SACLANTCEN MEMORANDUM
serial no.: SM-235

SACLANT UNDERSEA RESEARCH CENTRE

MEMORANDUM

Target detection using a three-layered neural network trained by supervised back-propagation

H. Meek

May 1990

Page count for SM-235
(excluding covers)

Pages	Total
i–vi	6
1–13	13
	19

SACLANT Undersea Research Centre
Viale San Bartolomeo 400
19026 San Bartolomeo (SP), Italy

tel: 0187 540 111
fax: 0187 560 770
telex: 271148 SACENT I

NORTH ATLANTIC TREATY ORGANIZATION

SACLANTCEN SM-235

Target detection using a three-layered neural network trained by supervised back-propagation

H. Meek

The content of this document pertains to work performed under Project 20 of the SACLANTCEN Programme of Work. The document has been approved for release by The Director, SACLANTCEN.

Issued by:
Systems Research Division

J. Marchment
Division Chief

intentionally blank page

Target detection using a three-layered neural network trained by supervised back-propagation

H. Meek

Executive Summary: A neural network should be considered as a computer program that has the ability to improve its performance at a defined task by making use of a 'learning phase' before it is applied to actual data. During this learning phase a large amount of representative data is repeatedly presented to the network. The network automatically adjusts its internal parameters to optimize its performance as measured by a simple test at the network's output. No information on the internal parameters are presented to the user.

This ability of neural networks to use a priori knowledge often leads them to being incorrecly referred to as 'intelligent systems' but it does not give them the potential to outperform more conventional techniques that do not involve prior training. However, it should be borne in mind that this potentially enhanced performance is only available for the previously defined task. To be as successful at a different task the network must be retrained.

To date, in the realm of sonar, neural networks have been used almost exclusively for classification where it is perceived that operator experience is a vital factor. They can, however, operate as a detection system and still make use of any available a priori knowledge.

In this memorandum a so called three-layered network has been applied to the detection problem using simulated data. The performance of networks as a function of signal-to-noise-ratio, absolute signal level and network complexity has been briefly evaluated.

The results indicate that such a network can perform at least as well as a conventional detection system when simulated data is used. However, work with real sonar data is required before any definite statements can be made as to its applicability to target detection.

intentionally blank page

Target detection using a three-layered neural network trained by supervised back-propagation

H. Meek

Abstract: In any sonar system a detection process has to be performed at the processor output to decide whether or not a particular signal is present in the water. In the particular case of an active sonar employing coherent processing the requirement is to examine the output of the matched filter and decide whether an output signifying the presence of a target echo is present or not. In the present study a neural network has been trained and then applied to this problem. Its performance has been evaluated by examining the statistics of the probability of detection and probability of false alarm using unfamiliar but synthesized data. A preliminary investigation of the effect of varying some of the network parameters has been performed.

Keywords: active sonar ∘ coherent processing ∘ neural networks ∘ probability of detection ∘ probability of false alarm

Contents

Acknowledgement: I would firstly like to thank my supervisor Stephane Jespers for his time and guidance, and for providing the simulation program without which I would surely have floundered. Secondly I must thank Peter Enoch for the major editing performed on this manuscript which has clarified the results significantly.

1
Introduction

The last few years have seen a resurgence of interest in neural networks and their applications. In particular new learning algorithms for networks have been developed, which can provide an alternative to traditional signal processing methods [1–3].

In the present study, neural networks were applied to the problem of target detection in sonar signals. A supervised back-propagation learning algorithm [4] was used to train three-layered networks to recognize the presence of a target echo. The performance of the networks was then evaluated by testing them on unfamiliar data sets. An investigation of the effect of varying some of the network characteristics was begun, but was not completed due to time constraints.

Section 2 will provide an introduction to the theory and architecture of neural networks, with emphasis on the three-layered architecture used in this study. Following this in Sect. 3 is a description of the back-propagation algorithm used in training the networks. Section 4 contains an account of the experiments performed. The sonar data is described, along with the methods employed in training and testing the networks. Some results are presented in Sect. 5. Suggestions for further work are given in Sect. 6.

2

Layered neural networks

A layered neural network, see Fig. 1, consists of at least two distinct layers of neurons or units. Input signals are presented to the network via the *input layer*. The signal resulting from processing by the network is communicated via the *output layer*. In addition there may be one or more *hidden layers* between input and output, so called because they are invisible to the outside world. Hidden layers are used by the network to represent knowledge for feature extraction and problem solving. In general there are weighted connections between all units in a layer and all units in adjacent layers.

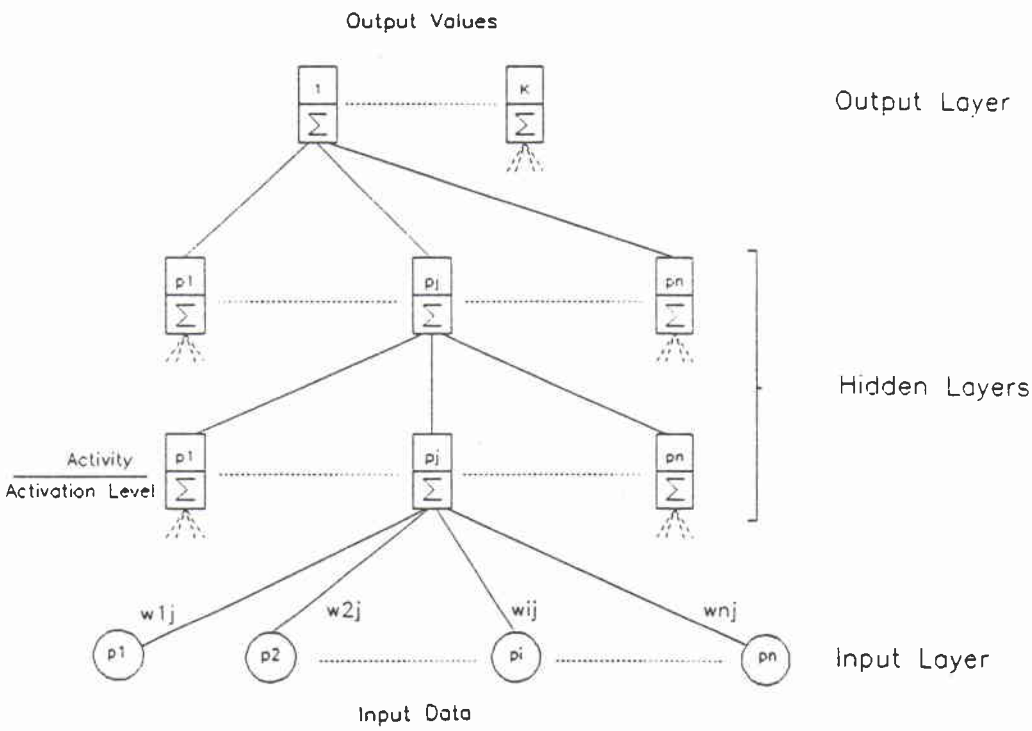

Figure 1 *Generalized neural network.*

The networks used in this study were *feed-forward*, meaning that information flows unidirectionally from input through hidden layers to the output layer. This information flow is in the form of unit *activity*, which is propagated forward from layer to layer. The activity of each unit in the network is computed as follows. The activity of a unit in the input layer is simply the value of the input signal at that unit. For the jth unit in a non-input layer, the *activation level* E_j is the linear weighted sum of its inputs:

$$E_j = \sum_i w_{ij} p_i, \tag{1}$$

where w_{ij} is the connection weight between it and the ith unit in the layer below, and p_i is the activity of this ith unit. A sigmoidal transformation function is then applied to the activation level to obtain the jth unit's activity p_j:

$$p_j = P(E_j) = \frac{1}{1 + e^{-\beta E_j}} - \frac{1}{2}, \tag{2}$$

where β is a constant that determines the slope of the sigmoidal function (a value of $\beta = 1.0$ was used for these experiments). The algorithm originally presented in [4] has been modified slightly as recommended in [5], to provide an activity range from $-\frac{1}{2}$ to $+\frac{1}{2}$. The output values of the network are the activities of the units in the output layer.

In general the behaviour of a neural network is modified by adjusting its connection weights by the repeated application of some learning rule. It can be trained by presenting it with a training set of input signals and corresponding output signals; the learning rule adjusts the weights so that the output produced in response to an input is as close as possible to the desired output. Once the network has been trained, it can be tested by presenting it with signals that were not used during training. If the network responds correctly to these unknown signals it is said that *generalization* has taken place. This capability for generalization is a highly significant attribute of neural nets.

3

Back-propagation algorithm

The networks in this study were trained using the supervised back-propagation method originally proposed in [4], also called the generalized delta rule. The goal of the algorithm is to minimize the average squared error between the output values produced in response to an input signal, and the desired output values for that signal.

For each input signal in the training set, unit activities are propagated forward through the network by applying (1) and (2). The resulting output values are compared to the desired output values, and an error $\delta_j^{(N)}$ is calculated for each output unit as follows:

$$\delta_j^{(N)} = (p_j^* - p_j)P'(E_i^{(N)}), \tag{3}$$

where N is the number of layers in the network (three in this case), p_j^* is the desired value of the jth output unit, p_j is its actual activity, and P' is the first derivative of P. If the difference between the desired and actual outputs $p_j^* - p_j$ is greater than some specified margin (a value of 2.0 was used in these experiments), the error is back-propagated recursively to each lower layer in the network as follows:

$$\delta_j^{(n)} = \sum_i \delta_i^{(n+1)} w_{ij}^{(n)} P'(E_j^{(n)}), \tag{4}$$

where w_{ij} is as in (1). Each connection weight in the network is adjusted in proportion to its contribution to the total error

$$\Delta w_{ij}^{(n)} = \epsilon \delta_j^{(n+1)} p_i^{(n)}, \tag{5}$$

where ϵ controls the rate of learning (a value of 1.0 was used here).

4 Experiments

The networks used in these experiments were three-layered, with 600 input units, one output unit, and a number of hiden units ranging from 25 to 200. An output value of 1.0 represented detection of a target, while 0.0 indicated that there was no target present.

Network training and testing programs were written in the C programming language, while data generation and format conversion programs were written in Fortran. All experiments were run on a VAX 8600 computer under VMS version 4.7.

4.1. SONAR DATA

The investigation did not use real data but instead used a computer program which within sensible bounds simulated the output of a standard matched filter processor when presented with a set of multipath arrivals in a gaussian white noise background. The assumed signal for all the simulations was a 2 sec LFM at a frequency of 200–300 Hz sampled at 1 kHz. Thus with one data value per network input unit, an input layer of size 600 represented a span of 0.6 sec. All signals were input into the network with the target echo in the same place, roughly centered over the 600 units.

The simulation program could generate three different types of data sets each containing 50 signals:

1. Alternate signals contained either an echo-response with noise or pure noise. These signals were used for 'training' purposes only.
2. Each signal contained noise only.
3. Each signal contained both an echo and noise to simulate the 'target present' situation.

In any noise-contaminated signal the noise was generated as a stationary process and chosen to have an equiprobable rms amplitude between two limits ($N_{\max}$ and $N_{\min}$). This represented the noise at the output of the correlator which can be referred to the correlator input using the standard relationship

$$N_{\text{in}} = N_{\text{out}}\sqrt{\tfrac{1}{2}TF_s}, \qquad (6)$$

where N_{in} and N_{out} are the rms noise amplitudes at the input and output of the correlator, T is the pulse time length, and F_s is the sampling frequency.

In order to simulate a complex propagation path the 'echo' within every signal contained 20 components, each with a random but statistically known amplitude and arrival time. The total time spread for all 20 signals was set at 0.2 sec whilst the arrival time of any one component was given a gaussian distribution with a 5 msec standard deviation about a fixed time.

The signal amplitude of one of these components (the 'main' arrival) was set to lie within limits given by the expressions

$$S_{\text{min}} = N_{\text{min}}(R-1),$$
$$S_{\text{max}} = N_{\text{max}}(R+1),$$

where R is a user-defined parameter describing the average ratio (over all 50 pulses) between the main arrival amplitude and the mean amplitude of the noise at the correlator output. (It should be noted that in the results presented in Sect. 5 this ratio has been converted to the more conventional signal-to-noise ratio (SNR) expressed in decibels.) The amplitudes of the other 19 components were described with an equiprobable destribution function lying between 0 and S_{max}.

4.2. *TRAINING THE NETWORKS*

Each network was trained to detect signals with a particular SNR in some particular range of noise amplitudes. Training sets consisted of 50 signals, in which every second signal contained a target echo. (It was determined after preliminary investigation that this ratio of one noise-only signal to one signal with target produced the fastest training rate.) Noise-only signals were given a desired output of 0.0, while signals with a target present had a desired output value of 1.0. The training set was presented repeatedly to the network until the network could correctly classify all of the signals.

The degradation in performance of networks trained on signals with decreasing SNR was examined. In addition the effect of varying the number of hidden units was investigated.

4.3. TESTING THE NETWORKS

Networks were tested as to their ability to detect targets in unfamiliar signals with the same SNR and range of noise amplitudes used in training. Testing sets were composed of either 50 noise-only signals or 50 signals with target present; within each test set the noise amplitude was constant.

The mean and standard deviation of output values were averaged over the 50 signals in each training set. The differences in these values for signal-present and signal-absent data at a particular noise value gave some indication of the network's ability to detect a target at this noise amplitude. For each network these values were averaged over all noise amplitudes to summarize the performance of the network.

5
Results and discussion

The results of testing a network with 50 hidden units, trained on noise amplitudes equivalent to 4–6 V at the correlator input and SNR 12 dB, are given in Table 1. It can be seen from the table that the ratio of average output to standard deviation of output with a target present increases with noise amplitude, indicating that detection is better at signal levels at the high end of the training range. In fact from results obtained using different ranges of noise amplitude it appears that the networks are sensitive to the absolute value of the noise amplitude used during training; the precise extent and nature of this dependence is not known as there was not sufficient time to do a detailed investigation.

Table 1 *Results of testing a network with 50 hidden units**

N	Noise-only		Target present	
	$\overline{P}$	σ_P	$\overline{P}$	σ_P
4.0	0.1083	0.1339	0.6499	0.2930
4.5	0.1832	0.2316	0.7537	0.2603
5.0	0.1955	0.2115	0.7384	0.2587
5.5	0.1828	0.2147	0.8066	0.2452
6.0	0.2021	0.2528	0.8588	0.2297
Avg.	0.1744	0.2089	0.7615	0.2574

* For each noise amplitude N the network was tested with 50 noise-only signals and 50 signals with target present. The SNR throughout was 12 dB. The mean and standard deviation are given for the output values P. The network had 50 hidden units.

On the assumption that the distribution of output values for noise-only signals was gaussian, the mean and standard deviation of this distribution were approximated by averaging those values over all test sets of noise-only data. The distribution of outputs for signals with target present was characterized in a similar manner. From these parameters it was determined that a threshold output value of 0.4564 would maximize the probability of detection (P_{d}) and minimize the probability of false alarm ($P_{\mathrm{f.a.}}$) of a system with these output distributions. This threshold yielded a theoretical P_{d} of 0.8820 and a theoretical $P_{\mathrm{f.a.}}$ of 0.0886; however, the actual P_{d} and

$P_{f.a.}$ obtained from the data using this threshold were 0.8480 and 0.1280. It is clear that the distributions are not gaussian but have heavy tails, probably as a result of averaging over such a wide noise range.

The degradation in performance with decreasing SNR was investigated using networks with 50 hidden units and an equivalent noise amplitude range at the input correlator of 4–8 V. Each network was trained and tested on pings with a constant SNR. The mean and standard deviation of output values, averaged over signals with and without a target as described above, are given in Table 2. In addition the number of repetitions of the training set required to train each network is given. As a means of comparing the performance of different networks, the P_d was fixed at 0.9 and the $P_{f.a.}$ was determined by counting the number of false alarms produced by each network. Figure 2 shows the performance of the networks as a function of SNR. It is clear that both performance and training rate deteriorate rapidly as the SNR goes below 12 dB.

Networks with sizes of hidden layer ranging from 25 to 200 were trained with the same training set. The training and testing signals had noise amplitudes of 4–6 V and a SNR of 12 dB. The effect of hidden-layer size on network training rate and output values is given in Table 3. Output values seem to improve with the number of hidden units, in the sense that the average output for noise-only signals decreases while that for signals with a target present increases. However, the variation in output also increases with hidden-layer size.

Figure 3 shows the effect on performance of the network, as defined above. Performance seems to stabilize for large hidden layers, while the training rate increases. This suggests that there is some optimal number of hidden units, which is supported by results obtained from preliminary testing on simple functions such as x^2 and x^{-1}. In fact it appeared in these cases that performance actually degraded above a certain size of hidden layer; it is unfortunate that there was insufficient time to train networks with more than 200 hidden units to see if the same effect was observed.

Table 2 *Degradation of performance with decreasing SNR**

SNR (V)	Repetitions	Noise-only		Target present	
		$\overline{P}$	σ_P	$\overline{P}$	σ_P
16.9	15	0.0610	0.0451	0.9620	0.0616
15.5	15	0.0400	0.0388	0.9297	0.1020
14.0	18	0.0919	0.0934	0.9146	0.1138
12.0	20	0.1424	0.1417	0.8648	0.1778
9.5	65	0.1564	0.2487	0.7637	0.3100
6.0	129	0.2284	0.3197	0.5047	0.3915

* Output mean and standard deviation were averaged over noise values. Networks had 50 hidden units; noise amplitudes were 4–8 V. Number of repetitions of training set required for training is also given.

Table 3 *Effect of hidden-layer size on network performance**

Units	Repetitions	Noise-only		Target present	
		$\overline{P}$	σ_P	$\overline{P}$	σ_P
25	27	0.2142	0.1840	0.7387	0.2492
50	27	0.1744	0.2089	0.7615	0.2574
100	23	0.1737	0.2142	0.7947	0.2533
200	38	0.1478	0.2221	0.8349	0.2567

* Noise amplitudes were 4–6V; SNR 12 dB.

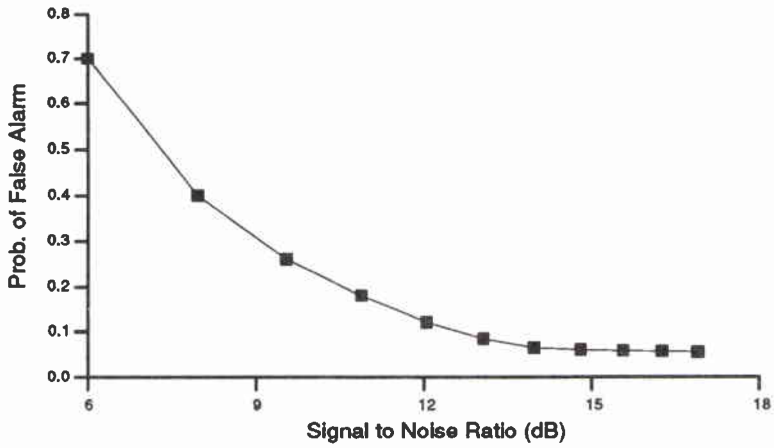

Figure 2 *Probability of false alarm when $P_d = 0.9$ vs signal-to-noise ratio; noise amplitudes were 4–6 V; the networks had 50 hidden units.*

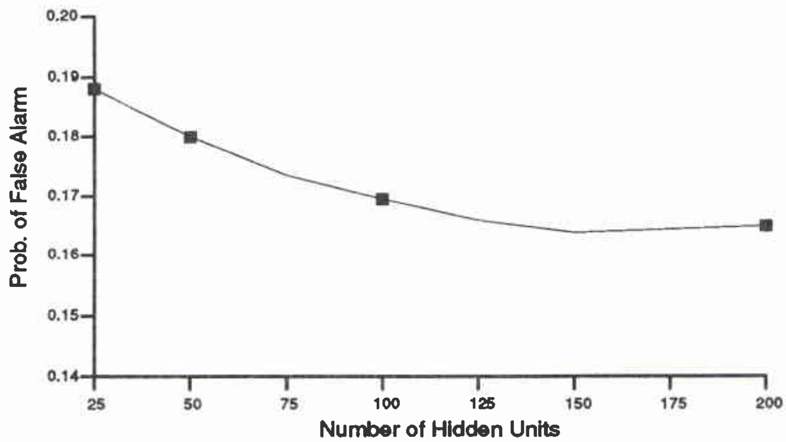

Figure 3 *Probability of false alarm when $P_d = 0.9$ vs size of hidden layer; noise amplitudes were 4–6 V, SNR was 12 dB.*

6

Suggestions for further work

Areas indicated for further investigation can be divided into two categories: data characteristics and network parameters. In the first case, the dependence of performance on absolute noise amplitude needs to be understood. Other factors that could be studied include size and composition of training set and sampling frequency of signals.

The sensitivity of the network's performance to varying the echo statistics would also deserve a deeper study. In fact, the present performance of the 50 hidden units' network *vs* SNR in presence of the signal statistics described in Subsect. 4.1 is not immediately comparable with detection statics of signals under normal circumstances (i.e. those signal statistics found in the standard textbooks). Due to the similarity with model-based processing, it may be anticipated that a neural network should achieve a performance that lies somewhere between that of a fully incoherent post-processor (e.g. the energy averager gain) and the fully coherent recombination gain of the multipath structure achieved by an optimal model-based processor.

It is likely that network performance can be improved by finding some optimal balance of parameters such as learning rate, error tolerance while training, number of input units and hidden units. Some of these parameters were not investigated at all in this study, while others were given a hasty examination. For instance, rudimentary testing indicated that for simple functions, training rate increased with learning rate while performance remained roughly constant. However for complicated data, the time needed for network connection weights to converge during training seemed extremely high with learning rate values greater than 0.1 (which is why this value was chosen). It would be interesting to discover just how much improvement could be obtained by fine-tuning these parameters.

References

[1] Burr, D.J. Experiments on neural net recognition of spoken and written text. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **36**, 1988: 1162–1168.

[2] Gorman, R.P. and Sejnowski, T.J. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, **1**, 1988: 75–89.

[3] Gorman, R.P. and Sejnowski, T.J. Learned classification of sonar targets using a massively parallel network. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **36**, 1988: 1135–1140.

[4] Rumelhart, D.E., Hinton, G.E. and Williams, R.J. Learning internal representation by error propagation. *In*: Rumelhart, D.E. and McClelland, J.L., *eds.* Parallel Distributed Processing: Exploration in the Microstructure of Cognition. Cambridge, Ma., MIT Press, 1986: pp. 318–362.

[5] Stornetta, W.S. and Huberman, B.A. An improved three-layer, back-propagation algorithm. *In:* INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. Proceedings of the first IEEE international conference on neural networks, San Diego, CA, 21–24 June, 1987. San Diego, CA, SOS Printing, 1987: pp. 637–643.

Report no. changed (Mar 2006): SM-234-UU

Initial Distribution for SM-235

Ministries of Defence	
JSPHQ Belgium	2
DND Canada	10
CHOD Denmark	8
MOD France	8
MOD Germany	15
MOD Greece	11
MOD Italy	10
MOD Netherlands	12
CHOD Norway	10
MOD Portugal	5
MOD Spain	2
MOD Turkey	5
MOD UK	20
SECDEF US	60

NATO Authorities	
NAMILCOM	2
SACLANT	3
CINCIBERLANT	1

SCNR for SACLANTCEN	
SCNR Belgium	1
SCNR Canada	1
SCNR Denmark	1
SCNR Germany	1
SCNR Greece	1
SCNR Italy	1
SCNR Netherlands	1
SCNR Norway	1
SCNR Portugal	1
SCNR Spain	1
SCNR Turkey	1
SCNR UK	1
SCNR US	2
French Delegate	1
SECGEN Rep. SCNR	1
NAMILCOM Rep. SCNR	1

National Liaison Officers	
NLO Canada	1
NLO Denmark	1
NLO Germany	1
NLO Italy	1
NLO Netherlands	1
NLO UK	1
NLO US	1

NLR to SACLANT	
NLR Belgium	1
NLR Canada	1
NLR Denmark	1
NLR Germany	1
NLR Greece	1
NLR Italy	1
NLR Netherlands	1
NLR Norway	1
NLR Portugal	1
NLR Turkey	1
NLR UK	1

Total external distribution	219
SACLANTCEN Library	10
Stock	31
Total number of copies	260