AN EXPERIMENTAL COMPARISON OF THE BAYESIAN YING-YANG CRITERIA AND CROSS VALIDATION FOR SELECTION ON NUMBER OF HIDDEN UNITS IN FEEDFORWARD NETWORKS*

Wing-kai Lam and Lei Xu

Department of Computer Science and Engineering,
The Chinese University Hong Kong, Hong Kong

ABSTRACT

Optimizing the number of hidden units in feedforward neural networks is an important issue in learning. Recently, a new criteria on selecting the number of hidden units in feedforward neual networks is proposed by one of the present author, based on the so-called Bayesian Ying-Yang (BYY) learning theory. The new criteria can be simply computed during the implementation of backpropagation training. In this paper, the criteria is experimentally studied and compared with the well-known Cross Validation approach. Simulation results show that obtained number of hidden units by the BYY criteria is highly consistent to the minimal generalization error and outperforms the Cross Validation approach.

1. INTRODUCTION

Optimizing the number of hidden unit in feedforward neural networks is important in two perspectives: One is to reduce the computation time and the other is to get a high generalization [1]. There are quite a number of methods already proposed in the literatures of statistics and neural networks. However, most of them tackle the problem by estimating the upper or lower bounds on generalization error and usually difficult to compute in practical implementation. In this paper, we experimentally study a new criteria on selecting the optimal number of hidden units in feedforward neural networks trained by backpropagation. It involves a small computing cost in implementation. This criteria is proposed in [8] by one of the present authors, based on the Bayesian Ying-Yang BYY) learning theory [9,6,7,8]. Experimental results show that the optimal number of hidden units selected by this criteria is highly consistent with the minimal generalization error. The results are compared with the hidden unit number selected by a generalization strategy, Cross Validation [2,4]. The simple computation of the BYY criteria not only outperforms the exhaustive training of the Cross Validation approach in computational cost considerably, but also in choosing the optimal hidden unit number that minimize the generalization error.

This paper is organized as follows. In Section 2, the brief review of the feedforward network architecture with backpropagation training, *Cross Validation* and the BYY (Bayesian Ying-Yang) criteria are given. In Section 3, we present the results of simulations of the criteria through the simplest three-layered feedforward network architecture. The results are compared with minimal generalization error and Cross Validation. Then a final conclusion is given in Section 4.

*THIS WORK WAS SUPPORTED BY HK RGC EARMARKED GRANTS CUHK484/95E AND CUHK 339/96E AND BY HO SIN-HANG EDUCATION ENDOWMENT FUND FOR PROJECT HSH95/02.

2. METHODS USED IN COMPARATIVE STUDY

2.1. Three-layered Feedforward Neural Networks with Backpropagation

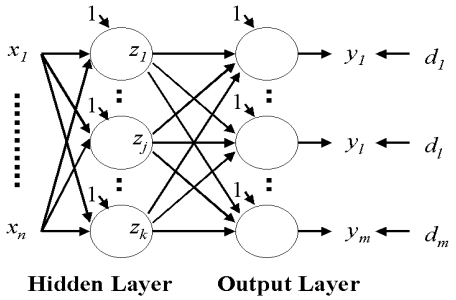


Fig. 1 The three-layered feedforward neural network architecture

The fully-connected three-layered feedforward network we used is shown in Fig. 1. The three layers are input layer, hidden layer and output layer. The bias signal for both hid-

den units and output units are assumed to be embedded into the input vectors of both hidden and output layer. So that the weights connecting the bias signals and units are being trained and updated as usual weights. The activation function f_h of the hidden units is differentiable nonlinear function. In our simulations, $f_h = f_h(W^{h\,T}x) = 1/(1 + e^{-\lambda W^{h\,T}x})$, $x = [x_1, ..., x_n]^T$, $W^h = \{W_1^h, ..., W_k^h\}$ represents the set of hidden unit weight vectors and λ equals to unity. The activation function of the output units, f_o, depends on the desire output (e.g. clustering, function approximation, ..., etc). We applied the same nonlinear function as the hidden units do when simulating the clustering problem, and $f_o = f_o(W^{o\,T}z) = W^{o\,T}z$, $z = [z_1, ...z_k]^T$ for the function approximation problem, and $W^o = \{W_1^o, ..., W_m^o\}$ represents the set of output unit weights.

Backpropagation training rule is a supervised learning rule for adjusting the hidden weights and output weights such that the following error function is minimized over the training set:

$$E(W^h\ ,\ W^o) = \frac{1}{2}\Sigma_{i=1}^{m}(d_i - y_i)^2 \tag{1}$$

The update rule for each layer are being derived by using delta rule directly.

For the $l-th$ of the m output units in the output layer, we have

$$W_l^{o\ new} = W_l^{o\ old} - \rho_o \frac{\partial E(W^h, W^o)}{\partial W_l^{o\ old}} \tag{2}$$

where $\frac{\partial E(W^h, W^o)}{\partial W_l^{o\ old}} = -(d_l - y_l)f_o'\ (W_l^{o\,T}z)z$, ρ_o is the output layer learning rate. Putting all the terms together,

$$W_l^{o\ new} = W_l^{o\ old} + \rho_o(d_l - y_l)f_o'\ (W_l^{o\,T}z)z \tag{3}$$

For the $j-th$ of the k hidden units in the hidden layer, we have,

$$\begin{aligned} W_j^{h\ new} &= W_j^{h\ old} - \rho_h \frac{\partial E(W^h, W^o)}{\partial W_j^{h\ old}} \\ &= W_j^{h\ old} - \rho_h \frac{\partial E(W^h, W^o)}{\partial z_j}\frac{\partial z_j}{\partial net_j^h}\frac{\partial net_j^h}{\partial W_j^{h\ old}} \end{aligned} \tag{4}$$

where $\frac{\partial E(W^h, W^o)}{\partial z_j} = -\Sigma_{l=1}^{L}\{(d_l - y_l)f_o'\ (W_l^{o\,T}x)w_{lj}\}$, $\frac{\partial z_j}{\partial net_j^h} = f_h'\ (W_j^{h\,T}x)$, $\frac{\partial net_j^h}{\partial W_j^{h\ old}} = x$, ρ_h is the hidden layer learning rate. Putting all terms together,

$$\begin{aligned} W_j^{h\ new} &= W_j^{h\ old} \\ &+ \rho_h\Sigma_{l=1}^{L}\{(d_l - y_l)f_o'\ (W_l^{o\,T}z)w_{lj}\}f_h'\ (W_j^{h\,T}x)x \end{aligned} \tag{5}$$

where w_{lj} is the synaptic weight connecting the $l-th$ output unit and the $j-th$ hidden unit. The iterative backpropagation learning procedures for three-layered feedforward neural networks are stated as Eq. 3 and Eq. 5 respectively. The learning continues unit the algorithm converges. For details, refer to (e.g. [3,5]).

2.2. Cross Validation [2,4]

Based on [2,4], the whole data set is split into three parts: training set, validation set and testing set. The training set is used to determine the internal weights of the networks. The validation set is used to decide when to stop training. The testing set is used for estimating the expected performance validation (generalization) of the resulted networks. Usually, Cross Validation is applied to stop-early training. We adapt the stop-early phenomena in choosing number of hidden units. Training starts with the training set with a small number of hidden units. After certain number of iterations (assumed larged enough for adapting the problem), the validation set is used for testing performance. The process repeats with one hidden unit increases and stops when the validation on the validation set cease to improve. Finally the testing set is used for testing performance of the trained networks.

2.3. The Bayesian Ying-Yang Learning Criteria [8]

Based on the so-called Bayesian Ying-Yang Learning Theory [9,6,7,8], the following criteria is proposed for selecting the optimal number of hidden units [8],

$$\begin{aligned} J(k) &= 0.5ln(E_{d|x}) \\ &- \frac{1}{N}\Sigma_{i=1}^{N}\Sigma_{j=1}^{k}[z_j\ ln(z_j) + (1 - z_j)ln(z_j)] \end{aligned} \tag{6}$$

where $E_{d|x} = \Sigma_{(x,d)\in D_{x,d}}\ ||d - f_o(W^{o\,T}f_h(W^{h\,T}x))||^2$, $d = \{d_1\ , ...,\ d_m\}$ and k , z_j's are output at the hidden units as in the Sec. 2.2. is the number of hidden units specified in the hidden layer. Eq. 6 is tested with various k . The k for the corresponding minimum point is the optimal number of hidden units.

3. SIMULATION RESULTS

The simulation is based on the architecture shown in Fig. 1. The training procedures those in Section 2.1.1 and 2.1.2. Two types of problem are simulated for testing, clustering and function approximation. For clustering, we have two benchmark data set, IRIS and HAYES-ROTH. For the function approximation problem, we have generate two curves.

3.1. Clustering Problem

The 4-dimensional IRIS data set consists of 3 clusters with 50 data points each. The first two and last two dimensions are plotted in Fig. 2 and 3. The 5-dimensional HAYES-ROTH data set also contains 3 clusters with 44 data points each.

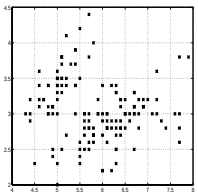


Fig. 1 1, 2 dimensions of IRIS

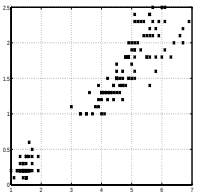
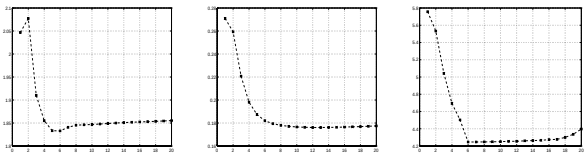


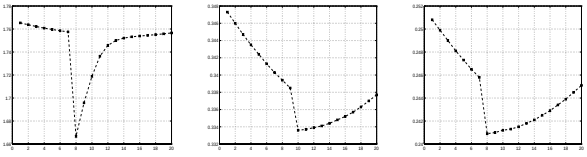
Fig. 2 3, 4 dimensions of IRIS

For each cluster in any data set, 1/4 is randomly sampled out as training set, another 1/4 for validation set and the rest 1/2 is for testing set. The validation set is only used for Cross Validation. For each of the algorithms, we extensively specify the number of hidden units starting from 1 to 20. Both hidden unit learning rate ρ_h and output unit learning rate ρ_o are fixed at 0.03. This number is chosen due to exhausive trials on different number for the promising balance between convergent speed and training error on a 4-D artificial data set similar to IRIS data set. The number of iterations for each training is 100. The results for each data set are shown in Fig. 4, 5 respectively.



(a) BYY Criteria (b) Cross Validation (c) Minimal Generalization

Fig. 4 The resulted minimums detected by (a) BYY Criteria, (b) Cross Validation and (c) Generalization Error on IRIS data set



(a) BYY Criteria (b) Cross Validation (c) Minimal Generalization

Fig. 5 The resulted minimums detected by (a) BYY Criteria, (b) Cross Validation and (c) Generalization Error on HAYES-ROTH data set

We perform 10 similar tests with different sampled training, validation and testing data set. The results are summarized in Table 1.

Trial	Criteria	C.V.	Generalize
1	6	12	6
2	6	12	6
3	8	6	8
4	6	8	6
5	7	8	7
6	7	8	7
7	8	10	8
8	6	11	6
9	6	10	6
10	6	9	6

(c) IRIS Data Set

Trial	Criteria	C.V.	Generalize
1	8	8	8
2	8	9	8
3	7	10	7
4	8	10	8
5	8	10	8
6	7	8	7
7	7	9	7
8	8	9	8
9	8	9	8
10	8	9	8

(d) HAYES-ROTH Data Set

Table 1 The resulted number of hidden unit selected by the BYY criteria (Criteria), Cross Validation (C.V.) and minimal generalization error (generalize) for various data set with different trials

As noticed from Table 1, the number of hidden unit selected by the BYY criteria always match the minimal testing error. However, it is not Cross Validation can. Moreover, the number selected by Cross Validation is usually larger.

3.2. Function Approximation Problem

We generate two different curves in the range $[0,\ 10]$. They are shown in Fig. 10 and 11 respectively.

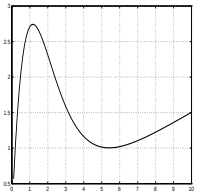


Fig. 10 The Curve 1

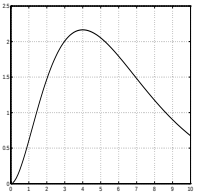
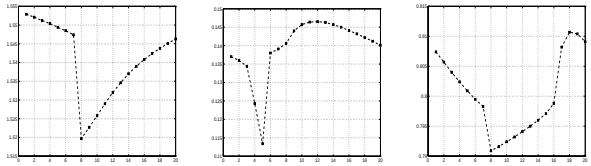


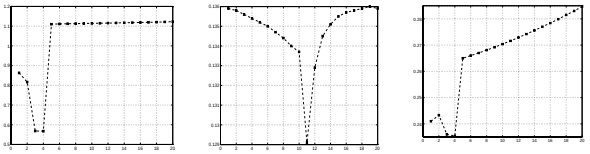
Fig. 11 The Curve 2

We randomly sample 10 points in range $[0,\ 10]$ for training set. Then 10 points for validation set and 20 points for testing set. Again, the validation set is only for Cross Validation. The number we choose for ρ_h and ρ_o are both 0.01. This is again result after an exhausive testing on another artificial curve generated in the same range. The results are shown in Fig. 12 and 13 respectively.



(a) BYY Criteria (b) Cross Validation (c) Minimal Generalization

Fig. 12 The resulted minimums detected by (a) BYY Criteria, (b) Cross Validation and (c) Minimal Generalization Error on curve 1



(a) BYY Criteria (b) Cross Validation (c) Minimal Generalization

Fig. 13 The resulted minimums detected by (a) BYY Criteria, (b) Cross Validation and (c) Minimal Generalization Error on curve 2

Again, we perform ten tests with different sampling for training, validation and testing data set, the results are summarized in Table 2.

Trial	Criteria	C.V.	Generalize
1	8	10	8
2	8	8	8
3	7	8	7
4	8	7	8
5	8	9	8
6	7	9	7
7	7	9	7
8	7	9	7
9	7	9	7
10	7	9	7

(a) Curve 1

Trial	Criteria	C.V.	Generalize
1	4	5	4
2	4	5	4
3	4	5	4
4	5	5	5
5	5	5	5
6	5	5	5
7	4	5	4
8	4	6	4
9	5	4	5
10	5	5	5

(b) Curve 2

Table 2 The resulted number of hidden unit selected by the BYY criteria (Criteria), Cross Validation (C.V.) and minimal generalization error (generalize) for various curves with different trials

As noticed from Table 2, similar to the clustering case, the number of hidden unit selected by the criteria always match the minimal testing error. It is not the case for Cross Validation. The number selected by Cross Validation is usually larger.

4. CONCLUSIONS AND FUTURE WORKS

The BYY criteria for selecting an optimal number of hidden unit is experimental proven that can select the hidden unit number with minimzed generalization error. It outperforms Cross Validation in selecting the appropriate hidden unit number for both clustering and function approximation.

5. REFERENCES

[1] Gorman, R. P., & Sejnowski, T. J. (1988). Analysis of hidden units in a layered network trained to classify sonar targets. Neural Networks, 1, 75-89.

[2] Stone, M. (1978). Cross-validation: A review, Mathematische Operationsforschung Statistischen, 9, 127-140.

[3] Hassoun, H. Mohamad, (1995). Fundamentals of ARTIFICIAL NEURAL NETWORKS, MIT Press.

[4] Finnoff, W. (1993). Diffusion approximations for the constant learning rate backpropagation algorithm and resistance to local minima, in Advanced in Neural Information Processing Systems 5 (Denver, 1992), S.J. Hanson, J.D. Cowan, and C.L. Giles, eds., pp459-466. Morgan Kaufmann, San Mateo, California.

[5] Rumelhart, D.E., Hinton, G.E., and Williams, R.J. (1986). Learning internal representations by error propagation, in Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. I, D.E. Rumelhart, J.L. McClelland, and the PDP Research Group eds. MIT Press, Cambridge, Mass.

[6] Xu, L., (1997a). Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach: (I) Unsupervised and Semi-Unsupervised Learning. An invited book chapter, S. Amari and N. Kassabov eds., Brain-like Computing and Intelligent Information Systems, 1997, New Zealand, Springer-Verlag, pp241-274.

[7] Xu, L., (1997b). Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach: (II) From Unsupervised Learning to Supervised Learning and Temporal Modelling. Invited paper, Lecture Notes in Computer Science: Proc. of International Workshop on Theoretical Aspects of Neural Computation, May 26-28, 1997, Hong Kong, Springer-Verlag, pp25-42.

[8] Xu, L., (1997c). Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach: (III) Models and Algorithms for Dependence Reduction, Data Dimension Reduction, ICA and Supervised Learning. Lecture Notes in Computer Science: Proc. of International Workshop on Theoretical Aspects of Neural Computation, May 26-28, 1997, Hong Kong, Springer-Verlag, pp43-60.

[9] Xu, L., (1995). Ying-Yang Machine: A Bayesian-Kullback scheme for unified learnings and new results on vector quantization. Keynote talk, Proceedings of International Conference on Neural Information Processing (ICONIP95), Oct. 30-Nov. 3, pp977-988 (1995).