

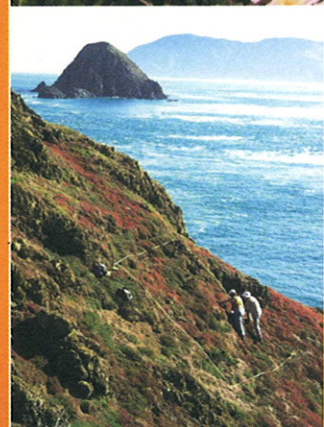
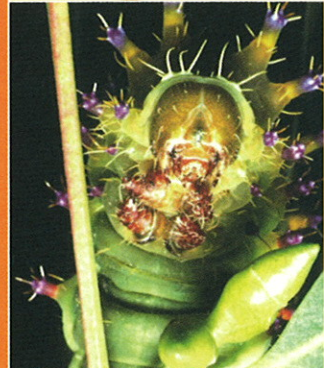
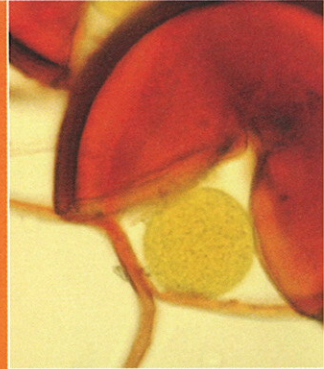
# Bio-Protection & Ecology Division

## Comparison of Multi-Layer Perceptrons and Simple Evolving Connectionist Systems over the Lincoln Aphid Data Set

Michael J. Watts and S.P. Worner



*Bio-Protection & Ecology Technical Report No. 2*



CHRISTCHURCH  
NEW ZEALAND  
[www.lincoln.ac.nz](http://www.lincoln.ac.nz)



**Lincoln**  
**University**  
*Te Whare Wānaka o Aoraki*



ISSN: 1177-7435 ISBN: 0-86476-175-9 978-0-86476-175-9

# Comparison of Multi-Layer Perceptrons and Simple Evolving Connectionist Systems over the Lincoln Aphid Data Set

Michael J. Watts and S.P. Worner

wattsm2, worner@lincoln.ac.nz

# Chapter 1

## Introduction

This report presents two further experiments over the aphid data set. The first is an evaluation of the adaptive abilities of backpropagation of errors trained MLP and a comparison of these capabilities with the Simple Evolving Connectionist System (SECoS) (Watts and Kasabov, 2000). The goal of the first experiment is to compare both the performance and the adaptive abilities of the two models. The second experiment is an investigation of the sensitivity of the SECoS to the exclusion of various input variables. The goal of the second experiment is to determine which of the thirteen input variables contributes the most to the modelling of the problem, that is, which variable the network is most sensitive to.

The same data set as used in the previous report (Watts, 2004), was used for these experiments.

## Chapter 2

# Comparison of MLP and SECoS

### 2.1 Method

Ten-fold cross-validation was again used for these experiments. For each run, eight of the subsets were concatenated into a single training set (Set A), one was used for validation and further training (Set B) and the remaining subset used as the test set (Set C). For each run, the network was trained on Set A, then tested over all three sets. The network was then further trained on Set B and tested again on all three sets. This was done to evaluate four things: firstly, how well the network had learned Set A; secondly, how well it generalised to Sets B and C; thirdly, after further training on Set B, how well it had adapted to Set B; and fourthly, how much it forgot Set A.

The MLP used had five neurons in their hidden layers. The training parameters were as in Table 2.1. These parameters are the same as used in previous work (Watts, 2004). The number of epochs for training over Set B were selected as one-eighth the number of epochs used in Training over Set A because the number of training vectors in Set B is one-eighth the number of vectors in Set A.

	Training Set	
	A	B
Epochs	500	63
Learning rate	0.5	0.5
Momentum	0.5	0.5

Table 2.1: Backpropagation parameters for training MLP

The parameters used to train the SECoS network are presented in Table 2.2. Several different sets of parameters were investigated for training the SECoS networks in an attempt to optimise the generalisation ability of the networks. These parameters are presented in Table 2.2

### 2.2 Results

The mean ME and approximate variance of the results over MLP are presented in Table 2.3.

Run	1	2	3	4	5	6	7	8	9	10	11	12
Error Threshold	0.1	0.2	0.3	0.05	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Sensitivity Threshold	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
Learning Rate One	0.5	0.5	0.1	0.1	0.8	0.65	0.25	0.37	0.45	0.4	0.47	0.41
Learning Rate Two	0.5	0.5	0.1	0.1	0.8	0.65	0.25	0.37	0.45	0.0	0.47	0.41

Table 2.2: Training parameters for ECoS networks

Recall Set	Train Set	
	A	B
A	7.92 / 0.17	9.09 / 0.26
B	9.41 / 0.25	7.71 / 0.24
C	9.50 / 0.26	9.95 / 0.30
Full	8.23 / 0.17	9.04 / 0.25

Table 2.3: Mean ME / approximate variance over MLP

The mean ME and standard deviation for the SECoS networks are presented in Tables 2.4 to 2.15.

Recall Set	Train Set	
	A	B
A	2.46 / 0.30	2.57 / 0.23
B	11.20 / 1.44	1.01 / 0.46
C	11.46 / 1.69	11.51 / 1.59
Full	4.23 / 0.37	3.30 / 0.28

Table 2.4: Mean ME / standard deviation over SECoS for Run 1

## 2.3 Discussion

A comparison of the accuracies of the SECoS networks for each set of training parameters showed that there were no significant differences in the accuracies of the networks despite the changes in the training parameters.

The forgetting and adaptation of the networks was evaluated by testing the statistical hypotheses in Table 2.16. These hypotheses compare the accuracies of the networks before and after further training over Set B. For each hypothesis, the first superscript,  $a$  or  $b$ , indicates which data set the network was trained on. The second superscript indicates which data set the network was recalled with, whether  $a, b, c$  or the full data set,  $f$ .

The results of testing these hypotheses over the MLP results are presented in Table 2.17. In this table and the tables that follow, an entry of “reject” indicates that the null hypothesis was rejected, while an entry of “accept” indicates that the null hypothesis was not rejected.

Inspection of the results in Tables 2.17 and 2.18 show that the MLP adapted very well to the new data, with the performance over Set B being significantly better after

Recall Set	Train Set	
	A	B
A	5.86 / 0.48	6.25 / 0.41
B	11.43 / 2.01	2.83 / 0.77
C	11.73 / 2.37	11.79 / 2.20
Full	7.00 / 0.36	3.30 / 0.32

Table 2.5: Mean ME / standard deviation over SECoS for Run 2

Recall Set	Train Set	
	A	B
A	7.96 / 0.40	8.12 / 0.50
B	12.25 / 2.55	6.50 / 1.42
C	11.72 / 1.79	12.03 / 2.15
Full	8.77 / 0.37	8.35 / 0.41

Table 2.6: Mean ME / standard deviation over SECoS for Run 3

further training on Set B. However, it also forgot, with the performance over Set A decreasing significantly.

The results of testing the Hypotheses in Table 2.16 over the accuracies of the SECoS networks yielded the results presented in Table 2.18.

Inspection of the results in Tables 2.18 and Tables 2.4 to 2.5 show that after further training the SECoS networks were able to significantly improve their performance over Set B without significant degradation of performance over Set A.

These results show that for this problem SECoS networks are able to adapt to new data without forgetting previously seen data. This has positive implications for the construction of future systems that utilise ANN to model the aphid problem.

Recall Set	Train Set	
	A	B
A	5.86 / 0.29	6.25 / 0.28
B	11.43 / 2.09	2.83 / 0.10
C	11.73 / 1.88	11.79 / 1.63
Full	7.00 / 0.37	3.30 / 0.30

Table 2.7: Mean ME / standard deviation over SECoS for Run 4

Recall Set	Train Set	
	A	B
A	3.02 / 0.31	3.41 / 0.30
B	11.61 / 1.82	1.32 / 0.60
C	11.52 / 1.93	11.73 / 2.23
Full	4.88 / 0.26	4.03 / 0.28

Table 2.8: Mean ME / standard deviation over SECoS for Run 5

Recall Set	Train Set	
	A	B
A	2.63 / 0.27	2.83 / 0.21
B	11.57 / 1.80	0.93 / 0.53
C	11.53 / 1.93	11.68 / 1.55
Full	4.41 / 0.26	3.52 / 0.24

Table 2.9: Mean ME / standard deviation over SECoS for Run 6

Recall Set	Train Set	
	A	B
A	1.92 / 0.35	2.04 / 0.34
B	11.17 / 1.62	1.18 / 0.48
C	11.46 / 1.98	11.21 / 1.88
Full	3.80 / 0.33	2.87 / 0.33

Table 2.10: Mean ME / standard deviation over SECoS for Run 7

Recall Set	Train Set	
	A	B
A	2.03 / 0.30	2.17 / 0.33
B	10.91 / 1.54	1.23 / 0.49
C	11.22 / 2.00	11.40 / 2.01
Full	3.84 / 0.29	3.00 / 0.31

Table 2.11: Mean ME / standard deviation over SECoS for Run 8

Recall Set	Train Set	
	A	B
A	2.29 / 0.35	2.42 / 0.28
B	10.98 / 1.59	1.02 / 0.49
C	11.04 / 1.55	11.06 / 1.72
Full	4.04 / 0.29	3.15 / 0.25

Table 2.12: Mean ME / standard deviation over SECoS for Run 8

Recall Set	Train Set	
	A	B
A	2.40 / 0.18	2.49 / 0.22
B	11.07 / 1.89	1.61 / 0.55
C	11.26 / 1.58	11.31 / 1.73
Full	4.15 / 0.26	3.28 / 0.23

Table 2.13: Mean ME / standard deviation over SECoS for Run 10

Recall Set	Train Set	
	A	B
A	2.33 / 0.34	2.46 / 0.29
B	11.10 / 1.27	1.10 / 0.47
C	11.08 / 1.51	11.13 / 1.51
Full	4.09 / 0.31	3.19 / 0.29

Table 2.14: Mean ME / standard deviation over SECoS for Run 11

Recall Set	Train Set	
	A	B
A	2.19 / 0.33	2.34 / 0.35
B	11.18 / 1.66	1.10 / 0.48
C	11.27 / 2.05	11.19 / 1.97
Full	4.00 / 0.39	3.10 / 0.33

Table 2.15: Mean ME / standard deviation over SECoS for Run 12

Hypothesis	$\delta A$	$\delta B$	$\delta C$	$\delta F$
$H_0$	$\mu^{aa} = \mu^{ba}$	$\mu^{ab} = \mu^{bb}$	$\mu^{ac} = \mu^{bc}$	$\mu^{af} = \mu^{bf}$
$H_0$	$\mu^{aa} \neq \mu^{ba}$	$\mu^{ab} \neq \mu^{bb}$	$\mu^{ac} \neq \mu^{bc}$	$\mu^{af} \neq \mu^{bf}$

Table 2.16: Statistical hypotheses for evaluating changes in accuracy after further training.

Hypothesis			
$\delta A$	$\delta B$	$\delta C$	$\delta F$
reject	reject	reject	reject

Table 2.17: Hypothesis test resultss for MLP networks ( $p = 0.01$ )



Run	Hypothesis			
	$\delta A$	$\delta B$	$\delta C$	$\delta F$
1	accept	reject	accept	reject
2	accept	reject	accept	reject
3	accept	reject	accept	reject
4	accept	reject	accept	reject
5	accept	reject	accept	reject
6	accept	reject	accept	reject
7	accept	reject	accept	reject
8	accept	reject	accept	reject
9	accept	reject	accept	reject
10	accept	reject	accept	reject
11	accept	reject	accept	reject
12	accept	reject	accept	reject

Table 2.18: Hypothesis test results for SECoS networks ( $p = 0.01$ )

## Chapter 3

# Sensitivity Analysis of SECoS Networks

### 3.1 Method

A sensitivity analysis of the SECoS networks to the effects of each of the input variables was performed. The speed and deterministic nature of the ECoS training algorithm makes these kinds of network particularly well-suited to this kind of investigation.

Each input variable was excluded in turn, with a SECoS network being created, trained and tested over the training data set each time. The SECoS were tested over the training data set because the goal is to determine which variables contribute the most to the learning of the network. Learning accuracy is best evaluated by determining the accuracy of the network over the training data set.

As ten-fold cross validation was again used, sensitivity of the network to the excluded variable was determined by performing a  $t$ -test comparing the results of the networks with all variables present, and the results of the networks with the target variables excluded.

Two different experiments were carried out. In the first, each of the forty-two time-stepped variables was excluded in turn. In the second, each of the fourteen original variables was excluded in turn.

### 3.2 Results

The mean ME and standard deviation of the SECoS networks, for the removal of time-stepped variables, are presented in Table 3.1. Included in the table is the results of the  $t$ -tests that were testing for equality (at  $p = 0.01$ ).

The mean ME and standard deviation of the SECoS networks, for the removal of entire variables, are presented in Table 3.2. Included in the table is the results of the  $t$ -tests that were testing for equality (at  $p = 0.01$ ).

### 3.3 Discussion

The results presented in Table 3.1 show that the network was sensitive to the average rainfall at time  $w - 1$  and time  $w$ , and the cumulative rainfall at time  $w - 2$ .

Excluded Input	Error	Significant
-	2.46 / 0.30	-
average rain (w-2)	2.43 / 0.23	accept
average rain (w-1)	5.61 / 1.17	reject
average rain (w)	5.71 / 1.28	reject
cumulative rain (w-2)	5.50 / 1.21	reject
cumulative rain (w-1)	2.26 / 0.47	accept
cumulative rain (w)	2.30 / 0.23	accept
wind (w-2)	2.43 / 0.38	accept
wind (w-1)	2.33 / 0.33	accept
wind (w)	2.47 / 0.32	accept
maximum temperature (w-2)	2.33 / 0.38	accept
maximum temperature (w-1)	2.41 / 0.25	accept
maximum temperature (w)	2.32 / 0.30	accept
minimum temperature (w-2)	2.48 / 0.38	accept
minimum temperature (w-1)	2.36 / 0.30	accept
minimum temperature (w)	2.51 / 0.41	accept
mean temperature (w-2)	2.48 / 0.47	accept
mean temperature (w-1)	2.41 / 0.24	accept
mean temperature (w)	2.35 / 0.37	accept
d-days (w-2)	2.48 / 0.47	accept
d-days (w-1)	2.41 / 0.24	accept
d-days (w)	2.36 / 0.37	accept
grass temperature (w-2)	2.45 / 0.33	accept
grass temperature (w-1)	2.44 / 0.41	accept
grass temperature (w)	2.50 / 0.38	accept
soil temperature (w-2)	2.50 / 0.32	accept
soil temperature (w-1)	2.47 / 0.35	accept
soil temperature (w)	2.37 / 0.31	accept
potential deficit (w-2)	2.37 / 0.39	accept
potential deficit (w-1)	2.45 / 0.31	accept
potential deficit (w)	2.47 / 0.32	accept
vapour pressure (w-2)	2.50 / 0.50	accept
vapour pressure (w-1)	2.48 / 0.39	accept
vapour pressure (w)	2.40 / 0.41	accept
penman (w-2)	2.40 / 0.43	accept
penman (w-1)	2.48 / 0.28	accept
penman (w)	2.46 / 0.33	accept
solar radiation (w-2)	2.35 / 0.27	accept
solar radiation (w-1)	2.38 / 0.29	accept
solar radiation (w)	2.42 / 0.32	accept
aphids (w-2)	2.46 / 0.36	accept
aphids (w-1)	2.21 / 0.33	accept
aphids (w)	2.15 / 0.34	accept

Table 3.1: Mean ME / standard deviation and results of hypothesis tests for excluded time-stepped variables.

Excluded Variable	Error	Significant
-	2.46 / 0.30	-
average rain	2.42 / 0.26	accept
cumulative rain	2.42 / 0.26	accept
wind	2.32 / 0.38	accept
maximum temperature	2.39 / 0.26	accept
minimum temperature	2.53 / 0.40	accept
mean temperature	2.52 / 0.42	accept
d-days	2.51 / 0.42	accept
grass temperature	2.39 / 0.36	accept
soil temperature	2.33 / 0.27	accept
potential deficit	2.37 / 0.31	accept
vapour pressure	2.48 / 0.46	accept
penman	2.48 / 0.36	accept
solar radiation	2.37 / 0.36	accept
aphids	13.82 / 0.47	reject

Table 3.2: Mean ME / standard deviation and results of hypothesis tests for excluded variables.

The results presented in Table 3.2 show that the network was sensitive to the previous aphid count variable.

This is an apparent contradiction: if the network is sensitive to the absence of the cumulative rainfall at time  $w - 2$ , then why isn't it sensitive to the complete absence of that variable? This suggests that it is not the values of the variables *per se* that influence the performance of SECoS, but the change over time of those variables. That is, it is not the amount of rain that the network learns but the amount the rainfall changes. This fits with the distance-based mechanism by which ECoS networks function. The same explanation applies to the results of omitting the cumulative rainfall at time  $w - 2$ .

This explanation does, however, raise a further question: why doesn't omitting the aphid numbers at previous weeks cause a change in accuracy, when omitting the entire variable causes a significant decrease in accuracy? In this case, it seems that the presence of any previous aphid count is sufficient for the network to learn. That is, the absence of one weeks aphid catch is not significant. This may be due to the two week life cycle of aphids.

Both of these explanations can be tested experimentally, and will be investigated in future work.

## **Chapter 4**

### **Conclusion**

The results of these experiments show that SECoS networks are able to learn the aphid flight prediction problem. As expected, the SECoS networks were able to adapt to new data better than the MLP networks were, and were able to learn the training set as well as the MLP. SECoS networks were not able to generalise as well as MLP, however, although the difference, while statistically significant, is not biologically significant.

The results of a sensitivity analysis of SECoS determined the importance of several variables, although interpretation of the results requires further experimentation.

## **Chapter 5**

### **Future Work**

Future work will focus firstly on testing the explanations offered for the results of the sensitivity analyses. That is, experiments will be performed that will verify or disprove the explanations offered for the behaviour of the SECoS networks in relation to omitted variables.

The next step to be taken in future work is to correct the data relating to the solar radiation and potential deficit variables. When that is done, a correlation analysis will be performed, to determine the degree of dependence between the variables. This will enable the selection of the least-correlated variables, which will allow for the creation of more parsimonious networks. This is because a large number of correlated variables makes it more difficult for an ANN to learn the data. Correlation analysis will also assist in making a final interpretation of the results presented in this report.

The ultimate goal of this work is to create a single ANN that performs well (in terms of generalisation) over the problem. This ANN needs to have a confidence (or prediction) interval (Chryssolouris et al., 1996) associated with it, so that ranges of values can be derived from each prediction.



# Bibliography

- Chryssolouris, G., Lee, M., and Ramsey, A. (1996). Confidence interval prediction for neural network models. *IEEE Transactions on Neural Networks*, 7(1):229–232.
- Watts, M. (2004). Initial results of using multi-layer perceptrons to model the aphid data set. Technical report, National Centre for Advanced Bio-Protection Technologies.
- Watts, M. and Kasabov, N. (2000). Simple evolving connectionist systems and experiments on isolated phoneme recognition. In *Proceedings of the first IEEE conference on evolutionary computation and neural networks, San Antonio, May 2000*, pages 232–239. IEEE Press.