Ensemble of Model Trees and Artificial Neural Networks models for forecasting

Gerald A. Corzo & Li Hong

***Abstract*— This explores the idea that modular models (e.g. model trees) and a global model (ANN-MLP) can compensate transitional, local and global weakness in a time series forecasting model. In the last decade, methods and tools have been increasingly explored to improve the performance of the models, e.g. attribute identification (e.g. PCA, AMI and Correlations), data preprocessing (e.g. data split, normalization), cross validation (10 fold), and ensemble of models (Committee machines). The methodology presented in this paper covers the use of some of the above techniques in a simplified and automatic procedure. The 10 fold cross validation is included with a procedure for the optimization of nodes in the network. The data used for the application of the methodology are 111 time series drawn from homogeneous population of empirical business (NN3 competition – IJCNN 2007). The average training error statistics of the 111 time series models of the ANN and Model Trees are similar. The models ensemble forecast for 18 time steps have visual agreement with the past time series information. The sensitivity of the models to the amount of data is better represented by the model trees process.**

I. Introduction

Data-driven models are used in time series forecasting characterizing the considered system as a whole. Since complex processes are composed of many smaller scale processes it is often inadequate to assume the existence of one single model handling all processes [1], [2]. In the other hand when a time series changes between regimes or states, the overall transition conditions are not well represented by using local models.

The use of Artificial Neural Networks (ANNs) has shown to be a good alternative on forecasting time series. However, there are situations where better performance is required. There are many procedures proposed to setup an optimal neural network model, however, there is still no unique solution. The framework is not straightforward since there are multiple trial and error experiments required to have certain degree of certainty in the model. In this paper, an automatic procedure was tested on the elaboration of 111 time series, where the goal was to forecast 18 time steps ahead.

Gerald Corzo is with the UNESCO-IHE Institute for Water Education P.O. Box 3015, 2601 DA Delft, The Netherlands (corresponding author, phone: +31-15-2151764, e-mail: corzo1@unesco-ihe.org) UNESCO-IHE Institute for Water Education

Li Hong is with the UNESCO-IHE Institute for Water Education and WL/Delft Hydraulics (e-mail: h.li@unesco-ihe.org)

This paper presents a methodology to elaborate and ensemble representative models from global models (ANN) and modular models (Model Trees). The paper covers the description of the automatic procedure to build an ANN model, the introduction of the model tree process, the applications of the methods and their ensembles.

II. Automatic Generation of Artificial Neural Network with optimal nodes

The inputs on forecasting series are typically the measurements at different time in the past (time series), and are used to predict several time steps ahead. In order to obtain the best out of the data set in an automatic procedure, the following steps are implemented in an MATLAB script (www.mathworks.com).

As part of preprocessing and input selection for an ANN model, an autocorrelation analysis is made. Since this correlation can vary highly in multiple experiments, a selection is made based on a threshold (0.5). For those time series with correlations less than 0.5, the first three time lags are selected.

For the selection of the best model in an automatic constrained method, it is necessary to implement a fast and simple model validation. The 10 fold cross validation used in this paper followed the procedure described by Hayken [1]. Since this process involves a number of model training and validations, it is necessary to optimize the fit each fold training sample. This is done by a hard optimization process, testing from one to twenty of the number of nodes. The best performance on the validation sample in each fold is selected as representative of that fold (Fig .1).

Each artificial neural network model used was a multilayer perceptron (MLP). The ANN-MLP was made on the basis of three layer, where the hidden layer had transfer function in each node. The output layer was composed only by a liner combination of the hidden layer. The number of nodes in its structure was optimized using the performance results from the cross validation samples. To train each ANN-MLP model the Levenberg algorithm was used [3]. The parameter for the training where set to be the same (i.e. learning rate = 0.1).

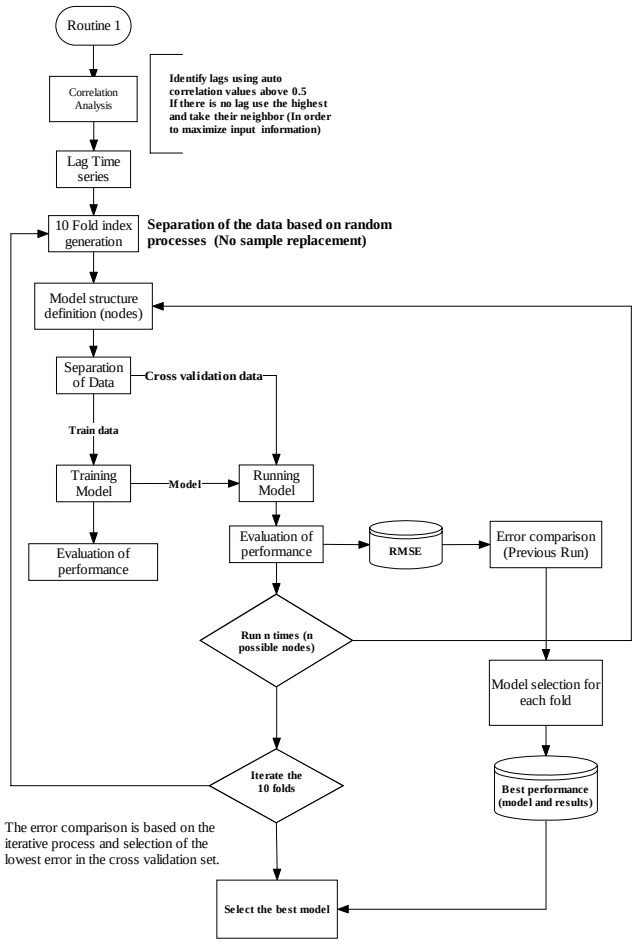

Figure 1 Automatic process for building neural networks

III. MODEL TREES (MODULAR MODELS)

Modularity in the modeling of process has been justified by the principle of divide and conquer. One of the most common problems with the ANN is the seasonality which is accompanied by the over fitting and the misrepresentation of noisy data. Global data-driven models, due to their structure are trained to be accurate on average across the whole span of the time series characterizing the output, they often do not generalize well in some regimes (local over fitting), on the other hand, before a particular pattern is learned, the model potentially could switch into another regime (local unbefitting) [4].

Decision Trees (DT) and Regression Trees (RT) are techniques widely used as classifiers. In the context of forecasting, they have proven to be a good alternative [5-7]. Their important features are sometimes related with the interpretability by human experts. Other recent researches show their applicability in the analysis of extrapolation capabilities.[8].

The algorithm for building regression trees proposed by Breiman [6], specified that the input space is progressively partitioned into subsets by hyperplanes $x_i=A$ (where x_i is one of the model inputs, and i and A are chosen by exhaustive search). A leaf is associated with an average output value of the instances sorted down to the tree (zero-order model). The algorithm used in this experiment followed the pruned version of the M5 tree [9].

IV. ENSEMBLE OF MODELS

The ensemble of models has been used as way to reduce the weakness of different type of models in a data-driven problem. It is assumed that the result of individual models with some weakness can be improved by the combination with other model. This can be seen as combining models with different features which in average will compensate the errors. In the context of computational intelligence, the combinations are widely covered by the concept of committee machine. The committee machine is commonly schematized as shown in Figure 2.3. The input data pass through a split unit (gate) which makes a selection or separation of the data. There is a model built for each selected or separated data stream, which will be combined in a final module. The final module is a unit that combines the values based on the separation or selection done in the split unit. The training process of such a model, as in any computational intelligent method, involves the feedback of the error through different models and then to their parameters.

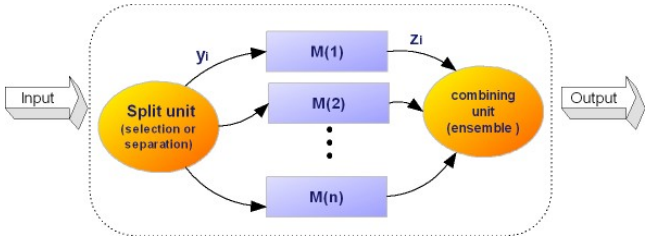

Figure 2 General scheme of a committee machine model

Committee machines applied as ensemble average have been widely used on computational intelligent researches. In this paper, a simple average model is applied to a global model (ANN-MLP) and a modular model (model trees).

V. APPLICATION AND RESULTS

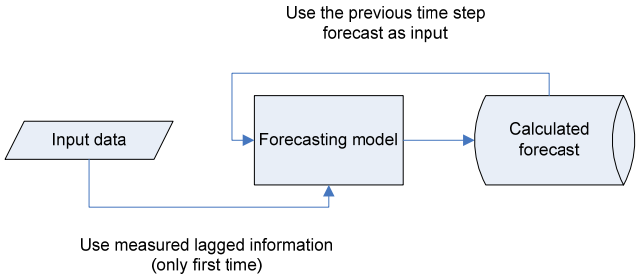

Figure 3 Feedback of model information for iterative forecast.

The models applied were built using 111 time series with a 10 fold cross validation. In the cross validation each model performance is evaluated using the RMSE and PERS[10]. To make an overall evaluation of the fit training results, the average of these measures are presented (Table 1). Since the data sets are part of the NN3 competition (IJCNN 2007), 18 time steps forecast were made. To generate the forecast for each time step, the output from the past step is used as input to the next step (this only if the input lag was less than the 18 steps- Fig 3).

Table1. Average a cross time series results in the training process

	ANN	*MT*	*Ensemble*
NRMSE	139.53	82.26	123.61
RMSE	1603.8	678.93	918.44
PERS	-1.15	0.26	-0.6

The data used as input for the models had two ranges of different number of instances available for the modeling process. In figure 4 we can see that before the time series 50 the number of instances is less than 55. The standard deviation and mean is similar along all the data set, with few exceptions.

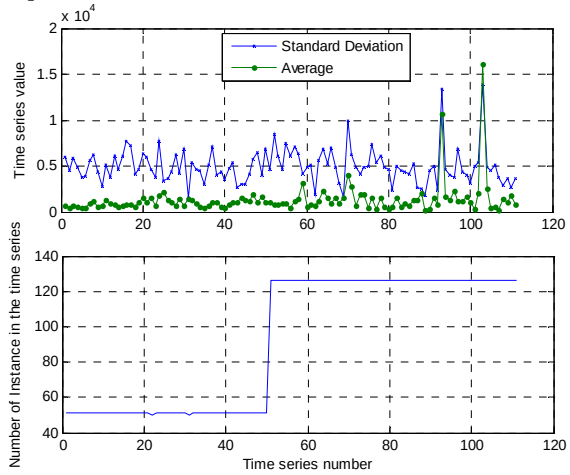

Figure 4 Average and standard deviation values for each time series compared with the number of instances.

The different time series models performance with low number of instance is highly variable (Fig. 4 & 5). This relates to their statistical properties and its level of complexity. The results of the model tree are highly variable but they seem to fit better than the ANN models in the training process. The ensemble model in this training part of the process does not represent a significant improvement (in such cases). This situation seems to be related with the ability of the two methods to adapt with less number of attributes and instances.

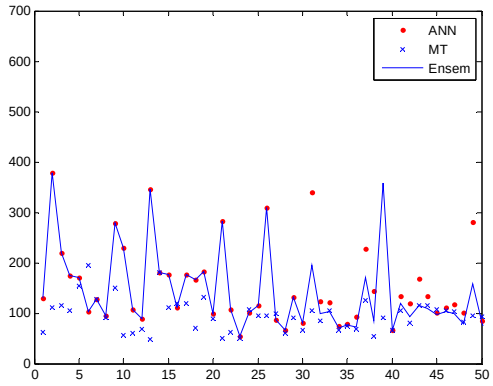

Figure 5 RMSE of the time series with low number of instances.

The figure 6 shows the probability distribution of the different time series in terms of its average and standard deviation. From this visualization it is possible to see that in average 95% of the samples is less than 2000 and the remaining 5% is spread till 16.000 (Fig.6). Comparing this with the standard deviation in the same samples it can be concluded that approximately 50% of the samples have standard deviation with more than three times its average value. This complexity is increased in the first 50 time series due to the low number of instance available.

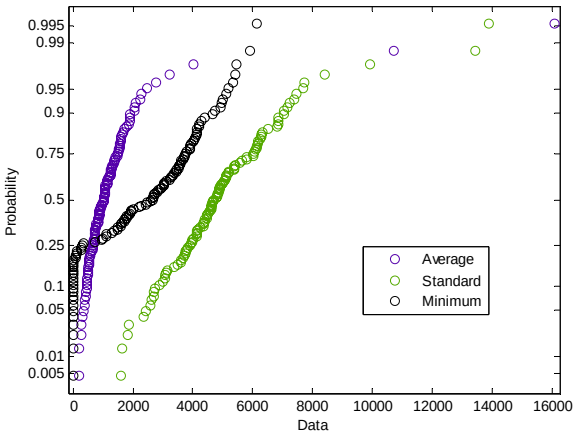

Figure 5 Probability distribution of the mean value, standard deviation and minimum in the time series.

The training of the models has clear visual agreement in some of the cases. Figure 7 shows the performance of one of the ensemble on the time series 77. This visual agreement is found in most of the time series with the higher number of instances available.

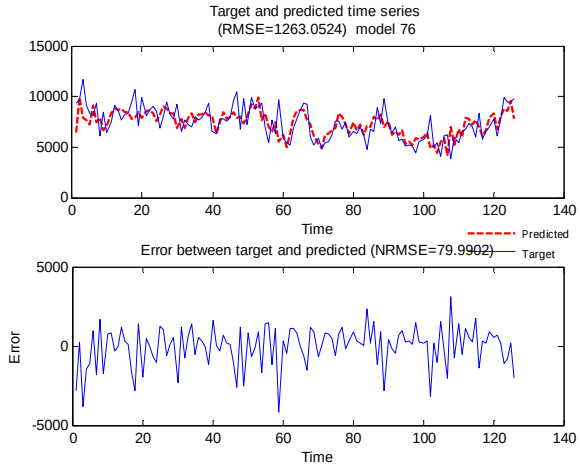

Figure 7 Target and predicted time series for the data set number 77.

The results of the forecasting where no data is available for validation is judged from a visual point of view (Fig 7). The initial part of this graph is generated with the measured information. This measured information is used as input for the steps above where no information is available. This figure shows how the ANN model extrapolates the values and the model tree has a low variability. This situation is averaged and the last 18 steps of the graph show similar frequency and a visual compensation of the two models.

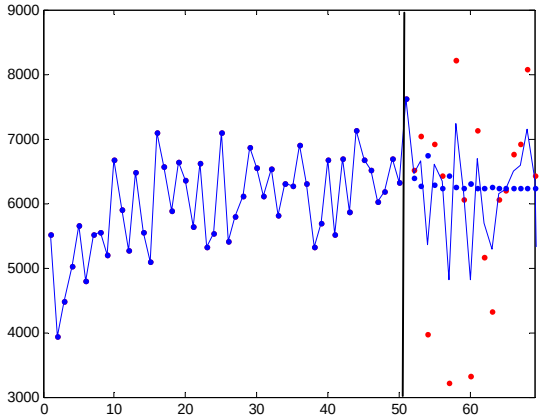

Figure 7 Forecasting 18 time steps ahead (time series data set 1)

VI. Conclusions

This paper makes an analysis of the applicability and some features of the use of ensemble models with global and modular (local) model procedures. The overall model is represented by an automatic methodology to generate a multilayer perceptron neural network and the modular model represented by M5 model trees. The results show that the two modeling techniques train the data in different ways for different conditions of the problem. However, in the forecasting process it appears that there is a compensation of the model results in ensemble model. This would appear to be more significant in situations where few instance are available. However, there are still problem in the automatization of the process since the correlations do not represent a reliable measure.

References

[1] S. Haykin, *Neural networks: a comprehensive foundation*, Second ed: Prentice Hall, 1999.

[2] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*: Wiley InterScience, 2004.

[3] K. Levenberg, "A method for the solution of certain problems in least squares," *Quart. Appl. Math* vol. 2, pp. 164-168, 1944.

[4] A. S. Weigend, M. Mangeas, and A. N. Srivastava, " Nonlinear gated experts for timeseries: Discovering regimes and avoiding overfiting.," *Journal Neural Systems*, vol. 6, pp. 373-399, 1995.

[5] D. P. Solomatine and K. N. Dulal, " Model tree as an alternative to neural network in rainfall-runoff modelling," *Hydrological Science Journal*, vol. 48, pp. 399–411., 2003.

[6] L. Breiman, *Classification and Regression Trees*: Chapman & Hall/CRC, 1984.

[7] L. Breiman, "Bagging Predictors," Department of Statistics, University of California, Berkeley, California 1994.

[8] G. Hooker, "Diagnosing extrapolation: tree-based density estimation," *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 569-574, 2004.

[9] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*: Morgan Kaufmann, 2000.

[10] G. Corzo and D. Solomatine, "Knowledge-based modularization and global optimization of artificial neural network models in hydrological forecasting," *Neural Networks*, vol. 20, pp. 528-536, 2007.