Long Term Time Series Prediction with Multi-Input Multi-Output Local Learning

Gianluca Bontempi
Machine Learning Group, Département d'Informatique
Faculté des Sciences, ULB, Université Libre de Bruxelles
1050 Bruxelles - Belgium
e-mail: gbonte@ulb.ac.be

Abstract. Existing approaches to long term time series forecasting are based either on iterated one-step-ahead predictors or direct predictors. In both cases the modeling techniques which are used to implement these predictors are multi-input single-output techniques. This paper discusses the limits of single-output approaches when the predictor is expected to return a long series of future values and presents a multi-output approach to long term prediction. The motivation for this work is the fact that, when predicting multiple steps ahead of a time series, it could be interesting to exploit the information that a future series value could have on another future value. We propose a multi-output extension of our previous work on Lazy Learning, called LL-MIMO, and we introduce an averaging strategy of several long term predictors to improve the final accuracy. In order to show the effectiveness of the method, we present the results obtained on the three training time series of the ESTSP'08 competition.

1 Introduction

A regular time series is a sequence of measurements φ^t of an observable φ at equal time intervals. Both a deterministic and a stochastic interpretation of the forecasting problem on the basis of historical dataset exist. The deterministic interpretation is supported by the well-known Takens theorem [13] which implies that for a wide class of deterministic systems, there exists a *diffeomorphism* (one-to-one differential mapping) between a finite window of the time series $\{\varphi^{t-1}, \varphi^{t-2}, \dots, \varphi^{t-m}\}$ (*lag vector*) and the state of the dynamic system underlying the series. This means that in theory it exists a multi-input single-output mapping (*delay coordinate embedding*) $f : R^m \rightarrow R$ so that:

$$\varphi^{t+1} = f(\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}) \tag{1}$$

where m (*dimension*) is the number of past values taken into consideration and d is the lag time. This formulation returns a state space description, where in the m dimensional space the time series evolution is a trajectory, and each point represents a temporal pattern of length m.

The representation (1) does not take into account any noise component, since it assumes that a deterministic process f can accurately describe the time series. Note, however, that this is only a possible way of representing the time series phenomenon and that alternative representations should not be discarded a priori. In fact, once we assume that we have not access to an accurate model of the function f, it is reasonable to extend the deterministic formulation (1) to a statistical Nonlinear Auto Regressive (NAR) formulation [8]

$$\varphi^{t+1} = f\left(\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}\right) + w(t) \tag{2}$$

where the missing information is lumped into a noise term w. In the rest of the paper, we will then refer to the formulation (2) as a general representation of the time series which includes as particular instance the case (1).

The success of a reconstruction approach starting from a set of observed data depends on the choice of the hypothesis that approximates f, the choice of the order m and the lag time d. In this paper we will address only the problem of the modeling of f, assuming that the values of m and d are available a priori or selected by conventional model selection techniques. Good references on the order selection are given in [7, 16].

A model of the mapping (2) can be used for two objectives: *one-step* prediction and *iterated* prediction. In the first case, the m previous values of the series are assumed to be available and the problem is equivalent to a problem of function estimation. In the case of iterated prediction, the predicted output is fed back as an input to the following prediction. Hence, the inputs consist of predicted values as opposed to actual observations of the original time series. A prediction iterated for H times returns a *H-step-ahead* forecasting. Examples of iterated approaches are recurrent neural networks [17] or local learning iterated techniques [9, 12].

Another way to perform *H-step-ahead* forecasting is to have a model which returns a direct forecast at time $t+h$, $h = 1, \dots, H$:

$$\varphi^{t+h} = f^h(\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1})$$

Direct methods often require high functional complexity in order to emulate the system. In some cases the direct prediction method yields better results than the iterated one [16]. An example of combination of local techniques of integrated and direct type is provided by Sauer [15].

Iterated and direct techniques for multi-step-ahead prediction share a common feature: they model from historical data a multi-input single-output mapping where the output is the variable φ^{t+1} in the iterated case and the variable φ^{t+h} in the direct case, respectively. This paper advocates that when a very long term prediction is at stake and a stochastic setting is assumed, the modeling of a single-output mapping neglects the existence of stochastic dependencies between future values, (e.g. φ^{t+h} and φ^{t+h+1}) and consequently biases the prediction accuracy. A possible way to remedy to this shortcoming is to move from the modeling of single-output mapping to the modeling of multi-output dependencies. This requires the adoption of a multi-output technique where the predicted value is no more a scalar quantity but a vector of future values of the time series. If there are multiple outputs it is common, apart from some exceptions [11], to treat the prediction problem as a set of independent problems, one per output. Unfortunately this is not effective if the output noises are correlated as it is the case in a time series. The contribution of the paper is to present a simple extension of the Lazy Learning paradigm to the multi-output setting[5, 2]. Lazy Learning (LL) is a local modeling technique which is *query-based* in the sense that the whole learning procedure (i.e. structural and parametric identification) is deferred until a prediction is required. In previous works we presented an original *Lazy Learning* algorithm [5, 2] that selects automatically on a query-by-query basis the optimal number of neighbors. Iterated versions of Lazy Learning were successfully applied to multi-step-ahead time series prediction [4, 6]. This paper presents instead a multi-output version of LL for the prediction of multiple and dependent outputs in the context of long term prediction.

2 Multi-step-ahead and multi-output models

Let us consider a stochastic time-series of dimension m described by the stochastic dependency

$$\varphi^{t+1} = f\left(\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}\right) + w(t) = f(X) + w(t) \tag{3}$$

where w is a zero-mean noise term and X denotes the lag vector

$$X = \{\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}\}$$

Suppose we have measured the series up to time t and that we intend to forecast the next H, $H \geq 1$, values. The problem of predicting the next H values boils down to the estimation of the distribution of the H dimensional random vector

$$Y = \{\varphi^{t+1}, \dots, \varphi^{t+H}\}$$

conditional on the value of X. In other terms, the stochastic dependency (2) between a future value φ^t of the time series and the past observed values X induces the existence of a multivariate conditional probability $p(Y|X)$ where $Y \in \mathbb{R}^H$ and $X \in \mathbb{R}^m$. This distribution can be highly complex in the case of a large dimensionality m of the series and a long term prediction horizon H. An easy way to visualize and reason about this complex conditional distribution is to use a probabilistic graphical model approach. Probabilistic graphical models [10] are graphs in which nodes represent random variables, and the lack of arcs represent conditional independence assumptions. For instance the probabilistic dependencies which characterize a multi-step-ahead prediction problem for a time series of dimension $m = 2$, lag time $d = 0$ and horizon $H = 3$ can be represented by the graphical model in Figure 1. Note that in this figure, $X = \{\varphi^t, \varphi^{t-1}\}$ and $Y = \{\varphi^{t+1}, \varphi^{t+2}, \varphi^{t+3}\}$. This graph shows that φ^{t-1} has a direct influence on φ^{t+1} but only an indirect influence on φ^{t+2}. At the same time φ^{t+1} and φ^{t+3} are not conditionally independent given the vector $X = \{\varphi^t, \varphi^{t-1}\}$.

Any forecasting method which aims to perform multi-step ahead prediction implements (often in an implicit manner) an estimator of the highly multivariate conditional distribution $p(Y|X)$. The graphical model representation can help us in visualizing the differences between the two most common multi-step-ahead approaches, the iterated and the direct one.

The iterated prediction approach replaces the unknown random variables $\{\varphi^{t+1}, \dots, \varphi^{t+H-1}\}$ with their estimations $\{\hat{\varphi}^{t+1}, \dots, \hat{\varphi}^{t+H-1}\}$. In graphical terms this method models an approximation (Figure 2) of the real conditional distribution where the topology of conditional dependencies is preserved though non observable variables are replaced by their noisy estimators. The direct prediction approach transforms the problem of modeling the multivariate distribution $p(Y|X)$ into H distinct and parallel problems where the target conditional distribution is $p(\varphi^{t+h}|X)$, $h = 1, \dots, H$. The topology of the dependencies of the original condition distribution is then altered as shown in Figure 3. Note

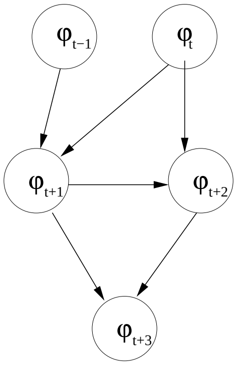

Fig. 1: Graphical modeling representation of the conditional distribution $p(Y|X)$ for $H = 3$, $m = 2$, $d = 0$

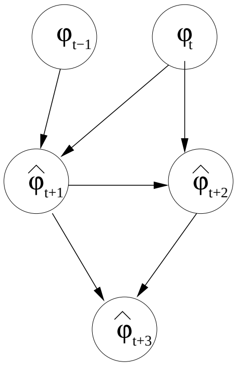
φ_{t-1}
φ_t
$\widehat{\varphi}_{t+1}$
$\widehat{\varphi}_{t+2}$
$\widehat{\varphi}_{t+3}$

Fig. 2: Graphical modeling representation of the distribution modeled by the iterated approach in the $H = 3$, $m = 2$, $d = 0$ prediction problem.

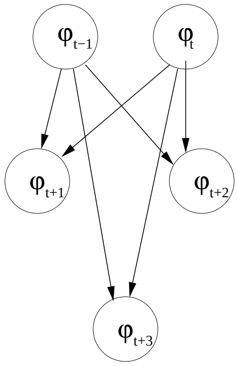

Fig. 3: Graphical modeling representation of the distribution modeled by the direct approach in the $H = 3$, $m = 2$, $d = 0$ prediction problem.

that in graphical model terminology this is equivalent to make a *conditional independence assumption*

$$p(Y|X) = p(\{\varphi^{t+1}, \dots, \varphi^{t+H}\}|X) = \prod_{h=1}^{H} p(\varphi^{t+h}|X)$$

Such assumption is well known in the machine learning literature since it is exploited by the Naive Bayes classifier to simplify multivariate classification problems. Figures 2 and 3 visualize the disadvantages associated to the adoption of the iterated and the direct method, respectively. Iterated methods may suffer of low performance in long horizon tasks. This is due to the fact that they are essentially models tuned with a one-step-ahead criterion and therefore they are not able to take temporal behavior into account. In terms of bias/variance decomposition we can say that the iterated approach returns a non biased estimator of the conditional distribution $p(Y|X)$ since it preserves the dependencies between the components of the vector Y though it suffers of high variance because of the propagation and amplification of the prediction error.

On the other side, direct methods, by making an assumption of conditional independence, neglect complex dependency patterns existing between the variables in Y and consequently return a biased estimator of the multivariate distribution $p(Y|X)$.

In order to overcome these shortcomings, this paper proposes a multi-input multi-output approach where the modeling procedure does not target any more single-output mappings (like $\varphi^{t+1} = f(X) + w$ or $\varphi^{t+k} = f^k(X) + w$) but the multi-output mapping

$$Y = F(X) + W$$

where $F : \mathbb{R}^m \rightarrow \mathbb{R}^H$ and the covariance of the noise vector W is not necessarily diagonal or symmetrical [11]. The multi-output model is expected to return a multivariate estimation of the joint distribution $p(Y|X)$ and, by taking into account the dependencies between the components of Y, to reduce the bias of the direct estimator. However, it is worth noting that, in case of a large forecasting horizons H, the dimensionality of Y is large too, and the multivariate estimation could be vulnerable to large variance. A possible countermeasure to such a side effect is the adoption of combination strategies, which are well reputed to reduce variance in case of low bias estimators. The idea of combining predictors is well known in the time series literature [15]. What is original here is that a multi-output approach allows the availability of a large number of estimators once the prediction horizon H is long. Think for example to the case where $H = 20$ and we want to estimate the value φ^{t+10}. A simple way to make such estimate more robust and accurate is to compute and combine several long term estimators which have an horizon larger than 10 (e.g. all the predictors with horizon between 10 and 20).

For multi-output prediction problems the availability of learning algorithms is much more reduced than in the single output case [11]. Most of existing approaches propose what is actually done by the direct approach, that is to decompose the problem into several multi-output single-output problems by making the assumption of conditional independence. What we propose here is to remove this assumption by using a multivariate estimation of the conditional distribution. For this purpose we adopt a nearest neighbor estimation approach where the problem of adjusting the size of the neighborhood (bandwidth) is solved by a strategy successfully adopted in our previous work on the Lazy Learning algorithm [5, 2].

3 A locally constant method for multi-output regression

We discuss here a locally constant multi-output regression method to implement a multi-step-ahead predictor. The idea is to return, instead of a scalar, a vector which smoothes the continuation of the trajectories which at time t resemble the most to the trajectory X. This method is a multi-output extension of the Lazy Learning algorithm [5, 2] and is referred to as LL-MIMO.

The adoption of a local approach to solve a prediction task requires the definition of a set of model parameters (e.g. the number of neighbors, the kernel function, the parametric family, the distance metric). In local learning literature different methods exist to automatically select the adequate configuration [1, 2] by adopting tools and techniques from the field of linear statistical analysis. One of these tools is the PRESS statistic which is a simple, well-founded and economical way to perform *leave-one-out* (l-o-o) cross-validation and to assess the performance in generalization of local linear models. By assessing the performance of each local model, alternative configurations can be tested and compared in order to select the best one in terms of expected prediction. This is known as the *winner-takes-all* approach in model selection. An alternative to the winner-takes-all approach was proposed in [5, 2] and consists in combining several local models by using the PRESS leave-one-out error to weigh the contribution of each term. This appears to particularly effective in large variance settings [3] as it is presumably the case of a stochastic multi-step-ahead task.

LL-MIMO extends the bandwidth combination strategy to the multi-output case where H denotes both the horizon of the long term prediction and the number of outputs. What we propose is a combination of local approximators with different bandwidths where the weighting criterion depends on the multiple step leave-one-out errors e_h, $h = 1, \dots, H$, computed over the horizon H.

In order to apply local learning to time series forecasting, the time series is embedded into a dataset D_N made of N pairs (X_i, Y_i), where X_i is a temporal pattern of length m, and the vector Y_i is the consecutive temporal pattern of length H.

Suppose the series is measured up to time t and assume for simplicity that the lag $d = 0$. Let us denote

$$\bar{X} = \{\varphi^t, \dots, \varphi^{t-m+1}\}$$

the lag embedding vector at time t. Given a metric on the space $\mathbb{R}^m$ let us order increasingly the set of vectors X_i with respect to the distance to $\bar{X}$ and denote by $[j]$ the index of the jth closest neighbor of $\bar{X}$. For a given number k of neighbors the H step prediction is a vector whose hth component is the average

$$\hat{Y}_h^k = \frac{1}{k} \sum_{j=1}^{k} Y_h^{[j]}$$

where $Y^{[j]}$ is the output vector of the jth closest neighbor of $\bar{X}$ in the training set D_N. We can associate to the estimation $\hat{Y}_h^k$ a multi-step leave-one-error

$$E^k = \frac{1}{H} \sum_{h=1}^{H} e_h^2$$

where e_h is the leave-one-out error of a constant model used to approximate the output at the h step. In case of constant model the l-o-o term is easy to derive [3]

$$e_h = k \frac{Y_h^{[j]} - \hat{Y}_h^k}{k-1}$$

Though the optimal number of neighbors k is not known a priori, in [5, 2] we showed that an effective strategy consists in (i) allowing k to vary in a set $k_1, \dots, k_b$ and (ii) returning a prediction which is the combination of the predictions $\hat{Y}_h^{k_i}$ for each bandwidth k_i, $i = 1, \dots, b$. If we adopt as combination strategy the *generalized ensemble method* proposed in [14], we obtain that the outcome of the LL-MIMO algorithm is a vector of size H whose hth term is

$$\hat{\varphi}^{t+h} = \hat{Y}_h = \frac{\sum_{i=1}^{b} \zeta_i \hat{Y}_h^{k_i}}{\sum_{i=1}^{b} \zeta_i}, \qquad h = 1, \dots, H \tag{4}$$

and the weights are the inverse of the multiple-step l-o-o mean square errors: $\zeta_i = 1/E^{k_i}$.

4 Experiments and final considerations

The LL-MIMO approach has been tested by applying it to the prediction of the three time series from the *ESTSP08 Competition*. The first time series (ESTSP1) has a training set of 354 three-dimensional vectors and the task is to predict the continuation of the third variable for $H = 18$ steps. The second time series (ESTSP2) has a training set of 1300 values and the task is to predict the continuation for $H = 100$ steps. The third time series (ESTSP3) has a training set of 31614 values and the task is to predict the continuation for $H = 200$ steps.

The experimental session aims to compare the following set of methods on a long term prediction task (i) a conventional iterated approach (ii) a direct approach (iii) a multi-output LL-MIMO approach (iv) a combination of several LL-MIMO predictors (denoted by LL-MIMO-COMB) (v) a combination of the LL-MIMO and the iterated approach (denoted by LL-MIMO-IT).

In the strategy LL-MIMO-COMB the prediction at time $t+h$ is

$$\hat{\varphi}^{t+h} = \frac{\sum_{j=h}^{H} \hat{Y}_h^{(H_j)}}{H-h+1},$$

where $\hat{Y}_h^{(H_j)}$ is the prediction of a multi-output LL-MIMO for an horizon $H_j \geq h$. In the strategy LL-MIMO-IT the prediction

$$\hat{\varphi}^{t+h} = \frac{\hat{Y}_h^{(H)} + \hat{Y}_h^{it}}{2}$$

where $\hat{Y}_h^{it}$ is the prediction returned by an iterated scheme. The rationale behind this two averaging methods is the reduction of the variance as discussed at the end of Section 2.

Note that in all the considered techniques the learner is implemented by the same local learning technique which combines a set of constant models whose number of neighbors range in the same interval $[5, k_b]$ with k_b parameter of the algorithm. In order to perform a correct comparison all the techniques are tested under the same conditions in terms of test intervals, embedding order m, values of k_b and lag time d. In detail

- the series ESTSP1 is used to assess the five techniques on the last portion of the training set of size $H = 18$, for values of m ranging from 5 to 20, for values of d ranging from 0 to 1 and and for k_b ranging from 10 to 25,

- the series ESTSP2 is used to assess the five techniques on the last portion of the training set of size $H = 100$, for values of m ranging from 5 to 35, for values of d ranging from 0 to 1 and for k_b ranging from 10 to 25,

- the series ESTSP3 is used to assess the five techniques on the last portion of the training set of size $H = 200$ for $m \in \{20, 50, 80, \ldots, 200\}$, for values of d ranging from 0 to 2 and for k_b ranging from 10 to 15.

Table 1 compares the average NMSE (Normalized[1] Mean Squared Error) prediction errors of the five techniques for the three datasets. The bold notation designs the technique which is significantly better than all the others (with 0.05 significativity level of the permutation test). Table 2 compares the minimum of the NMSE prediction errors attained by the five techniques over all the different configurations in terms of dimension m, lag time d and number k_b.

The experimental results show that for long term prediction tasks the LL-MIMO-COMB and LL-MIMO-IT strategies, i.e. the averaging formulations

[1] The normalization is done with respect to the variance of the entire series

Table 1: Average NMSE of the predictions for the three time series. The bold notation stands for significantly better than all the others at 0.05 significativity level of the paired permutation test.

Test data	LL-IT	LL-DIR	LL-MIMO	LL-MIMO-COMB	LL-MIMO-IT
ESTSP1	1.016	0.239	0.240	**0.219**	0.453
ESTSP2	0.426	0.335	0.335	0.326	**0.189**
ESTSP3	1.63e-2	1.05e-2	1.04e-2	**1.02e-2**	1.12e-2

Table 2: Minimum NMSE of the predictions for time series.

Test data	LL-IT	LL-DIR	LL-MIMO	LL-MIMO-COMB	LL-MIMO-IT
ESTSP1	0.228	0.171	0.172	0.1678	0.190
ESTSP2	0.188	0.130	0.125	0.115	0.104
ESTSP3	1.00e-2	0.96e-2	0.95e-2	0.88e-2	0.93e-2

of the LL-MIMO algorithm, can outperform conventional direct and iterated methods. LL-MIMO alone does not emerge as a competitive algorithm probably because of the excessive variance induced by the large dimensionality. The low biased nature of LL-MIMO however makes of this approach a good candidate for averaging approaches, as demonstrated by the good performance of LL-MIMO-COMB and LL-MIMO-IT. On the basis of these experiences we decided to submit to the Competition the LL-MIMO-IT prediction of the continuation of ESTP2, and the LL-MIMO-COMB prediction of the continuation of ESTP1 and ESTP3. A plot of the LL-MIMO-COMB prediction on the last portion of ESTP3 is illustrated in Figure 4.

We hope that the final validation provided by the Competition continuation series will confirm the importance of multi-output strategies in long term time series forecasting.

References

[1] C. G. Atkeson, A. W. Moore, and S. Schaal. Locally weighted learning. *Artificial Intelligence Review*, 11(1–5):11–73, 1997.

[2] M. Birattari, G. Bontempi, and H. Bersini. Lazy learning meets the recursive least-squares algorithm. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *NIPS 11*, pages 375–381, Cambridge, 1999. MIT Press.

[3] G. Bontempi. *Local Learning Techniques for Modeling, Prediction and Control.* PhD thesis, IRIDIA- Université Libre de Bruxelles, 1999.

[4] G. Bontempi, M. Birattari, and H. Bersini. Lazy learning for iterated time series prediction. In J. A. K. Suykens and J. Vandewalle, editors, *Proceedings of the International Workshop on Advanced Black-Box Techniques for Nonlinear Modeling*, pages 62–68. Katholieke Universiteit Leuven, Belgium, 1998.

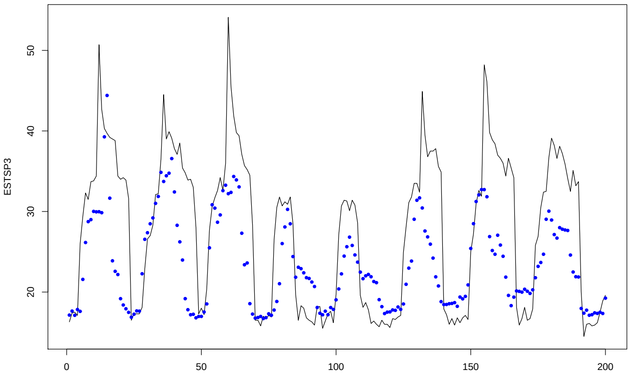

Fig. 4: ESTSP3: time series (line) vs. LL-MIMO-COMB prediction (dots).

[5] G. Bontempi, M. Birattari, and H. Bersini. Lazy learning for modeling and control design. *International Journal of Control*, 72(7/8):643–658, 1999.

[6] G. Bontempi, M. Birattari, and H. Bersini. Local learning for iterated time-series prediction. In I. Bratko and S. Dzeroski, editors, *Machine Learning: Proceedings of the Sixteenth International Conference*, pages 32–38, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

[7] M. Casdagli, S. Eubank, J. D. Farmer, and J. Gibson. State space reconstruction in the presence of noise. *Physica D*, 51:52–98, 1991.

[8] J. Fan and Q. Yao. *Nonlinear Time Series*. Springer, 2005.

[9] J. D. Farmer and J. J. Sidorowich. Predicting chaotic time series. *Physical Review Letters*, 8(59):845–848, 1987.

[10] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer, 2001.

[11] J. M. Matias. Multi-output nonparametric regression. In *Progress in Artificial Intelligence*, pages 288–292, 2005.

[12] J. McNames, J. Suykens, and J. Vandewalle. Winning contribution of the k.u. leuven time-series prediction competition. *International Journal of Bifurcation and Chaos*, 1999. to appear.

[13] N. H. Packard, J. P. Crutchfeld, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Physical Review Letters*, 45(9):712–716, 1980.

[14] M. P. Perrone and L. N. Cooper. When networks disagree: Ensemble methods for hybrid neural networks. In R. J. Mammone, editor, *Artificial Neural Networks for Speech and Vision*, pages 126–142. Chapman and Hall, 1993.

[15] T. Sauer. Time series prediction by using delay coordinate embedding. In A. S. Weigend and N. A. Gershenfeld, editors, *Time Series Prediction: forecasting the future and understanding the past*, pages 175–193. Addison Wesley, Harlow, UK, 1994.

[16] A. Sorjamaa, J. Hao, N. Reyhani, Y. Ji, and A. Lendasse. Methodology for long-term prediction of time series. *Neurocomputing*, 2007.

[17] R. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280, 1989.