# Learning and generalisation in a stable network

**Anthony Robins[1] and Marcus Frean[2]**

[1]Computer Science, The University of Otago, PO Box 56, Dunedin, New Zealand
coscavr@otago.ac.nz
[2]Electrical and Computer Engineering, University of Queensland, Brisbane 4072, Australia
marcusf@elec.uq.edu.au

## Abstract

Most neural networks suffer from excessive plasticity, the learning of new information interferes with information already stored in the network. In this paper we review the pseudorehearsal solution to this problem proposed by Robins [1995]. By localising the changes to the function learned by the network pseudorehearsal allows networks to be stable in the face of new learning, successfully integrating both new and previously learned information. In this paper we explore the impact that this mechanism has on the ability of the network to generalise.

## 1 Introduction

Ideally the representations developed by a learning system should be stable enough to preserve important information over time, but plastic enough to incorporate new information when necessary. The use of variable connection weights as a medium for encoding information leads most ANNs to err on the side of excessive plasticity – new learning changes the weights and thus disrupts any old information (patterns previously learned by the network). Grossberg [1987] suggests the analogy of a human trained to recognise the word "cat", and subsequently to recognise the word "table", being then unable to recognise "cat". This effect has been identified in many guises in the ANN literature under headings such as catastrophic forgetting, catastrophic interference, or the serial learning problem. It is due to the catastrophic forgetting problem that most ANN learning algorithms are based on "concurrent" learning, i.e. the whole population of interest is presented and trained as a single, complete entity. Training is then "finished" and no further information is learned by the network.

To be restricted to concurrent learning is undesirable in practical terms, making it very difficult to modify or extend any given ANN application without completely retraining the network (compared with a traditional rule based system where information or rules can easily be added to or removed from the system). It is also a highly implausible constraint for ANN based cognitive models, where so much of human learning is clearly sequential or incremental in nature.

Ideally a solution to the catastrophic forgetting problem would facilitate the training of ANNs in a "sequential" or cumulative way, allowing new information to be learned by a network at any time. In this paper we briefly review the pseudorehearsal method proposed by Robins [1995] and extended in [Robins, 1996; Frean and Robins, 1997; French, 1997; Robins and McCallum, 1997], which provides a practical and interesting solution to the catastrophic forgetting problem. We then explore the impact of the pseudorehearsal mechanism on generalisation.

## 2 Catastrophic Forgetting

A number of recent studies have used back-propagation type networks to highlight the problem of catastrophic forgetting and explore various issues [McCloskey and Cohen, 1989; Hetherington and Seidenberg, 1989; Ratcliff, 1990; Murre, 1992; French, 1992, 1994, 1997; Lewandowsky and Li, 1995; Sharkey and Sharkey, 1995; Robins, 1995, 1996; Frean and Robins, 1997]. Typical illustrations of the problem use a back-propagation network to learn a population of items (input / output vector pairs) in the usual way [Rumelhart *et al.,* 1986]. Subsequently a new item or population of items is learned. The effect of this new learning on the original population can be illustrated by re-testing the original population. The error of the old population of items typically increases "catastrophically" after learning even a single new item.

Following Frean and Robins [1997] in this paper we will initially present the catastrophic forgetting effect in a different way, focusing directly on the the function learned by the network. To do this we use a simple architecture consisting of a single input unit, 20 hidden units, and a single output unit (with a learning constant of 0.05, a momentum of 0.9, and an error criterion of 0.001). The initial population consists of 6 items (input value / output value pairs), which can be plotted as data points in two dimensions. After training we can plot the function learned by the network by systematically sampling the space of possible input values. As expected this function passes through the population data points, see Fig. 1(a)[1]. To illustrate catastrophic forgetting we now train the network on a single new item. Fig. 1(b) shows the resulting function, which correctly fits the new item data point. Notice however that this function no longer fits the initial data points at all (the original population inputs will not generate the correct outputs). After learning just a single new item catastrophic forgetting has occurred.

A range of solutions to catastrophic forgetting have been explored in the studies noted above, focusing in particular on

[1]This network architecture and training population have been chosen so as to reduce the learning task to its simplest possible conditions and allow us to visualise the function learned by the network as a two dimensional graph. We intend this as an approximation of typical ANN learning tasks, where in general a network with a large number of free parameters (weights) is used to learn a smaller number of training items that define a function over some (multi-dimensional) input space.

either reducing the overlap of hidden unit representations, or rehearsing old items as new items are learned (see Frean and Robins [1997] for a brief review). Rehearsal based methods are the most directly effective, with "broad but shallow" rehearsal regimes (rehearsing many old items as each new item is introduced but not necessarily training them to criterion) producing particularly good results. Using such a regime we can solve the catastrophic forgetting problem shown in Fig. 1(b) and learn a function that fits both the old population and the new item data points. The obvious limitation of rehearsal based methods, however, is that they require access to the population on which the network was originally trained. This may be impractical [Sharkey and Sharkey, 1995], particularly in the case of possible cognitive mechanisms. In any case, retaining old items for rehearsal in memory seems somewhat artificial, as it requires that they be available on demand from some other source, which would seem to make the memory itself redundant!

## 3 The Pseudorehearsal Solution

The catastrophic forgetting problem can also be solved using pseudorehearsal [Robins, 1995]. This mechanism is similar to rehearsal, but it does not require the storage and access of old information. In other words we can get the benefits of rehearsal even when we do not have the original population to rehearse! The method is based on the use in the rehearsal process of artificially constructed populations of "pseudoitems" instead of the previously learned items themselves.

A pseudoitem is constructed by generating a new input vector at random (setting input unit values randomly in the range 0 to 1), and passing it forward through the network in the standard way. Whatever output vector this input generates becomes the associated target output. For a given network we can construct a population of pseudoitems (input / output pairs) of any size in this way. Pseudorehearsal is the use in a rehearsal process of a population of pseudoitems instead of the actual previously learned items themselves. In other words, rehearsal relearns the old population data points to preserve the shape of the function during subsequent learning, while pseudorehearsal samples the function at random places to construct new data points which are then relearned in the same way to preserve the original function. The pseudoitems "approximate" the old population and "map out" the function learned by the network. Relearning the pseudoitems restricts any change in the function to be local to the area of the new item(s) being learned. Frean and Robins [1997] extend this account, providing a more technical description of pseudorehearsal and a formal analysis of a simple case for a linear network.

The effectiveness of pseudorehearsal at reducing catastrophic forgetting has been proven using a range of populations, including: randomly constructed autoassociative and hetroassociative data sets [Robins, 1995]; the Iris data set [Robins, 1996]; a classification task using the Mushroom data set [French, 1997]; and an alphanumeric character set using a Hopfield type network [Robins and McCallum, 1997][2].

[2]The Iris and Mushroom data sets are taken from [Murphy and Aha, 1994].

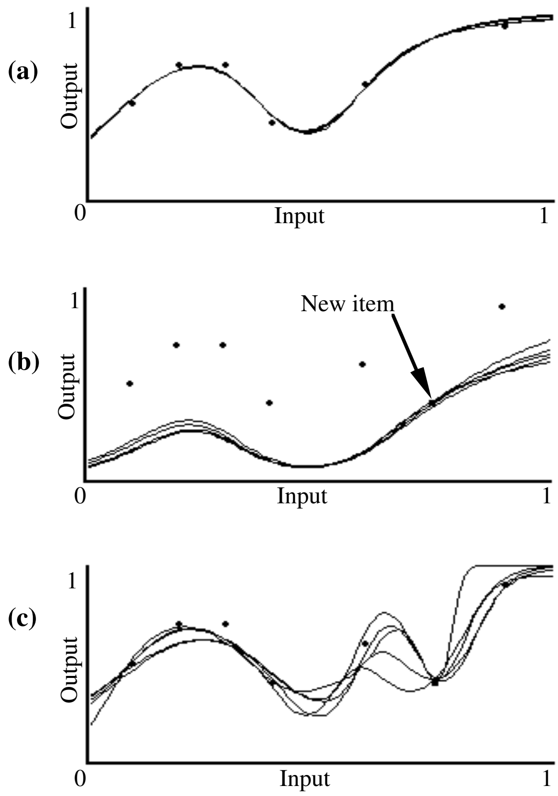


**Fig. 1.** Functions learned by a network (five replications). (a) shows a network trained on six data points. (b) and (c) show the subsequent learning of the same single new data point, using normal learning and using pseudorehearsal respectively.

We can illustrate the effects of pseudorehearsal in the task described in Section 2. After training on the old population data points (i.e. from the network shown in Fig. 1(a)) a population of 20 pseudoitems is constructed as described above. The new item is then learned not alone, but in a buffer which contains (in this case) ten pseudoitems. Pseudoitems to be included in the buffer are selected from the population of pseudoitems at random for each epoch of training (replacing the ten pseudoitems used in the previous epoch). Training continues until the single new item, which is always in the buffer, has reached the error criterion. Fig. 1(c) shows the kinds of function which result from pseudorehearsal in this task. These functions not only fit the new data point, but also reasonably fit the original data points. Pseudorehearsal has eliminated the catastrophic forgetting that occurs in normal learning (Fig. 1(b)).

There are many ways to further improve the performance of pseudorehearsal. Pseudoitems could be trained to criterion like the new item. Up to a certain limit the more pseudoitems in the population, and the more pseudoitems used in the training buffer, the better. A large number of pseudoitems both samples the structure of the function in more detail, and also leads to greater "inertia" in the sense that the function is unwilling to change anywhere except where it is being "pulled" by the error associated with the new item. The method used here of "rotating" a number of pseudoitems through the rehearsal buffer also has some useful properties. Without this rotation a buffer containing a fixed set of pseudoitems would use (for each new item) fewer pseudoitems more frequently, and would thus be prone to "over-fitting" these pseudoitems and to conflicts when pseudoitems fall very close to the new item.

It is important to note that pseudorehearsal does *not* preserve the weights of a network, these can change dramatically during learning. The method works directly at the level of the function embodied by the network. The old population function is preserved, and changes necessary to accommodate the new item are restricted to being *local* to the region of the new item. This localisation of changes to the function is the essence of the pseudorehearsal method. These and other issues are pursued in more detail in Frean and Robins [1997] and Robins [1995].

## 4 Pseudorehearsal and Generalisation

Generalisation is based on smooth interpolations (and, less reliably, extrapolations) of and between training data points in the function learned by the network. We might expect, therefore, that restricting changes to the function to areas which are local to each new item learned will have a detrimental effect on the ability of the network to generalise. The main focus of this paper is to undertake an initial investigation of this hypothesis.

In the following experiments we use the Iris data set [Murphy and Aha, 1994] consisting of 150 items (specific plants as characterised by four real valued measures), which fall in to three classes (distinct species of iris) of 50 items each. We use two training populations, A and B, each drawn from a specific class, and corresponding generalisation populations, A-gens and B-gens, drawn from the same two classes. The network is a 4:3:4 autoassociative back-propagation net with a learning constant of 0.05 and a momentum constant of 0.9. Populations are trained to a criterion of 0.01 using an average of the sum of the squared errors. Results reported for each of the simulations below are averaged over 25 individual replications.

Generalisation performance typically varies with the size of the population. In order to test performance in detail, therefore, the following simulations use training populations A and B ranging in size from 1 to 15 items. The generalisation populations always consist of 20 items.

As a baseline condition we wish to test the network's underling capacity to learn and generalise the populations of interest. We train the network on a combined training set of the A and B populations, and subsequently test it on A, B, A-gens and B-gens. Fig. 2 shows the performance of the network (note, for a range of sizes for populations A and B). We can see that although the combined A and B was trained to an average criterion of 0.01, when tested separately A has a slightly higher error, and B slightly lower (for all population sizes). In both cases the ability of the network to generalise improves as the size of the population increases, as expected. For these particular classes of iris, however, A does not generalise as well as B. Using this baseline measure of the networks capacity we can now explore conditions where the populations are learned sequentially, using standard learning and using pseudorehearsal.

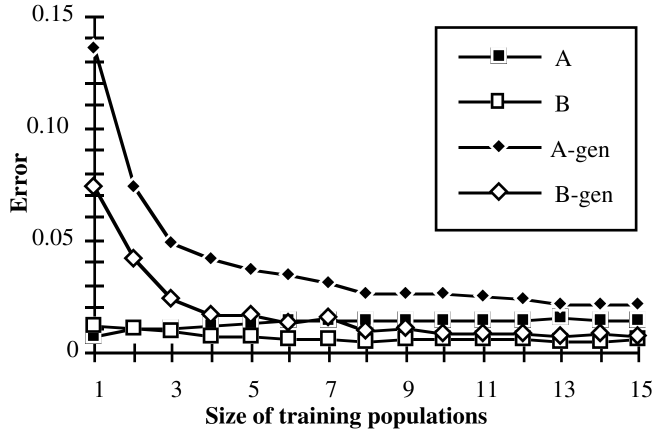


**Fig. 2.** Learning populations A and B as a single group.

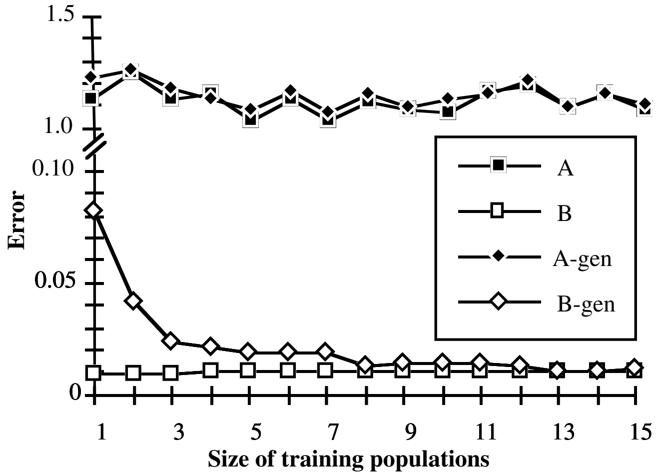


**Fig. 3** Learning population A followed by population B. (Note the two different scales on the *y* axis).

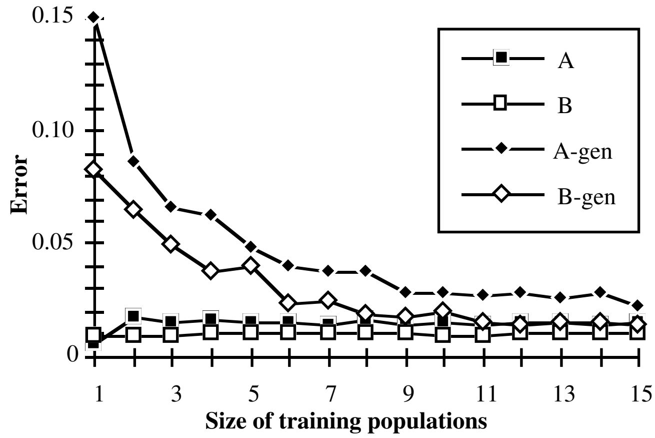


**Fig. 4** Learning population A followed by population B with pseudorehearsal.

Sequentially learning the populations using standard methods illustrates the catastrophic forgetting effect. We train the network to criterion A, and then train it to criterion on B in the usual way. Fig. 3 shows the performance of the network when subsequently tested on A, B, A-gens and B-gens. Performance on A and A-gens has been reduced to effectively random levels by the subsequent learning of B.

Note that performance on B and B-gens is almost identical to the case illustrated in Fig.1.

Sequentially learning the populations using pseudorehearsal illustrates both the protective effect on A, and also interesting consequences for the network's ability to generalise. We train the network to criterion A, and then train it to criterion on B using pseudorehearsal. Specifically, before the learning of B begins we generate a population of 256 pseudoitems. The items of the B population are then learned not alone, but in combination with three times as many pseudoitems randomly selected from the population of pseudoitems for each epoch. (In other words, if the size of the B population is 2 then every epoch 6 pseudoitems are included in the training set). After B has been trained to criterion we can test the network on A, B, A-gens and B-gens in the usual way. In Fig. 4 we can see that the use of pseudorehearsal has done an excellent job of preserving performance on A and A-gens (cf normal learning illustrated in Fig. 3). B is trained to criterion as expected. Of particular interest is performance on B-gens. This has suffered somewhat compared to the other training methods (Figures 2 and 3).

From Fig. 2 we know that the network has the capacity to learn both A and B with a certain level of performance on B-gens. From Fig. 3 we know that learning B after A does not itself impact on this generalisation, performance on B-gens is practically identical in the concurrent and sequential cases. The reduced performance on B-gens in Fig. 4 must be a consequence of the pseudorehearsal process. Specifically, we suggest that it arises because pseudorehearsal forces changes made to the function to be local to the new items learned, rather than unconstrained as is the case with normal learning.

## 5 Discussion

Catastrophic forgetting is a very basic consequence of a plastic representing medium, and most ANNs are vulnerable to this problem. Consequently most learning algorithms involve learning all information of interest concurrently. By constraining changes to the function to be local to the new item(s), pseudorehearsal provides a useful solution to this problem that allows new information to be learned at any time (without needing access to old information to rehearse).

We have previously claimed [Robins, 1996] that, unlike other proposed solutions to the catastrophic problem, pseudorehearsal has no detrimental effect on the ability of the network to generalise. This claim was based on large populations where performance is generally good. In this paper we have explored generalisation in more detail, considering a range of population sizes. We have shown that localising changes to the function does in fact have a slightly detrimental impact on generalisation, especially for small populations. It remains to be seen if this proves to be a significant problem in practical applications.

## References


[Frean and Robins, 1997] Frean,M. and Robins,A.V. Catastrophic Forgetting: A Review and an Analysis of The Pseudorehearsal Solution. (Tech. Rep. *AIM-36-97-2*). Otago, New Zealand: Otago University, Department of Computer Science. (Submitted to Network)

[French, 1992] French,R.M. Semi-distributed Representations and Catastrophic Forgetting in Connectionist Networks. *Connection Science, 4,* 365 - 377.

[French, 1994] French,R.M. Dynamically Constraining Connectionist Networks to Produce Distributed, Orthogonal Representations to Reduce Catastrophic Interference. *Proceedings of the 16th Annual Cognitive Science Society Conference.* Hillsdale, NJ: Earlbaum, 335 - 340.

[French, 1997] French,R.M., Interactive Connectionist Networks: An Approach to the "Sensitivity-Stability" Dilemma. *Connection Science,* in press.

[Grossberg, 1987] Grossberg,S. Competitive Learning: From Interactive Activation to Adaptive Resonance. *Cognitive Science, 11,* 23 - 63.

[Hetherington and Seidenberg, 1989] Hetherington,P.A. and Seidenberg,M.S., Is There "Catastrophic Interference" in Connectionist Networks? *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society.* Hillsdale NJ: Lawrence Earlbaum, 26 - 33.

[Lewandowsky and Li, 1995] Lewandowsky,S. and Li,S., Catastrophic Interference in Neural Networks: Causes, Solutions, and Data. In Dempster,F.N. and Brainerd,C. (Eds.) *Interference and Inhibition in Cognition.* San Diego: Academic Press, 329 - 361.

[McCloskey and Cohen, 1989] McCloskey,M. and Cohen,N.J., Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In Bower,G.H. (Ed.) The Psychology of Learning and Motivation: Volume 23. New York: Academic Press, 109 - 164.

[Murphy and Aha, 1994] Murphy,P.M. and Aha,D.W., UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science.

[Murre, 1992] Murre,J.M.J., *Learning and Categorization in Modular Neural Networks.* Hillsdale, NJ: Earlbaum.

[Ratcliff, 1990] Ratcliff,R., Connectionist Models of Recognition Memory: Constraints Imposed by Learning and Forgetting Functions. *Psychological Review, 97,* 285-308.

[Robins, 1995] Robins,A.V. Catastrophic Forgetting, Rehearsal, and Pseudorehearsal. *Connection Science, 7,* 123 - 146.

[Robins, 1996] Robins,A.V. Consolidation in Neural Networks and in the Sleeping Brain. *Connection Science, 8,* 259 - 275.

[Robins and McCallum, 1997] Robins,A.V. and McCallum,S., Pseudorehearsal in Hopfield Type Networks (Tech. Rep. *AIM-36-97-3*). Otago, New Zealand: Otago University, Department of Computer Science. (Submitted to Connection Science)

[Rumelhart *et al.*, 1986] Rumelhart,D.E. Hinton,G.E. and Williams,R.J., Learning Internal Representations by Error Propagation. In Rumelhart,D.E., McClelland,J.L. and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations.* Cambridge MA: MIT Press.

[Sharkey and Sharkey, 1995] Sharkey,N.E. and Sharkey,A.J.C., An Analysis of Catastrophic Interference. Connection Science, 7, 301 - 329.