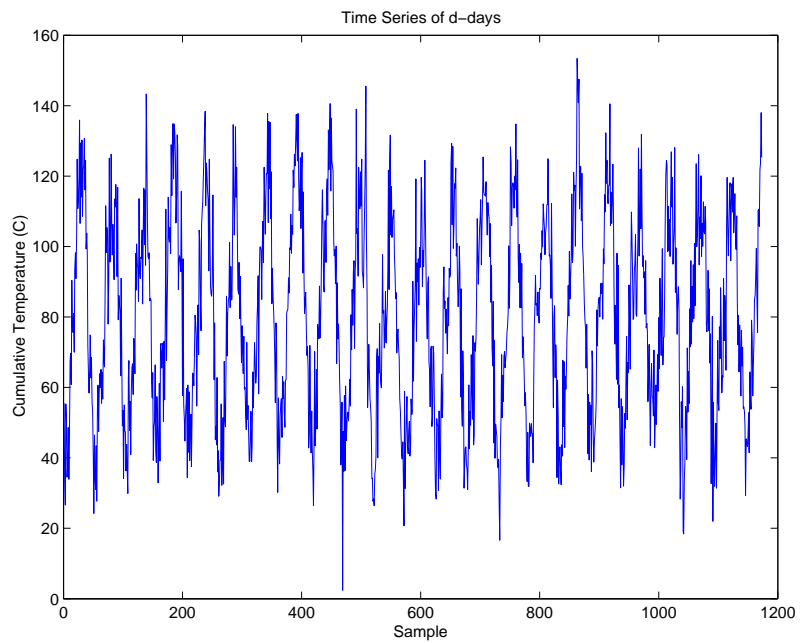


Figure 2.7: Time series plot of d-days variable (C).

2.3 Preparation of the Data for Presentation to the MLP

To correct for the large tail in the potential deficit plot (Figure 2.11, the base-ten logarithm was taken of all non-zero values of this variable. The results of this transforma-

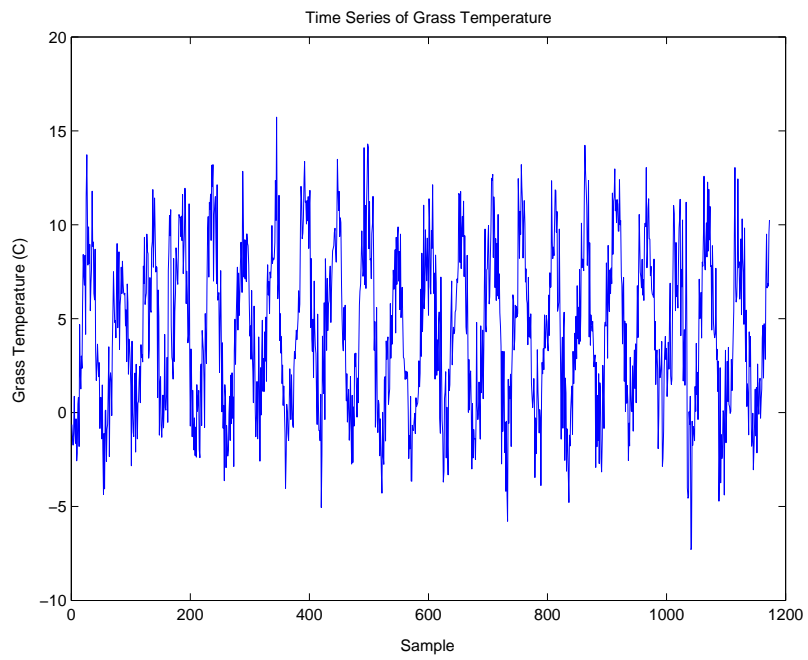


Figure 2.8: Time series plot of grass temperature (C).

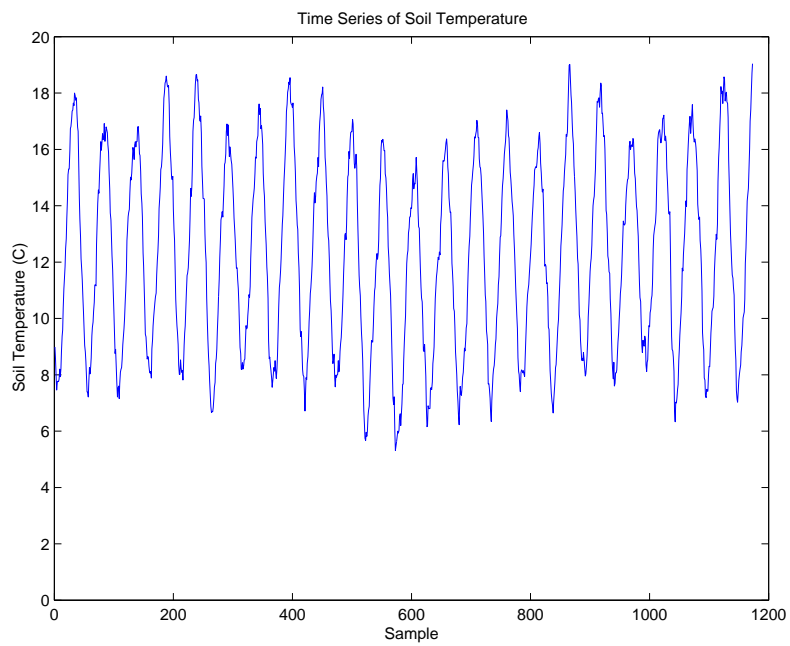


Figure 2.9: Time series plot of soil temperature at 100 cm depth (C).

tion are displayed in Figure 2.16.

Again, to correct for the large spikes in the number of aphids observed, the base-

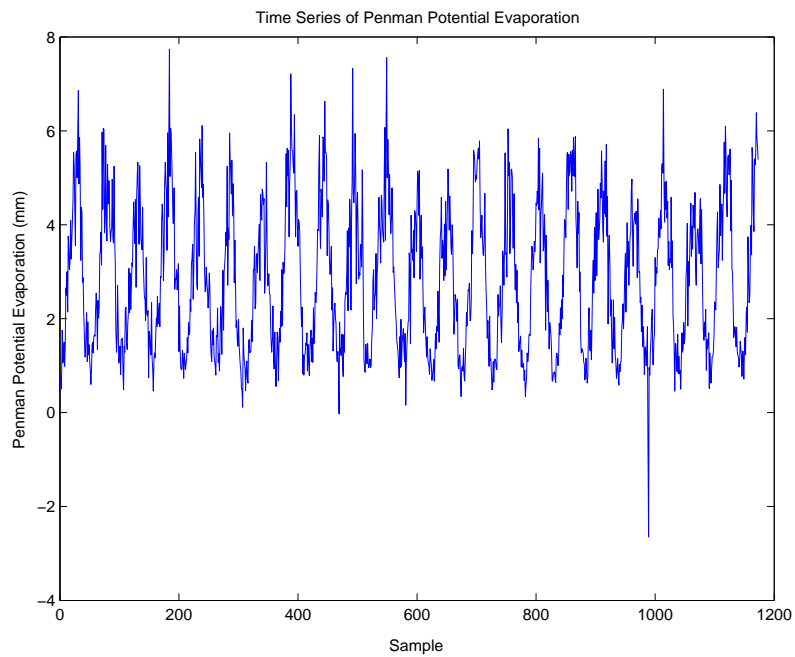


Figure 2.10: Time series plot of Penman potential evaporation (mm).

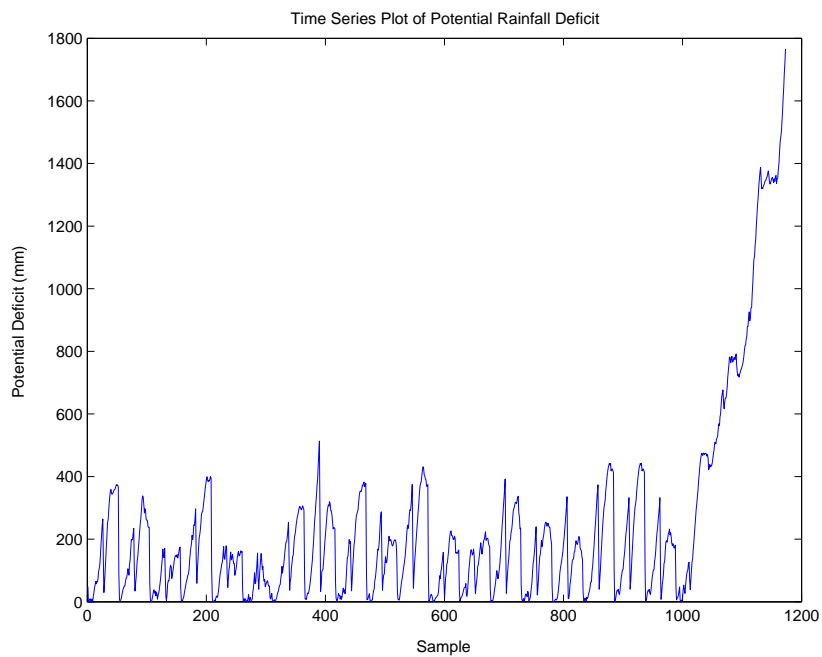


Figure 2.11: Time series plot of potential rainfall deficit (mm).

ten logarithm was taken for all non-zero values of this variable. The results of this transformation are plotted in Figure 2.17. This transformation does, however, raise

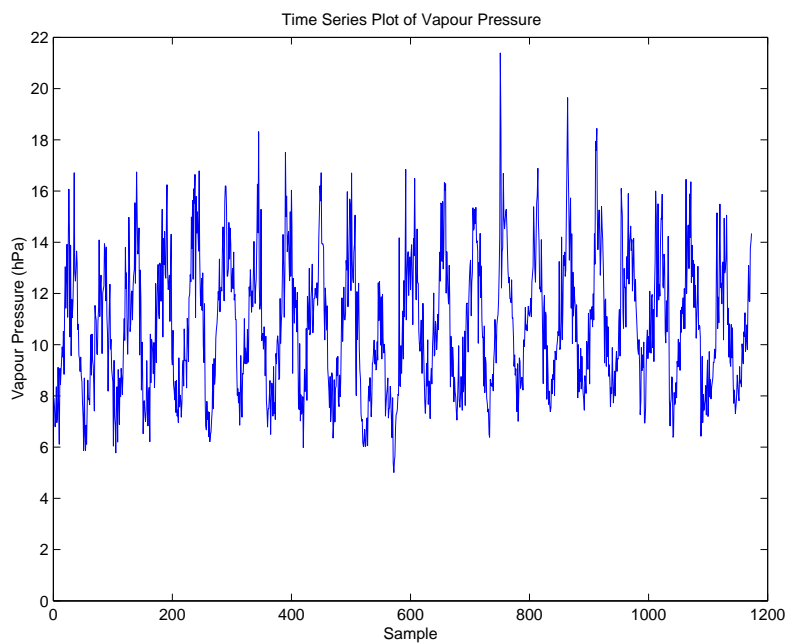


Figure 2.12: Time series plot of vapour pressure (hPa).

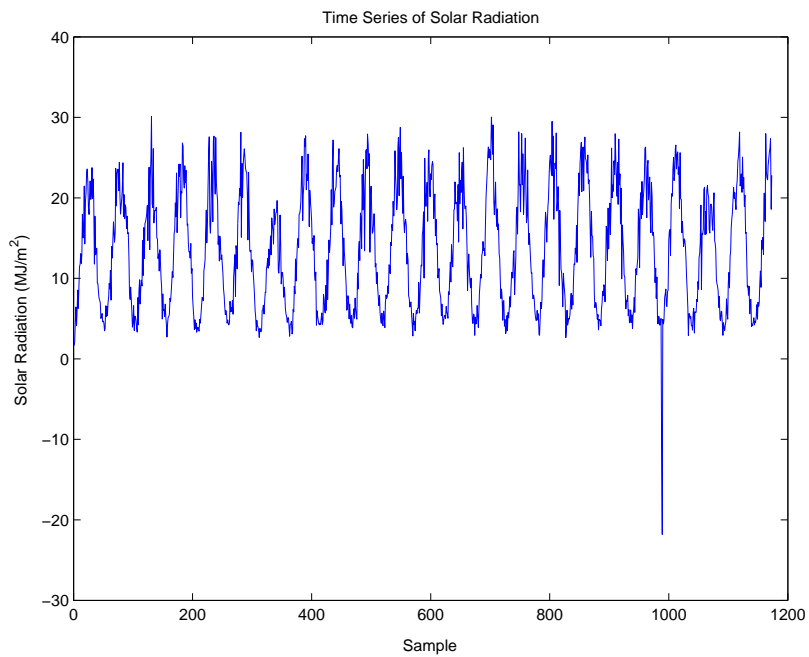


Figure 2.13: Time series plot of solar radiation (MJ/m^2).

a problem, in that when the logarithmic transformation is reversed, those values that should be zero will be transformed as unity. This is not a big problem, however as in

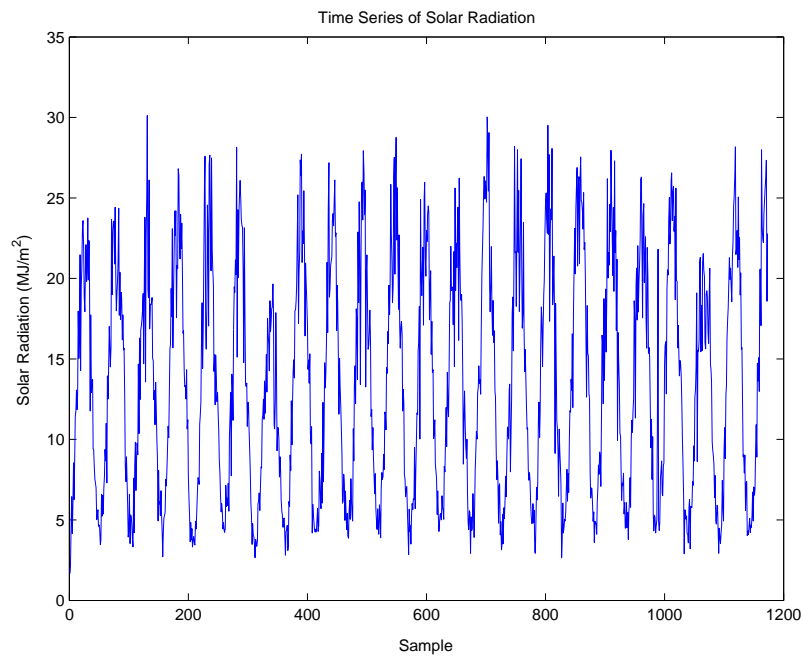


Figure 2.14: Corrected time series plot of solar radiation (MJ/m^2).

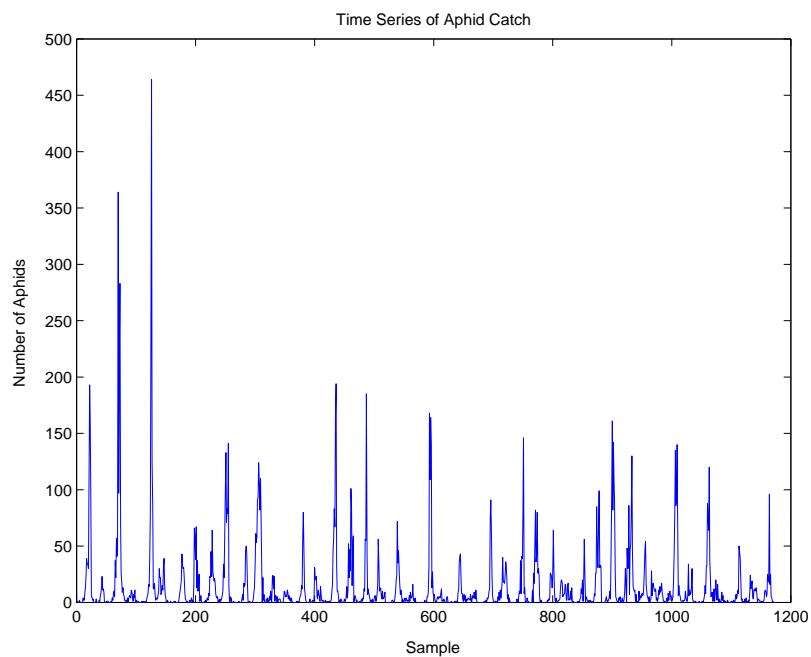


Figure 2.15: Time series plot of aphid catch.

this application the tolerance to prediction errors is quite high.

After the log transformation was applied, all variables, including the target aphid

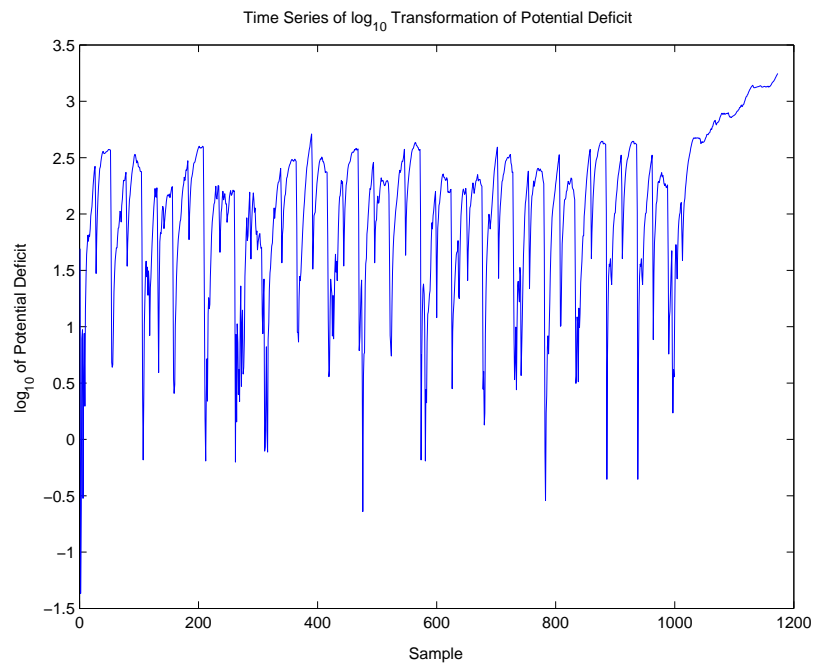


Figure 2.16: Time series plot of log of potential rainfall deficit (mm)

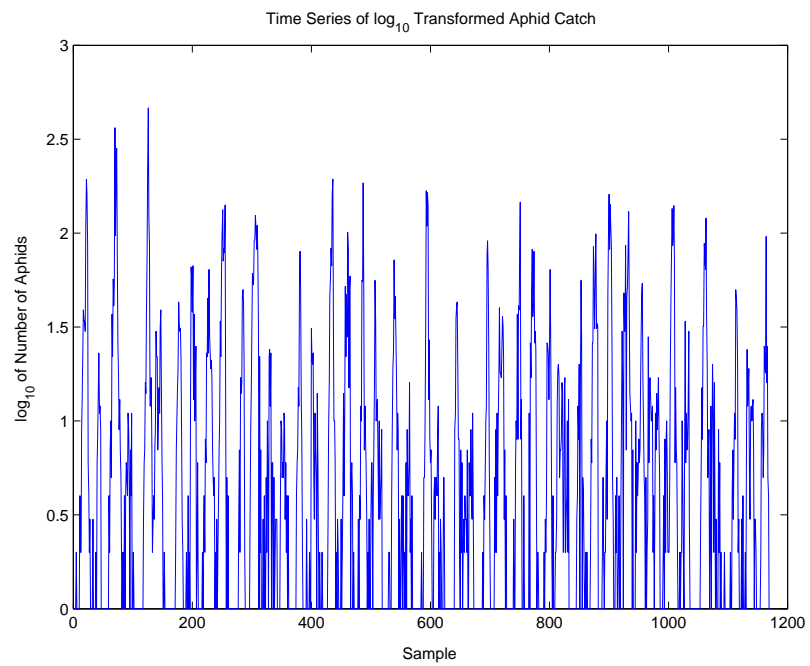


Figure 2.17: Time series plot of log of aphid catch.

counts, were linearly normalise to the range $[0, 1]$ according to Equation 2.1:

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.1)$$

where:

x_n is the normalised value of the variable x ,

x_{min} is the minimum value of x , and

x_{max} is the maximum value of x .

The input data was then time-stepped three steps behind. In other words, if the current week is w , the variables were taken at w , $w - 1$ and $w - 2$. Past aphid counts, again at weeks w , $w - 1$ and $w - 2$ were also included as inputs. The target values were therefore the aphid counts at week $w + 1$, that is, the number of aphids the following week. This time-stepping yielded 1170 examples.

After timestepping, the data set was randomly apportioned into ten equally-sized subsets of 117 examples each.

2.4 General Experimental Procedure

Ten-fold cross validation (Stone, 1974) was used throughout these experiments, where nine of the ten data subsets were used for training and the remaining subset used for testing. This was repeated ten times, so that each subset was used for testing once. For each run, a new MLP of the appropriate architecture was created, trained on the training data set, then tested over the training data set, the test set, and the complete dataset in its original order. There were twenty-five runs performed over each ‘fold’ of the data, for a total of two hundred and fifty runs.

Single hidden-layer MLP were used, and a bias input was included in all cases. The initial weights of the MLP were randomly selected, within the range $[-1, 1]$.

The performance of the networks over each of the data sets was assessed according to Section 2.5.

2.5 Performance Measures and Presentation of Results

Two different measures of accuracy were used in these experiments. The first is the R^2 measure, as defined in Equation 2.2:

$$R^2 = \frac{\sum_i^n p_i^2 - \sum_i^n (p_i - t_i)^2}{\sum_i^n p_i^2} \quad (2.2)$$

where:

p_i is the predicted value of element i

t_i is the target value of element i

n is the number of elements in the target set.

This measure was used because it is the error measure that was used in (Lankin, 2002) and a comparison of these results with those in (Lankin, 2002) is informative. There are, however, problems with the R^2 measure, including difficulties with interpretation, especially with data that contains irreducible error. The R^2 will only be displayed for the tests run over the entire data set, as those are the results that are comparable to those in (Lankin, 2002).

The second measure used was the Mean Absolute Error (ME), as defined in Equation 2.3:

$$e = \frac{\sum |t_o - a_o|}{n} \quad (2.3)$$

where:

e is the mean absolute error,
 n is the number of examples in the data set,
 t_o is the target output value, and
 a_o is the actual output value.

This measure can be more informative than the R^2 , as it gives an estimate of the difference between the number of aphids predicted by the MLP and the actual number of aphids observed.

Since multiple trials were performed over each 'fold' of the data, the variation of the performance measures cannot be measured as a simple standard deviation. Instead, the approximate variance is used, as defined by Equation 2.4.

$$\sigma_t = \frac{\sqrt{\sum_{i=1}^n \sigma_i^2}}{n} \quad (2.4)$$

where:

σ_t is the total, approximated variance,
 n is the number of folds in the data
 σ_i is the standard deviation over the i th fold of the data.

Chapter 3

Experiment One

3.1 Parameter Settings

The backpropagation training parameters used in this experiment are presented in Table 3.1. Ten different network topologies were investigated, with hidden layer sizes of two, four, six, eight, ten, twelve, fourteen, sixteen, eighteen and twenty. Based on the size of the data set, twenty neurons was the maximum feasible size of the hidden layer: larger than this size and over-training would almost certainly occur.

Epochs	1000
Learning rate	0.5
Momentum	0.5

Table 3.1: Training parameters for experiment one.

3.2 Results

The results of these experiments are presented in Table 3.2 for the R^2 measure, and Table 3.3 for the ME measure. Each row

	Number of Hidden Neurons				
Data Set	2	4	6	8	10
Full	-1.11 / 0.22	-0.30 / 0.20	0.18 / 0.18	0.37 / 0.14	0.45 / 0.13
	Number of Hidden Neurons				
Data Set	12	14	16	18	20
Full	0.51 / 0.13	0.57 / 0.10	0.59 / 0.11	0.60 / 0.11	0.60 / 0.11

Table 3.2: Mean R^2 / approximate variance (to 2 d.p.) per network size

		Number of Hidden Neurons				
Data Set		2	4	6	8	10
Train		8.87 / 0.14	8.13 / 0.17	7.66 / 0.20	7.29 / 0.20	7.01 / 0.20
Test		9.46 / 0.18	9.42 / 0.26	9.81 / 0.31	10.12 / 0.32	10.59 / 0.36
Full		8.93 / 0.14	8.26 / 0.17	7.88 / 0.19	7.57 / 0.19	7.44 / 0.20
		Number of Hidden Neurons				
Data Set		12	14	16	18	20
Train		6.94 / 0.22	6.86 / 0.24	6.70 / 0.22	6.57 / 0.22	6.50 / 0.23
Test		10.91 / 0.37	11.32 / 0.40	11.49 / 0.37	11.52 / 0.40	11.65 / 0.40
Full		7.34 / 0.22	7.31 / 0.24	7.18 / 0.22	7.06 / 0.22	7.01 / 0.23

Table 3.3: Mean ME / approximate variance (to 2 d.p.) per network size

3.3 Discussion

Plotting the mean ME for each test yielded the plot in Figure 3.1. It is clear from this plot that the training error is steadily decreasing, while the testing error steadily increases. This is strongly suggestive of over-training, especially among those networks with more than six to eight hidden neurons.

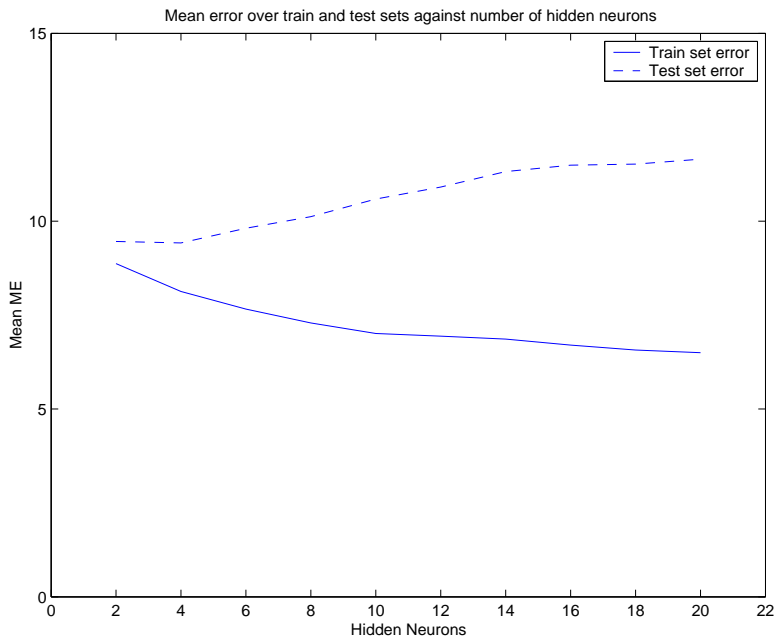


Figure 3.1: Plot of mean error over the train and test data sets against number of hidden neurons.

To determine if the differences in accuracy were statistically significant, two-sided t -tests were carried out comparing the mean ME per network size. The results of these tests on the accuracies over the training data sets are presented in Table 3.4. In this table, an entry of 'reject' indicates that the null hypothesis was rejected, while an entry of 'accept' indicates that the null hypothesis was not rejected. All tests were performed

with $p = 0.01$.

Neurons	20	18	16	14	12	10	8	6	4
2	reject	reject	reject	reject	reject	reject	reject	reject	reject
4	reject	reject	reject	reject	reject	reject	reject	reject	
6	reject	reject	reject	reject	reject	reject	reject		
8	reject	reject	reject	reject	reject	reject			
10	reject	reject	reject	reject	reject				
12	reject	reject	reject	accept					
14	reject	reject	reject						
16	reject	accept							
18	accept								

Table 3.4: Hypothesis test outcomes for the training set results ($p = 0.01$).

The large numbers of rejects indicates that the decrease seen in error over the train sets are statistically significant. In other words, the networks were learning the training data sets significantly better with larger hidden neuron layers. Only near the upper range of hidden layer size do the differences become insignificant. In Figure 3.1 this corresponds to the area where the train error curve starts to flatten out.

Performing the same tests on the accuracies over the testing data sets yielded the results displayed in Table 3.5.

Neurons	20	18	16	14	12	10	8	6	4
2	reject	reject	reject	reject	reject	reject	reject	accept	accept
4	reject	reject	reject	reject	reject	reject	reject	accept	
6	reject	reject	reject	reject	reject	reject	accept		
8	reject	reject	reject	reject	reject	reject			
10	reject	reject	reject	reject	accept				
12	reject	reject	reject	accept					
14	accept	accept	accept						
16	accept	accept							
18	accept								

Table 3.5: Hypothesis test outcomes for the testing set results ($p = 0.01$).

There are a large number of accepts around the tails of the range, which correspond to flat regions of the test error curve. The large number of rejects in the middle of the range, however, show that the increase in the test error was statistically significant. In other words, the generalisation performance of the MLP fell significantly with larger hidden neuron layers.

Overall, this experiment has shown that over this data set, it is very easy to over-train MLP. The following experiment will focus on smaller hidden neuron layers and less aggressive training parameters.

Chapter 4

Experiment Two

4.1 Parameter Settings

Experiment One exhibited a large amount of over-training. This experiment will investigate two strategies for mitigating this. Firstly, utilising smaller hidden neuron layers. Secondly, utilising less aggressive backpropagation training parameters.

The backpropagation training parameters used in this experiment are presented in Table 4.1. Eight different network topologies were investigated, with hidden layer sizes ranging from one to eight neurons. Based on the results of Experiment One in Chapter 3, it was felt that more than eight hidden neurons would not improve performance.

Epochs	500
Learning rate	0.5
Momentum	0.5

Table 4.1: Training parameters for experiment one.

4.2 Results of Experiment Two

The mean R^2 and approximate variances over the full data set are presented in Table 4.2, while the mean ME and approximate variances for this experiment are presented in Table 4.3. The row labels are as in Section 3.2.

	Number of Hidden Neurons			
Data Set	1	2	3	4
Full	-3.16 / 0.57	-1.13 / 0.24	-0.66 / 0.21	-0.43 / 0.21
	Number of Hidden Neurons			
Data Set	5	6	7	8
Full	-0.11 / 0.19	-0.22 / 0.20	0.10 / 0.18	0.19 / 0.15

Table 4.2: Mean ME / approximate variance (to 2 d.p.) per network size

Number of Hidden Neurons				
Data Set	1	2	3	4
Train	9.71 / 0.14	8.89 / 0.14	8.48 / 0.16	8.21 / 0.17
Test	10.07 / 0.15	9.45 / 0.17	9.27 / 0.19	9.24 / 0.23
Full	9.75 / 0.14	8.94 / 0.14	8.56 / 0.16	8.31 / 0.17
Number of Hidden Neurons				
Data Set	5	6	7	8
Train	7.95 / 0.18	7.80 / 0.18	7.64 / 0.17	7.53 / 0.17
Test	9.30 / 0.25	9.34 / 0.27	9.36 / 0.25	9.43 / 0.27
Full	8.08 / 0.17	7.96 / 0.17	7.81 / 0.16	7.72 / 0.17

Table 4.3: Mean ME / approximate variance (to 2 d.p.) per network size

4.3 Discussion

Plotting the mean ME against the size of the hidden neuron layers yielded Figure 4.1.

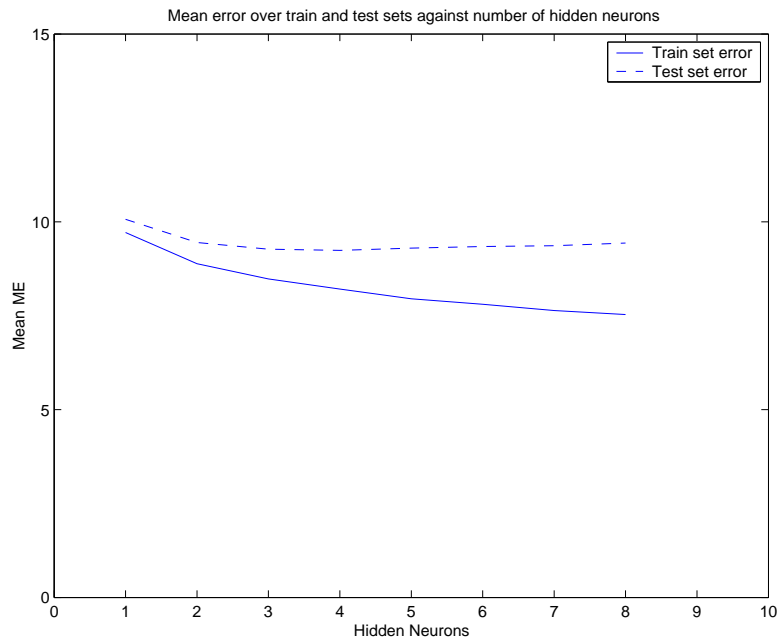


Figure 4.1: Plot of mean error over the train and test data sets against number of hidden neurons.

From Figure 4.1 it can be seen that while the training error continues to decline with increasing hidden neuron layer size, the testing error stays fairly constant. Repeating the statistical tests from Experiment One on the mean training set errors yielded the results presented in Table 4.4. These results show that the performance of the MLP over the training set improved significantly with the increase in size of the hidden neuron layer.

Performing the same tests on the mean testing set errors gave the results displayed in Table 4.5. These results show that there are no significant differences in the general-

Neurons	8	7	6	5	4	3	2
1	reject	reject	reject	reject	reject	reject	reject
2	reject	reject	reject	reject	reject	reject	
3	reject	reject	reject	reject	reject		
4	reject	reject	reject	reject			
5	reject	reject	reject				
6	reject	reject					
7	reject						

Table 4.4: Hypothesis test outcomes for the training set results ($p = 0.01$).

isation performance for any networks with a hidden neuron layer size greater than one. This verifies what is shown in Figure 4.1, where the error curve over the test data set flattens from two hidden neurons on.

Neurons	8	7	6	5	4	3	2
1	reject	reject	reject	reject	reject	reject	reject
2	accept	accept	accept	accept	accept	accept	
3	accept	accept	accept	accept	accept		
4	accept	accept	accept	accept			
5	accept	accept	accept				
6	accept	accept					
7	accept						

Table 4.5: Hypothesis test outcomes for the testing set results ($p = 0.01$).

Chapter 5

Conclusions

In conclusion, it is apparent that it is very easy to over-train an MLP on this data set. From the experiments reported here, it is suggested that a hidden neuron layer size of approximately five neurons is sufficient, with a training period of no more than five hundred epochs.

The performance reported here is not, however, the best reported, as the best mean R^2 is less than that reported in (Lankin, 2002). However, Lankin did report problems with over-training, which suggests that the high R^2 came from memorisation of the training data, rather than a superior generalisation ability.

To illustrate the fitting ability of the MLP created in these experiments, the plot in Figure 5.1 was created. This plots the predicted and observed number of aphids from the best of the five hidden neuron MLP. It can be seen that while it predicts the small peaks very well, the large spikes in the observed aphid numbers is not well modelled.

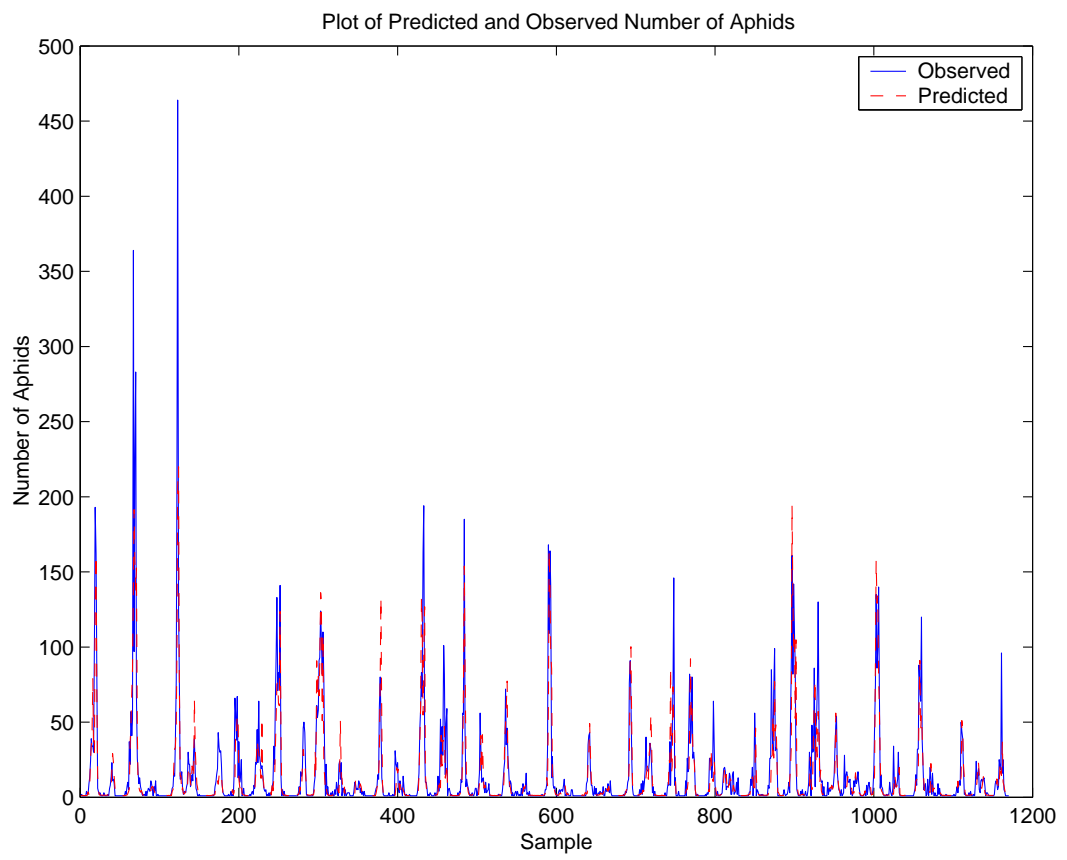


Figure 5.1: Plot of observed and predicted aphid numbers.

Chapter 6

Future Work

There are several avenues for future work. Firstly, a correlation analysis of the data can be performed, so that redundant variables can be identified and eliminated. Secondly, some of the variables may be further transformed before presentation to the MLP. For example, the average and cumulative rainfall variables both have spikes in values that may be removed by a logarithmic transformation.

A sensitivity analysis of the trained MLP would be a useful exercise, as it may help identify the environmental variables that the aphids are most sensitive to.

Further optimisation of the MLP via evolutionary algorithms is also a possibility, such as was done in (Watts et al., 2002).

Bibliography

- Lankin, G. (2002). Neural networks models for the prediction of autumn migration of the cereal aphid *Rhopalosiphum padi* at Lincoln, Canterbury, New Zealand. Master's thesis, Lincoln University.
- Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.
- Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, 36:111–147.
- Watts, M., Major, L., and Tate, W. (2002). Evolutionary optimisation of mlp for modelling protein synthesis termination signal efficiency. In *Proceedings of the Congress on Evolutionary Computation (CEC) 2002, Honolulu, Hawaii*, pages 606–610.
- Worner, S., Lankin, G., Peacock, L., Soltic, S., and Kasabov, N. (2003). Neurocomputing for decision support in ecological and environmental research and its application. In *Proceedings of the Conference on Neuro-Computing and Evolving Intelligence*, pages 39–40.