LEARNING AND VALIDATION IN NEURAL NETWORK ENSEMBLES

P. M. Granitto, H. D. Navone, P. F. Verdes and H. A. Ceccatto

Instituto de Física Rosario (CONICET-UNR), Blvd. 27 de Febrero 210 Bis, 2000 Rosario

Abstract: Ensembles of artificial neural networks (ANN) have been used in the last years as classification/regression machines, showing improved generalization capabilities that outperform those of single networks. We propose here a simple method for learning and validation in regression/classification ensembles of ANN that leads to overtrained aggregate members with an adequate balance between accuracy and diversity. The algorithm is favorably tested against other methods recently proposed in the literature, producing an improvement in performance on the standard statistical databases used as benchmarks.

Ensemble techniques improve the generalization capabilities of single ANN [1]. However, aggregation is effective only for accurate and diverse ensemble members, *i.e.*, networks with good individual performances and independently distributed predictions for the test points. We provide here a simple way of generating an ANN ensemble with members that have a good compromise between accuracy and diversity. The method essentially amounts to the sequential aggregation of individual predictors where, unlike standard techniques that combine individually optimized ANN [2], the learning process of a new member is validated by the *overall* aggregate prediction performance. That is, the early-stopping method is applied by monitoring the generalization capabilities of the previous-stage aggregate predictor *plus* the network being currently trained. In this way we retain the simplicity of independent network training and only the validation process becomes slightly more involved, leading in general to some controlled overtraining ("late-stopping") of the individual networks.

We propose to train and validate members of ANN aggregates by the following procedure:

Step 1: Generate a training set T_1 by a bootstrap re-sample [3] from dataset D and a validation set V_1 by collecting all instances in D that are not included in T_1. Produce a model f_1 by training a network on T_1 until a minimum $E(V_1\,;f_1)$ of the generalization error on V_1 is reached.

Step 2: Generate new training and validation sets T_2 and V_2 respectively, using the procedure described in step 1. Produce a model f_2 by training a network until the generalization error on V_2 of the *aggregate* predictor $\Phi_2 = ½\ (f_1+f_2)$ reaches a minimum $E(V_2\ ;\ \Phi_2\ )$. In this step the parameters of model f_1 remain constant and the model f_2 is trained with the usual (quadratic) cost function on T_2.

Step 3: Iterate the process until an optimal number N_A of models are produced. This optimal number can be estimated by keeping an external validation set or from the behavior of $E(V_n\ ;\ \Phi_n\ )$ as a function of n.

Notice that in this algorithm the individual networks are trained in the usual way, but with a late-stopping method based on the current *ensemble* generalization performance. A careful analysis shows that at every stage the algorithm is seeking for a new diverse model anticorrelated with the current ensemble [4].

We tested the above method in the regression setting by comparing it against a standard bagging technique adapted from [5], a simple early-stopping method of individual networks, and the recently-proposed NeuralBAG algorithm (a description of these methods can be found in [6]). For this comparison we used as benchmarks the Ozone, Boston Housing and Friedman#1 statistical databases. In addition, we applied the method to the well-known sunspot time series and compared the results of our algorithm with those of an optimal ensemble averaging of independently-trained ANN [2]. For this problem, the results obtained here are, to the best of our knowledge, the most accurate ones reported in the vast literature on sunspot prediction.

In order to compare the different methods' performances we used the same training process of individual networks for all of them, changing only the stopping-point selection criterion. We also set

$N_A = 30$ to allow direct comparison with results in [6], although this number of networks is not necessarily optimal for our method. All the results quoted below correspond to the average over 50 independent runs of the whole procedure, without discarding any anomalous case. Notice also that in the tables below the indicated standard deviations only characterize the dispersion in performances due to different realizations of training and test sets; they have no direct relevance in comparing the average performances for different methods, since in each run all methods use the same data.

Dataset	Single	Simple	Benchmark	NBAG	This Work
Ozone	21.55 ± 4.15	18.91 ± 3.21	18.48 ± 3.03	18.72 ± 3.22	18.59 ± 3.20
Boston	19.95 ± 8.87	14.78 ± 6.97	14.50 ± 6.70	14.96 ± 7.40	14.46 ± 6.89
Friedman#1	4.82 ± 1.54	2.49 ± 0.45	2.43 ± 0.38	2.50 ± 0.48	2.32 ± 0.35

Table 1: Mean-squared test errors averaged over 50 runs corresponding to five different algorithms for ensemble learning. The Simple, Benchmark and NBAG algorithms are described in [6]; the results for Single correspond to the average performance of a single ANN. The standard deviations only characterize the performance fluctuations due to different realizations of training and test sets.

>From Table 1 we can see that the average mean-squared error obtained with our method is smaller than the corresponding errors produced by the Simple and NeuralBAG algorithms. Only for the ozone dataset it is slightly bigger than that of the Benchmark algorithm (which uses information contained on a dataset twice as large as the other methods). Furthermore, for Friedman#1 our algorithm produced more than 20% of reduction on the standard deviation of ensemble errors.

Finally, it is of interest to consider the average number of training epochs that individual networks have to be trained before being aggregated to the ensemble. Table 2 gives these figures for the different methods considered and shows that both NeuralBAG and the algorithm here proposed lead to an important overtraining (late stopping) compared to the Simple and Benchmark methods (which essentially correspond to the standard early-stopping method of single networks).

Dataset	Simple	Benchmark	NBAG	This Work
Ozone	2318	2173	3194	3927
Boston	3879	3722	4955	5460
Friedman#1	7640	6935	14692	17510

Table 2: Average number of training epochs of individual networks required by the different aggregation methods considered.

Application: The Sunspot Time Series

Sunspots are dark blotches on the sun whose mechanism for appearance is not exactly known. Yearly averages of the number of sunspots have been recorded since 1700, and this time series has served many times as a benchmark in the statistical literature. Here we will apply the method described in the previous section to the sunspot time series in order to compare its performance with that of an optimal ensemble averaging of independently-trained ANN [2].

We used the records in the period 1921-1955 as the test set and the remaining ones as training set. Results are appraised in terms of the average relative variance

$$ARV_S = (1/\sigma_S)^2 \, \mathrm{E}[(t_i - f(\mathbf{x}_i))^2 | (t_i, \mathbf{x}_i) \in S]$$

where S is either the training or test set and σ_S its standard deviation. Here $\mathbf{x}_i=(s_{i-1}, s_{i-2},.., s_{i-12})$ is an input vector, $t_i=s_i$ the associated target output and s_i the mean annual sunspot number for year i.

For the test set above defined, the best performance reported in the literature correspond to an optimal ANN ensemble averaging [2]. In order to compare with this work, we have considered the same network architecture (12:4:1), learning rate (η=0.001) and maximum number of training epochs (N_E=150k). We generated an aggregate predictor according to the algorithm described above, using N_A=30. The average of 25 independent runs of the whole procedure produced $ARV_{1921\text{-}1955} = 0.0636 \pm 0.0036$, which compares favorably with the corresponding result $ARV_{1921\text{-}1955} = 0.0713$ in [2].

Conclusions

We proposed a simple method for balancing diversity and accuracy of ANN ensemble members. At every stage, the algorithm seeks for a new member that is at least partially anticorrelated with the previous-stage ensemble estimator. This is achieved by applying a late-stopping method in the training process of individual networks, leading to a controlled level of overtraining of the ensemble members. The algorithm retains the simplicity of independent network training and, moreover, it largely reduces the computational burden compared to other algorithms like NeuralBAG [6] or the method proposed in [2] (which require saving the intermediate networks during training, since the selection of stopping points for the ensemble members is performed only at the end of all the training processes). Our method is a stepwise construction of the ensemble, where each network is selected at a time and only its parameters have to be saved. We showed, by comparison with other methods proposed in the literature, that this strategy is effective, as exemplified by the results on three standard statistical benchmarks, the Ozone, Boston Housing and Friedman#1 datasets, and on the sunspot time series.

The results are encouraging and we are presently performing a more extensive check of the algorithm by comparing it with several other strategies for ensemble learning proposed in the literature. We are also applying our method on new databases and different learning tasks, in order to establish its real capabilities and possible weaknesses under varying conditions.

References

[1] A. J. C. Sharkey, Ed., “Combining Artificial Neural Nets”, (Springer-Verlag, London, 1999)

[2] U. Naftaly, N. Intrator and D. Horn, “Optimal ensemble averaging of neural networks”, *Network: Comput. Neural Syst.* **8**, 283-296 (1997)

[3] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap* (Chapman and Hall, London, 1993)

[4] P. M. Granitto, H. D. Navone, P. F. Verdes and H. A. Ceccatto, “Late-Stopping Method for Optimal Aggregation of Neural Networks”, Submitted to *International Journal of Neural Systems* (2001)

[5] L. Breiman, “Out-of-bag estimation”, Technical Report, Statistics Department, University of California at Berkeley (1996)

[6] J. Carney and P. Cunningham, “Tuning diversity in bagged ensembles”, *International Journal of Neural Systems* **10**, 267-280 (2000)