ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 65-70

A resampling and multiple testing-based procedure for determining the size of a neural network

Andrés Yáñez Escolano, Elisa Guerrero Vázquez,
Pedro L. Galindo Riaño, Joaquín Pizarro Junquera

Universidad de Cádiz, Dpto. Lenguajes y Sist. Informáticos
Grupo de Investigación *"Sistemas Inteligentes de Computación"*
C.A.S.E.M. 11510 – Puerto Real (Cádiz), Spain
{andres.yaniez, elisa.guerrero, pedro.galindo, joaquin.pizarro}@uca.es

Abstract. One of the most important difficulties in using neural networks for a real-world problem is the issue of model complexity, and how affects the generalization performance. We present a new algorithm based on multiple comparison methods for finding low complexity neural networks with high generalization capability.

Keywords: network size, regression, multiple comparison procedures, generalization, statistical tests

1. Introduction

The task of learning from examples is to find an approximate definition for an unknown function $f(x)$ given training examples of the form $<x_i, f(x_i)>$. The goal of network training is not to learn an exact representation of training data itself, rather to build a statistical model of the process which generates the data. So the network will exhibit good generalization, that is, make good predictions of new inputs [1].

It is well known that choosing how many hidden units should have a neural network is a crucial matter in order to achieve good generalization behaviour. A network with too few hidden units gives poor predictions for new data, because it has too little flexibility (it has a large bias). On the other hand, too many hidden units will give us poor generalization because it fits too much to the noise on training data (it has a large variance). The best generalization is therefore obtained when the trade-off between poor approximation and overfitting is achieved.

In this paper we propose a statistical method in order to obtain a neural network with a low number of hidden units, but with a great power of generalization. This method, which is based on previous works ([6], [9]), suggests using multiple comparison techniques and repeated measures procedures.

The plan of the paper is as follows. In section 2 we briefly introduce basic concepts, such as power of a test, parametric and non-parametric tests, repeated measures and multiple comparisons procedures. Section 3 describes the proposed methodology and the results obtained when applied on RBF and BP networks is shown in Section 4.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 65-70

2. Basic terminology

The first step in the decision-making procedure is to state the *null hypothesis* (H_0), which says that any difference is due to chance. It is usually formalized for the express purpose of being rejected. If it is rejected, the *alternative hypothesis* (H_1) is supported. Our null hypothesis is that two networks with different complexity will give similar performance on unseen data.

If a statistical test yields a value whose associated probability of occurrence under H_0 is equal to or less than some small probability, which is called level of significance (α), we will reject H_0.

Each statistical test is based on certain assumptions about the population from which the data are drawn. If a particular statistical test is used to analyze data collected from a sample that does not meet the expected assumptions, then the conclusions drawn from the results of the test will be flawed ([2], [10]). *Parametric tests* assume that the data were sampled from a particular form of distribution, while *nonparametric tests* do not make such assumption. Parametric tests are more powerful and should be used if possible (the *power* of a test is the probability of rejecting H_0 when it is in fact false).

There are two basic designs for comparing k samples ([5], [8], [10], [11]). In the first design, k *independent samples*, not necessarily of the same size from each population are analyzed. The second design, k *related samples* of equal size are matched according to some criterion which may affect the values of the observations. For example, in our method we analyze the errors of k neural networks with different complexities which are trained with the same training samples in order to discover if the differences between networks are significant. Statistical tests for related samples are shown in Table 1.

	Parametric		**Nonparametric**
k	**Test**	**Assumptions**	**Test**
2	t paired test	• The difference scores are independently drawn from a normal distribution	Wilcoxon matched pairs test
> 2	Repeated measures ANOVA test	• The k error measures for each training sample set are drawn from a normal distribution • Compound symmetry / sphericity [4] about the variance-covariance matrix	Friedman test

Table 1: Statistical parametric and non-parametric tests for k related samples.

When repeated measures ANOVA or Friedman tests are significant, it indicates that at least two of the groups in the analysis are significantly different, but not which are. At that point multiple individual comparisons should be computed on pairs of groups. The problem is that each comparison is done with the level of significance set at probability of α, and, if a test comparison is made on each of p comparisons, the experimentwise level of significance will be $1-(1-\alpha)^p$. Statistical

methods to compare three or more groups while controlling the experimentwise error are called *Multiple Comparison Procedures* (MCP). In this study, two statisticians have been applied in order to control the experimentwise error rate:

- Nemenyi test [11] is a medium power nonparametric MCP analogous to Tukey test, using rank sums instead of means.
- t paired and Wilcoxon matched pairs tests with Bonferroni correction [7] are comparison procedures whose power is highly dependent on the number of comparisons made.

In the proposed strategy, Nemenyi test was first applied in order to minimize the number of models which take part in the computation of the Bonferroni factor.

3. Methodology description

In this section we propose a stepwise methodology to model selection that makes use of MCP to keep the experimentwise error under control and exhibits good generalization behaviour under small and large sample size situations.

The steps of the proposed method for a given dataset may be outlined as follows:

1) Take the whole data set and create b ($b \geq 30$) training sets by a resampling technique [3] taking the rest of the instances for testing.
2) Train models (neural networks) with a degree of complexity (hidden units) ranging from 1 to k and generate b test error measures per model.
3) Apply Nemenyi test on error values and return the set S of models whose errors are not significantly different from that corresponding to the model with minimum error median.
4) If size(S) = 1, return S, that is to say, the model with minimum error median and exit.
5) If repeated measures Anova assumptions are met on S, apply this test, otherwise apply Friedman test.
6) If the null hypothesis is not rejected, select the simplest model of S and exit.
7) Apply multiple comparison procedure (t paired or Wilcoxon tests with Bonferroni correction) on S and select the least complex model from the subset of models that are not significantly different from the model with minimum error median.

Some remarks concerning the method should be highlighted. First, we propose to use error medians instead of error means to reduce outlier problems. In any case, as training sample increases, error medians and error means tend to approach each other. Second, let us note that all tests have been applied using α = 5 %

4. Experimental results

After introducing the main aspects of the procedure, we now explore its behaviour via a simulation study using two different neural networks, radial basis function and two-layer feedforward networks. RBF were designed as having one hidden layer for which the combination function is the Euclidean distance between

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 65-70

the input vector and the weight vector. We use the *exp* activation function. The width of the basis functions has been set to $\|max(x_i - x_j)\|/\sqrt{2n}$ where n is the number of kernels. The second layer is a linear mapping from the RBF activations to the output nodes. BP networks were designed as having hyperbolic tangent sigmoid transfer function in the hidden layer and linear transfer function in the output layer and were trained using Levenberg-Marquardt algorithm.

In order to illustrate our method, we carried out a whole series of experiments on simulated data sets. A total of 30 data sets of several sample sizes were generated according to the following experimental functions:

$$y = 10sin(2x + 6) + \varepsilon\ ,\ x \in (-2,+2) \quad (1)$$

$$y = -0.2x^4 + 1.5x^3 - 6x + 3 + \varepsilon\ ,\ x \in (-2,+2) \quad (2)$$

where ε is gaussian noise with zero mean and variance equal to two per cent of generalization sample standard deviation.

We trained each model with every generated data set, and estimated the generalization errors applying each network to a large size (10000 samples) test set, which gave us good estimations of the generalization capability of each trained model. We then applied the stepwise strategy to each dataset in order to compare the performance of the proposed methodology.

The results are summarized in tables 2, 3, 4 and 5. The internal structure of each table is the following:

- data set size column: size of simulated training data
- generalization columns: results of applying the method on generalization data. First column includes the model with minimum error median and the second one, the least complex model from those not significantly different from the previous one.
- resampling columns: frequency of selection of each model when applying the method on *b*=50 bootstrapped samples for each dataset. Both columns have the same structure as described in the previous paragraph.

	Generalization		**Resampling**	
Data set size	**Min**	**Lowest complexity**	**Min**	**Lowest complexity**
75	5	5	5 (40,00 %) 6 (30,00 %) 7 (10,00 %) 8 and 9 (6,67 %) 10 and 15 (3,33 %)	4 (76,67 %) 5 (20,00 %) 6 (3,33 %)
500	5	5	5 (33,33 %) 6 (30,00 %) 7 (26,67 %) 8 (6,67 %) 13 (3,33 %)	5 (86,67 %) 6 (10,00 %) 7 (3,33 %)

Table 2: Number of hidden units in RBF networks trained with samples generated from function (1).

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 65-70

	Generalization		Resampling	
Data set size	**Min**	**Lowest complexity**	**Min**	**Lowest complexity**
75	3	3	3 (63,33 %) 4 (23,33 %) 5 (6,67 %) 6 (6,67 %)	2 (10,00 %) 3 (86,67 %) 4 (3,33 %)
500	3	3	3 (56,67 %) 4 (23,33 %) 5 (10,00 %) 6 (6,67 %) 9 (3,33 %)	3 (83,33 %) 4 (10,00 %) 5 (6,67 %)

Table 3: Number of hidden units in BP networks trained with samples generated from function (1).

	Generalization		Resampling	
Data set size	**Min**	**Lowest complexity**	**Min**	**Lowest complexity**
75	7	5	4 (6,67 %) 5 (26,67 %) 6 (36,67 %) 7 y 8 (13,33 %) 9 (3,33 %)	3 (3,33 %) 4 (40,00 %) 5 (50,00 %) 6 (6,67 %)
500	7	6	5 (3,33 %) 6 and 7 (33,33 %) 8 (13,33 %) 9 (6,67 %) 10, 11 and 13 (3,33 %)	5 (10,00 %) 6 (53,33 %) 7 (30,00 %) 8 and 11 (3,33 %)

Table 4: Number of hidden units in RBF networks trained with samples generated from function (2).

	Generalization		Resampling	
Data set size	**Min**	**Lowest complexity**	**Min**	**Lowest complexity**
75	3	3	2 (10,00 %) 3 (50,00 %) 4 (26,67 %) 5 (6,67 %) 6 and 8 (3,33 %)	2 (30,00 %) 3 (66,67 %) 4 (3,33 %)
500	3	3	3 (46,67 %) 4 (16,67 %) 5 (20,00 %) 6 (10,00 %) 7 (6,67 %)	3 (80,00 %) 4 (13,33 %) 5 (6,67 %)

Table 5: Number of hidden units in BP networks trained with samples generated from function (2).

From tables 2 through 5 we have found that the behaviour of the proposed methodology correlates very well with the actual performance on generalization data. We found that taking the minimum of a resampling technique produces a systematic overfitting, which is very strong independently of sample size. The proposed methodology exhibits a slight underfitting with small-sample problems,

which is not very dangerous, because in this situation, the gross errors always come from overfitted decisions. On the other hand, when the sample size is large enough, the procedure corrects the overfitting tendency when taking the minimum. The general trend we have observed is remarkably consistent across the different networks considered in the study.

5. Conclusions

In this work we have presented a statistical method which is based on repeated measures and multiple comparisons procedures. The goal of this method is to find a neural network having the best generalization performance with a minimum complexity. We have shown the usefulness of this methodology in order to overcome the overfitting and underfitting problems.

Future work will address the application of the methodology to other non-linear models, such as decision trees and different neural networks, as well as consider other MCP and resampling techniques to improve the performance, specifically in low sample size situations.

References

1. Bishop, C. M.: Neural network for pattern recognition. Clarendon Press-Oxford (1995)
2. Don Lehmkuhl, L.: Nonparametric statistics: methods for analyzing data not meeting assumptions required for the application of parametric tests. Journal of prosthetics and orthotics Vol. 8, num. 3, pp.105-113 (1996)
3. Efron, B., Tibshirani, R.: Introduction to the bootstrap, Chapman & Hall (1993)
4. Field, A. P.: A bluffer's guide to sphericity. Newsletter of the Mathematical, Statistical and computing section of the British Psychological Society, 6 (1), pp. 13-22 (1998)
5. Girden, E. R.: Anova repeated measures, Sage Publications (1993)
6. Guerrero, E., Yañez, A., Galindo, P., Pizarro, J.: Repeated measures multiple comparison procedures applied to model selection in neural networks. Proceedings of 6th International Work-Conference on Artificial and Natural Neural Networks Vol 2, pp. 88-95, Granada, Spain (2001)
7. Hsu, J.C. Multiple comparisons. Theory and methods, Chapman & Hall. (1996)
8. Minke, A.: Conducting repeated measures Analyses: Experimental design considerations, Annual meeting of the Southwest Educational Research Association, Austin (1997)
9. Pizarro, J. Guerrero, E., Galindo, P.: A statistical model selection strategy applied to neural networks. Proceedings of the European Symposium on Artificial Neural Networks Vol 1, pp. 55-60, Bruges (2000)
10. Siegel, S., Castellan, N. J.: Nonparametric statistics for the Behavioral Sciences, 2nd ed. McGraw-Hill (2000)
11. Zar, J. H.: Biostatistical Analysis, Prentice Hall (1996)