ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 129-134

One or two hidden layers Perceptrons

Mercedes Fernández †, Carlos Hernández †.

† Department of Computers.
Jaume- I University.
Campus Penyeta Roja
12071 CASTELLON
SPAIN
E-mail: espinosa@inf.uji.es

Abstract. In this paper we present an experimental comparison of the generalization capability of one, and two hidden layers perceptrons. We have used seven different real world problems in order to measure the generalization of both architectures. For each problem and each architecture, it is carefully selected by a trial and error procedure the minimal network which solves the problem and several runs with different initial conditions are obtained in order to get an average performance. According to our results, the generalization capability of a one hidden layer perceptron is better than the one of two hidden layer perceptron; furthermore two hidden layer perceptrons are more prone to fall into bad local minimum.

1. Introduction.

The research on the different capabilities of one and two hidden layer multilayer feedforward neural networks has received a lot of attention recently.
In this sense, there is a theorem which guaranties that a one hidden layer perceptron can solve any problem with an appropriate number of neurons in its hidden layer, so at first, there is no reason why we must think of using two hidden layers perceptrons. However, it is still an open problem to decide an architecture. For instance, we do not know whether a two hidden layer perceptron might have a better generalization capability than a one hidden layer network or if we can get a simpler model, i.e., less number of weights, by using a two hidden layer perceptron.
In [1], Babri and Tong presents a research on the sensitivity of the network output to weight perturbations for one and two hidden layers feedforward neural networks. From this research it is clear that during the initial states of learning, the sensitivity of a two layer neural network is greater than the one of one hidden layer network and can lead to an unstable learning. Two methods for dealing with the problem are proposed, an appropriate weight initialization and a modification of the transfer function.
In [2], it is shown that a three-layered feedforward network with N-1 hidden units can give any N input-target relations exactly, and that a four-layered network can be constructed to give N input-target relations with a negligible small error using only (N/2)+3 hidden units. After this result, it is easy to think that four layer neural networks with less neurons and weights can be obtained. However, the main question would be whether the training algorithm is able to obtain such a minimal network.

In [3] there is a comparison between one and two layer perceptrons in the context of nonlinear real-valued approximation, the mapping between automobile engine control variables and performance parameters. The comparison is performed for networks with the same number of hidden units, and in the paper it is concluded that four-layered networks (two hidden layer ones) with more nodes in the first hidden layer than in the second hidden layer out-perform the three layered network in accuracy and training.
In [4] and [5], we can find two experimental comparison of the generalization capability of one and two hidden layers feedforward neural networks; the comparisons are performed for classification problems. In [5], the test problems were artificial data of gaussian distributions, and they concluded that the performance of one and two hidden layer perceptrons is similar, however they pointed out that a four layer network (two hidden layer one) is more prone to fall into bad local minimum. In [4] two problems were used for the test, the classification among twelve Chinese numerals and the classification among the ten Arabic numerals. In this case, the conclusions were that the performance of one hidden layer perceptron is better than the one of two hidden layer perceptron.
Unfortunately, in these papers networks with three and four layers and the same complexity, i.e. the same number of weights, are compared. We should point out that this is not a good comparison because the task of a neural network designer is to get the minimal network, which solves the problem well. A proper comparison would be to compare the minimal one hidden layer network with the minimal two hidden layer one, and this is the objective of this paper.

2. Experimental Results.

We have selected seven problems for this comparison, the problems are public and can be found in the UCI repository of machine learning databases. We give a brief description of these problems.

Image Segmentation Data (IS): This problem has 19 continuous attributes, 7 classes, 1500 training instances and 811 test instances.

Credit Approval (CA): This problem concerns card applications. It has 15 nominal and continuous attributes, 2 classes, 453 training instances and 200 test instances.

Pima Indians Diabetes (PI): This problem has 8 attributes, 2 classes, 518 training instances and 250 test instances.

The Monk's Problems (MO1, MO2): We have used two of the three monk's problems. These problems were the basis of the first international comparison of learning algorithms. They are two problems with six attributes and two classes, 332 training instances and 100 test instances.

Display 1 (D1): This problem contains seven attributes, the seven segments of a light-emitting LED display, and 10 classes, the set of decimal digits. Each attribute value has a 10% probability of having its value inverted.

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 129-134

Display 2 (D2): It is the D1 problem, but additional seven irrelevant attributes are added to the instance space. It has 900 training instances and 2000 test instances.

For each problem, we obtained the minimal one hidden layer perceptron by a trial and error procedure and using cross-validation. We also obtained the minimal two hidden layer network by the same procedure and using an exhaustive search between the structure with N hidden nodes in the first layer and two hidden nodes in the second layer, and the structure of two hidden nodes in the first layer and N hidden nodes in the second layer; where N is the number of hidden nodes for the minimal one hidden layer network. In some cases, there were several candidates with a different structure for the minimal network, in these cases we have kept all the candidates.
For the case of two hidden layers networks we have followed the suggestions of paper [1] in order to improve the learning convergence, we have used a modified transfer function for the neurons:

$$f(x) = \frac{1-2\cdot\varepsilon}{1+e^{-x}} + \varepsilon$$

with epsilon equal to 0.02, and weights initialization uniformly distributed in intervals of width 0.2, centered at –0.5 and 0.5.
The training algorithm for one and two hidden layers networks was conjugate gradient with restarts and learning step adaptation [6]. We used this training algorithm in order to diminish the computational burden.
After obtaining the minimal structure, we trained ten neural networks of each type (one and two hidden layers) for each problem. We wanted to obtain an average result with an error, and make the result not depend on the initial conditions.
The results for the seven problems are shown in the following table

Table 1. Experimental Results.

Problem	**One Hidden Layer**			**Two hidden Layers**		
	Structure	Number of Weights	Percentage Correct	Structure	Number of Weights	Percentage Correct
IS	19-13-7	358	91.0±0.5	19-7-5-7	222	73±2
CA	15-9-2	164	86.6±0.6	15-5-3-2	106	81±4
CA				15-6-2-2	116	84±3
PI	8-15-2	167	74.9±0.3	8-14-4-2	196	73±1
PI				8-9-9-2	191	74±1
MO1	6-5-2	47	88±5	6-3-5-2	53	77±5
MO1				6-5-3-2	61	78±4
MO2	6-14-2	128	70.5±1.5	6-6-4-2	80	68.1±1.5
D1	7-15-10	280	71.9±1.3	7-6-6-10	160	54±8
D2	14-21-10	535	70.9±0.8	14-13-10-10	445	62±6
D2				14-17-7-10	461	69.6±0.7
D2				14-20-4-10	434	50±9

In table 1, it is shown the network structure for one and two hidden layers, the number of free parameters, weights and thresholds, of the structure and a performance measure, the percentage correct in the test set.
In the case of two hidden layer we have kept several minimal structures in the problems *CA*, *PI*, *MO1*, *D2*. The performance was similar in the cross-validation set and the number of free parameters is also similar, so we decided to choose several minimal networks and test all of them.
From the results we can see that the number of free parameters of the minimal network depends on the problem but it is usually less in the case of two hidden layers networks (problems *IS*, *CA*, *MO2*, *D1* and *D2*).
We can also see that *the generalization capability of one hidden layer network is in general better than the one of two hidden layers networks*. We got a better result for one hidden layer networks in the problems *IS*, *CA*, *MO1*, *D1*, and in the rest of the problems the performance is nearly the same and undistinguishable within the errors.
We found that the variability in the performance results is greater in the case of two hidden layers networks. We can see this from the results by observing the error, we trained ten networks for each case and the error in the case of two hidden layers networks is greater than in the case of one hidden layer network.
Finally, in some problems like *MO1* it was really difficult to get ten properly trained networks, and we had to trained more than ten networks because there were a lot of networks which did not converge.

3. Conclusions.

We have presented an experimental comparison of the generalization capability of one and two hidden layers perceptrons. We have used seven different real world problems in order to measure the generalization of both architectures.
We selected for each problem and each architecture (one and two hidden layers) the minimal network by using a carefully trial and error procedure.
We have trained ten networks for each architecture and problem in order to avoid dependency on initial conditions and get an error in the results.
From our results, we can conclude that the generalization performance of one hidden layer network is better than the one of two hidden layers networks in the case of classification problems. So with the training algorithms we have, there is no reason for using two hidden layers perceptrons.
It was also observed that the convergence of two hidden layers networks is quite bad in some problems and we may obtain many networks which do not converge.

References.

1. Babri, H.A.; Tong, Y., "Deep Feedforward Networks: Application to Pattern Recognition", Proceedings of the 1996 International Conference on Neural Networks, vol. 3, pp. 1422-1426, 1996.

2. Tamura, S.; Tateishi, M., "Capabilities of a Four-Layered Feedforward Neural Network: Four Layers Versus Three", IEEE Trans. on Neural Networks, vol. 8, no. 2, pp. 251-255, March 1997.

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 129-134

3. Xiao, J.; Chen, Z.; Cheng, J., "Structure Study of Feedforward Neural Networks for Approximation of Highly Nonlinear Real-valued Functions", Proceedings of the 1996 International Conference on Neural Networks, vol. 1, pp. 258-263, 1996.

4. Saratchandran, P., "Effect of Hidden Layers on Generalization Properties of Feedforward Neural Networks", Neural Parallel & Scientific Computations, vol. 1, no. 2, pp. 227-240, 1993.

5. Villiers, J.; Bernard, E., "Backpropagation Neural Nets with One and Two hidden Layers", IEEE Trans. on Neural Networks, vol. 4, no. 1, pp. 136-141, January 1992.

6. Yu, X.; Chen, G. Cheng, S., "Dynamic Learning Rate Optimization of the Backpropagation Algorithm", IEEE Trans. on Neural Netwroks, vol. 6, no. 3, May 1995.

7. Yamasaki, Masami, "The Lower Bound of the Capacity for a Neural Network with Multiple Hidden Layers", Proceeding of the World Congress on Neural Networks (WCNN'93), vol. 3, pp. 544-547, 1993.

8. Cosnard, M.; Koiran, P.; Paugam-Moisy, H., "A Step Toward the Frontier between One-Hidden–layer and Two-Hidden-layer Neural Networks", Proceedings of 1993 International Joint Conference on Neural Networks, vol. 3, pp. 2292-2295, 1993.