Nonlinear Combination of Neural Networks From a Diverse and Evolving Population

Thomas M. English

Computer Science Department
Texas Tech University
Lubbock, Texas USA 79409-3104
english@cs.ttu.edu
(806) 742-3527

Abstract—The objective is to predict on the basis of recent observations the behavior of a nonlinear dynamical system. An evolutionary algorithm searches a large space of model topologies and precision hyperparameters, fitting limited-precision models to older observations and assessing how well the models generalize to more recent observations. In each generation, topology-precision pairs are ranked according to an inductive criterion favoring validation errors that are not only small but low in redundancy with those of higher-ranked pairs. The fitted models associated with the highest ranks of a generation are combined in a nonlinear fashion. Their predictions are treated collectively as the state vector of the modeled system, and memory of errors in similar states of the past is used to correct and combine predictions in the present. Model combinations of different generations are themselves similarly combined. Empirical results are given for populations of recurrent neural networks modeling chaotic systems.

Keywords—stochastic complexity, time series prediction, model validation, stacked generalization, model combination

1. Introduction

Small sets of observations of a nonlinear dynamical system typically admit of multiple interpretations. In scientific explanation, it is undesirable for several models to fit the data well, because explanation is equated with obtaining a single model that is better than all others. In automated induction of models for prediction, however, it is highly desirable to obtain several good models. If the mutual information of the models' errors is low, then a weighted sum of predictions is likely to be more accurate than the predictions of any one model.

The foundation of the modeling strategy sketched here is evolutionary search for a diverse collection of good models. Under Gaussian assumptions, independence of errors is equivalent to orthogonality of error vectors. The evolutionary algorithm attempts to obtain a collection of models with short, mutually orthogonal error vectors. To be more precise, a model is fitted to less-recent (training) observations of the dynamical system, and the error vector is the sequence of errors in predicting more-recent (validation) observations.

As formulated by Rissanen (1989), the stochastic complexity of a model depends not only upon the number of free parameters, but upon the precision with which the parameters are specified. This motivates the association of precision hyperparameters with model topologies. When a model's fit to the training data is evaluated, noisy realizations of its parameters are used in place of the actual values. The noise is zero-mean and uniform, and is scaled appropriately to model round-off to the precision specified by hyperparameters. Intuitively, generalization is cast as a problem of finding a good topology along with the problem of finding the precision that allows a model of that topology to extract the right level of detail from the training set.

The fitted models associated with the highest ranks of a generation are nonlinearly combined. The predictions of the models are treated collectively as the state vector of the dynamical system (Breeden and Packard 1994). Past state vectors similar to the present vector are identified, and memory of the models' errors in those past states allows estimation of the models' errors in the present state. The variance of a model's errors in the past states is taken as an index of the model's reliability in the present state. That is, the current predictions are adjusted through subtraction of estimated errors, after which a weighted sum of the adjusted predictions is computed. The weights are set inversely proportional to the error variances.

The combination scheme has several parameters of its own (e.g., number of similar past states to use in correction of present predictions). These are tuned to maximize accuracy in prediction of a secondary validation set–the very most recent observations, disjoint from those in the validation set. There may be ambiguity, however, as to which parameter settings are best. Furthermore, each generation of the evolutionary algorithm produces a new set of ranked models. An answer is to combine the model combinations of different generations and of different parameter settings. Expressed in terms of stacked generalization (Wolpert 1992), the level-2 combination of level-1 predictions is similar to the level-1 combination of level-0 predictions.

The following sections give several details about ranking and evolution of level-0 models, and empirical results for two chaotic time series. These results have been reported in (English 1995; English, in press).

2. Evolution of Level-0 Models

All models in level 0 are members of a particular class of recurrent neural networks. This contradicts the stated objective of obtaining a diverse collection of good models, but makes for efficient execution of the evolutionary algorithm on a data-parallel computer, the MasPar MP-1. The algorithm for evolving a population of networks follows.

```
Initialize population of networks
Repeat
    Fit the parameters of each network to the training set
    Compute the validation-set error vector of each network
    Use validation error vectors to rank networks (as described below)
    Select the better-ranked half of the population as parents
    For each parent, generate two offspring through mutation
    Discard parents, include all children in the new population
Until computing resources are exhausted
```

The procedure for fitting networks to the training set is itself a simple evolutionary algorithm (i.e., embedded within the one just described). The networks are evaluated with "noisy" parameters during fitting. During validation, precise connection weights are used. In generation of offspring, the parent's topology, parameters, and precision hyper-parameters are randomly mutated. Transmission of learned parameters from one generation to the next can lead to fitting of the validation set, and this problem is ameliorated by maintaining high rates of structural mutation.

Models are ranked inductively. Let $I \equiv \{1, 2, \ldots, N\}$ index the models and their validation error vectors $\{\mathbf{e}_i\}$. The n-th rank is assigned

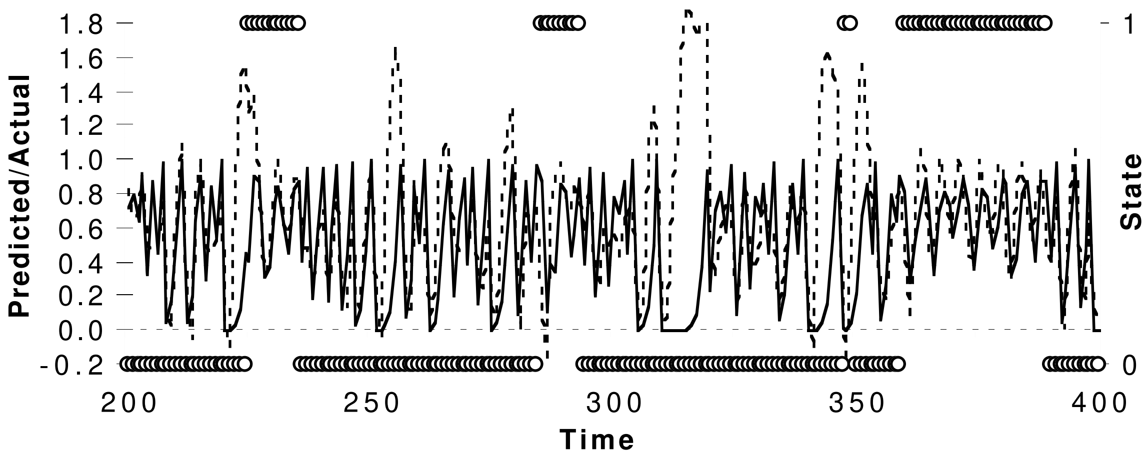

Figure 1. Predictions (*dotted*) of the 2-state logistic map (*solid*) by the top-ranked net of generation 127.

$$r(n) = \underset{i \in I(n)}{\operatorname{argmin}} \sum_{j \in R(n) \cup \{i\}} \frac{<\mathbf{e}_i, \mathbf{e}_j>^2}{<\mathbf{e}_j, \mathbf{e}_j>},$$

where $R(n) \equiv \{r(1), r(2), \ldots, r(n-1)\}$, $I(n) \equiv I - R(n)$, and $n \in I$.

3. Empirical Results

Two chaotic systems have been studied, one synthetic and one natural. In the case of the synthetic system (Section 3.1), the techniques described above were employed. In the case of the sunspots series (Section 3.2), the model combination strategy was somewhat more elaborate (English, in press). In the following, mean squared prediction error is expressed as a fraction of the variance of the entire set of observations.

3.1 Two-State Logistic Map

This system is conceived as one in which high-dimensional chaos triggers transitions from one state of low-dimensional chaos to another. System behavior is predictable over the short term, provided that there is no state change. At each time step the probability of state change is 1/20, however. For current state $s(t) \in \{0, 1\}$, the map is computed

$$x_{t+1} = 4a_{s(t)}x_{t-s(t)}[1 - x_{t-s(t)}] + u_t,$$

where dynamical noise $u_t \sim U(-0.01, 0.01)$. Operations restricting x_{t+1} to the interval [0, 1] have been omitted for the sake of clarity. The state-dependent parameters are $a_0 = 0.9991$ and $a_1 = 0.9009$. The first 200 points generated by the map are used for model fitting, model combination, and validation. Of those points, the first 75 comprise the training set, the following 75 the primary validation set, and the remaining 50 the secondary validation set. The following 200 points (201-400) comprise the test set.

In 44 generations spaced throughout a 195-generation run of the evolutionary algorithm, the highest-ranked parents in the were "harvested" for use in model combination strategies. For each of these generations, a corresponding level-1 predictor combines the predictions of the harvested models. A single level-2 predictor combines the predictions of all 44 level-1 predictors.

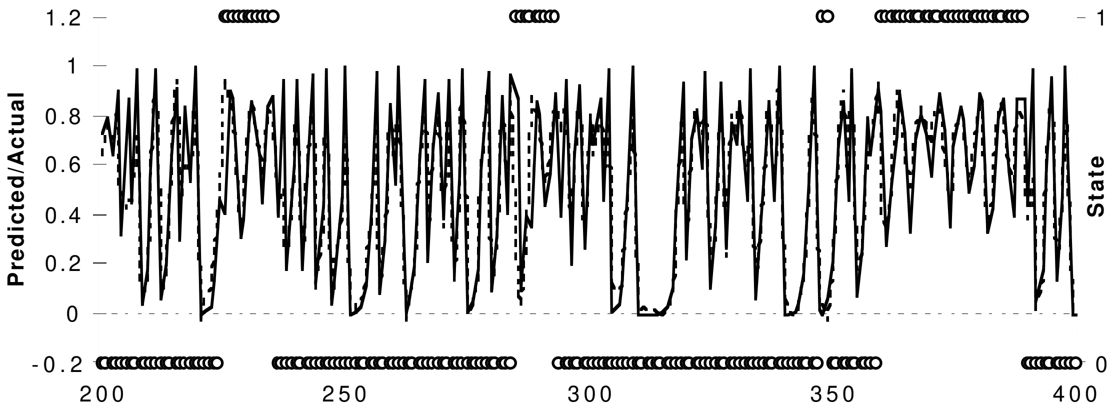

Figure 2. Two-state logistic map (*solid*) and level-2 predictions (*dashed*).

The 44 best-of-generation networks have minimum, median, and maximum testing errors of 0.26, 1.02 and 2.61, respectively. The test-set predictions of the top-ranked net in generation 127 are shown in Fig. 1. Remarkably, the level-1 predictor based on the top 60 nets of generation 127 achieves the least secondary validation error of all level-1 predictors. Its testing error of 0.17 improves substantially upon the best obtained by any best-of-generation neural net. The level-2 combination achieved test-set error of 0.15, which was better than that of any level-1 predictor (see Fig. 2). If eight "outlier" residuals corresponding to unpredictable state transitions are discarded, the error is reduced to 0.05.

3.2 Annual Sunspots Activity

Excellent results have also been obtained in the classic benchmark problem of predicting annual sunspots activity. Table 1 (adapted from English, in press) compares the prediction accuracy obtained with model combination to that obtained with other techniques.

	Train NMSE 1700-1920	**Test NMSE 1921-1955**	**Test NMSE 1956-1979**
Tong & Lim (1980)	0.097	0.097	0.28
Weigend et al. (1990)	0.082	0.086	0.35
Nowlan & Hinton (a) (1992)	—	0.077	—
Nowlan & Hinton (b) (1992)	—	0.072	—
Svarer et al. (1993)	0.090	0.082	0.35
Deco et al. (1994)	0.091	0.087	0.32
McDonnell & Waagen (1995)	0.094	0.083	0.25
English (in press)	0.064	0.064	0.19

Table 1. Normalized mean squared error (NMSE) obtained in various studies of sunspots prediction.

4. Conclusion

Although the techniques described here are certainly not the final answer to issues of model combination, the excellent results are certainly not accidental. When apparently good models often generalize very poorly from one block of a sample to another, it is reasonable to expect that such models will often generalize poorly from the sample to out-of-sample data. If combination of models increases the likelihood of good generalization within-sample, it is again reasonable to expect that such a combination will increase the likelihood of good generalization to out-of-sample data. What sort of modeling principle would recommend committing to a single model when in-sample analysis indicates otherwise? The existence of cases in which in-sample analysis leads to bad modeling decisions does not mean that there is an alternative to assuming that the future will be similar to the past.

References

Breeden, J. L., and N. H. Packard. 1994. A learning algorithm for optimal representation of experimental data. *International Journal of Bifurcation and Chaos* 4: 311-326.

English, T. M. 1995. Stacked generalization and fitness ranking in evolutionary algorithms. In *Evolutionary Computation IV: Proceedings of the Fourth Annual Conference on Evolutionary Programming*, ed. J. R. McDonnell, R. G. Reynolds, and D. B. Fogel, pp. 205-218.

English, T. M. In press. Stacked generalization and simulated evolution. *BioSystems*.

Rissanen, J. 1989. *Stochastic Complexity in Statistical Inquiry* (World Scientific, River Edge, NJ).

Wolpert, D. H. 1992. Stacked generalization. *Neural Networks* 5:241-259.

[Oops. The references in Table 1 need to be added to this list.]