Long-Term Prediction of Time Series using a Parsimonious Set of Inputs and LS-SVM

Jarkko Tikka and Jaakko Hollmén

Helsinki University of Technology, Laboratory of Computer and
Information Science, P.O. Box 5400, FI-02015 TKK, Finland
e-mail: tikka@mail.cis.hut.fi, Jaakko Hollmen@tkk.fi

Abstract. Time series prediction is an important problem in many areas of science and engineering. We investigate the use of a parsimonious set of autoregressive variables in the long-term prediction task using the direct prediction approach. We use a fast input selection algorithm on a large set of autoregressive variables for different direct predictors, and train non-linear models (LS-SVM and a committee of LS-SVM) on the parsimonious set of non-contiguous set of autoregressive variables. Results will be shown for the time series competition task.

1 Introduction

Time series prediction aims at predicting future values of a time series based on a recorded, finite collection of time series samples. Time series prediction has many applications in natural sciences, engineering, and economics [2, 5, 11].

In this work, we perform input selection of autoregressive variables to produce a parsimonious, or sparse set of input variables for the prediction problem. The resulting set of variables is typically a non-contiguous set of variables, underlying the importance of a few relevant variables that will produce a good predictor. Inputs of the sparse models are selected from a large set of autoregressive input variables for a given past horizon [8, 9]. In addition, the reduction of input space dimension helps us to avoid the problems caused by the curse of dimensionality [10]. On the basis of the selected variables, we train a non-linear predictor, since the linear models do not produce sufficiently accurate predictions in many applications. Here, we have chosen to use the least squares support vector machine (LS-SVM) [7], since it is relatively fast to train and it has only two tuning parameters in the case of Gaussian kernels. The LS-SVM has been successfully used in the time series prediction context, for instance in [6].

In addition to selecting the best model based on cross-validation, we train several perturbed models around the optimum, and give the average prediction of the set of models as our final prediction. This approach takes into account that the tuning parameters are evaluated in predefined grids and small changes in their values could lead even better prediction performance.

The article is organized as follows. The Section 2 describes our approach to long-term time series prediction. In Section 3, the two phase modeling strategy is described. We briefly present the data provided by the conference organizers and present the empirical experiments in Section 4 followed by summary and conclusions in Section 5.

2 Long-Term Prediction of Time Series

In a time series prediction problem, future values of time series are predicted using the previous values. In the long-term prediction, or multi-step-ahead prediction, recurrent or direct approaches can be used. In the recurrent prediction, the previous predictions are used as inputs, thus errors may accumulate and diminish the quality of predictions. In this work, we rely on direct prediction strategy, where the model

$$\hat{y}_{t+k-1} = f_k(y_{t-1}, y_{t-2}, \ldots, y_{t-l}) \tag{1}$$

is used for k-step-ahead prediction. The predicted values are not used as inputs at all in this approach, thus the errors in the predicted values are not accumulated into the next predictions. However, when all the values from y_t to y_{t+k-1} need to be predicted, k different models must be constructed. This increases the computational complexity, but more accurate results are achieved using the direct than the recursive strategy as shown in [8]. In addition, the direct approach allows us to select different input variables to various k-step-ahead prediction models.

3 Two Phase Prediction Approach

3.1 Phase I: Input Selection

Consider the regression problem that there are N measurements available from an output variable y and input variables $x_i, i = i, \ldots, l$. In this phase, the dependency is assumed to be linear

$$y_j = \sum_{i=1}^{l} \beta_i x_{j,i} + \varepsilon_j, \quad j = 1, \ldots, N \ . \tag{2}$$

The errors ε_j are assumed to be independently normally distributed with zero mean and common finite variance. All the variables are assumed to have zero mean and unit variance, thus the constant term is dropped out from the model. The ordinary least squares (OLS) estimates of the regression coefficients $\hat{\beta}_i$ are obtained by minimizing the mean squared error (MSE).

Usually, the quality of the model (2) can be improved by selecting the input variables. Firstly, the generalization ability of the model may increase if only a subset of input variables are used. Secondly, the final model highlights the most important dependencies better and the underlying model is easier to understand or interpret. In this work, we use a previously proposed efficient input selection procedure [8], [9]. The procedure starts by estimating the linear model using all the available inputs. The sampling distributions of OLS estimates $\hat{\beta}_i$ and the standard deviation s_{tr} of the training MSEs are estimated using M times k-fold cross-validation. Mk estimates of each coefficient β_i formulate the sampling distributions. The median m_{β_i} is used as the location parameter for the distribution, since it is a reasonable estimate for skewed distributions and distributions with outliers. The width of the distribution of $\hat{\beta}_i$ is evaluated using the difference $\Delta_{\beta_i} = \hat{\beta}_i^{high} - \hat{\beta}_i^{low}$, where $\hat{\beta}_i^{high}$ is $Mk(1-q)$th and $\hat{\beta}_i^{low}$ is Mkqth value in the ordered list of the Mk estimates $\hat{\beta}_i$ [3] and q can be set, e.g., $q = 0.165$. With this choice of q, the difference Δ_{β_i} is twice as large as the standard deviation in the case of the normal distribution. The difference Δ_{β_i} describes well the width of both asymmetric and symmetric distributions.

The ratio $|m_{\beta_i}|/\Delta_{\beta_i}$ is used as a measure of significance of the corresponding input variable. The input with the smallest ratio is dropped out from the set of inputs. After that, the cross-validation procedure using the remaining inputs and pruning is repeated as long as there are variables left in the set of inputs.

In the end, we have l different models. The purpose is to select a model which is as sparse as possible, but it still has comparable prediction accuracy. The final model is the least complex model whose validation error is under the threshold $E_v^{min} + s_{tr}^{min}$, where s_{tr}^{min} is the standard deviation of training MSE of the model having the minimum validation error E_v^{min}.

Advantages of the used algorithm is the ranking of the inputs according to their explanatory power and sparseness of the final model. In addition, it is applicable in the case of large number of inputs, since the computational complexity is linear $\mathcal{O}(l)$ with respect to the number of available inputs l.

3.2 Phase II: Non-linear modeling using LS-SVM

Although the linear models are easy to interpret and fast to calculate they are not accurate enough in some problems. Our proposal is to use the selected inputs also in the non-linear model. Goals of this approach are to avoid the curse of dimensionality, over-parameterization, and over-fitting in the non-linear modeling phase. In addition, the interpretability of the non-linear model increases, since only a subset of inputs is included to the model. In this work, we choose the least squares support vector machines (LS-SVM) as a nonlinear predictor [7]. In the primal space, the LS-SVM model is defined as

$$\hat{y} = \boldsymbol{w}^T \phi(\boldsymbol{x}) + b, \tag{3}$$

where $\phi(\cdot)$ is the mapping to the high dimensional feature space. Given a data set $\{\boldsymbol{x}_j, y_j\}_{j=1}^N$, the optimization problem is formulated in the primal space as follows

$$\begin{aligned} \min \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{\gamma}{2}\sum_{j=1}^{N} e_j^2 \\ \text{s.t} \quad & y_j = \boldsymbol{w}^T\phi(\boldsymbol{x}_j) + b + e_j, \quad j = 1, \ldots, N, \end{aligned} \tag{4}$$

where γ is the regularization parameter. In practice, the problem (4) is solved in the dual space and the resulting model is

$$y(\boldsymbol{x}) = \sum_{j=1}^{N} \alpha_j K(\boldsymbol{x}, \boldsymbol{x}_j) + b, \tag{5}$$

where α_j are Lagrange multipliers and the kernel trick $K(\boldsymbol{x}_l, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_l)^T \phi(\boldsymbol{x}_j)$ is applied. The Gaussian kernels $K(\boldsymbol{x}_l, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_l - \boldsymbol{x}_j\|^2/\sigma^2)$ are used in this work. In this case we have two tuning parameters, which are selected using cross-validation. The LS-SVM model is presented in detail, for example, in [7].

From the input selection point of view, the two phase approach is a so called filter approach [4], since a linear model is used to select a parsimonious set of inputs and a non-linear predictor is trained on top of them. In the earlier work [9], we also performed sensitivity analysis on the selected input variables in order to investigate the importance of variables in non-linear model. Since the competition setting does not give too much weight on the interpretation, we omit this aspect from this work.

4 Experiments

In this section, we present our empirical experiments with the competition data set provided by the symposium organizers. The data set is briefly described in the Section 4.1, model selection in Section 4.2 and the results are shown in the section 4.3.

4.1 Prediction competition data set

Time series competitions are a good way to measure out-of-the sample generalization ability of predictors; they have a long history [11] and have become a relatively popular setting for comparing methods for time series predictions. In this competition, the domain of origin of the data remains so far completely unknown to the competitors.

The data set provided by the organizers of the conference is a scalar time series $y_t,\ t = 1, \ldots, N$ with $N = 875$ samples measured in a discrete time grid. The goal of the competition is two-fold: both the 15 and 50 next predicted values of the time series will be used to measure the goodness of the prediction approach. The accuracy of the predictors will be measured by the MSE between the predicted values and the true values, which are not accessible by the competitors. The goodness measure has then the form with $K = 15, 50 : MSE_K = \frac{1}{K}\sum_{k=1}^{K}(y_{t+k-1} - \hat{y}_{t+k-1})^2$. The MSE measures an overall performance averaged over the whole prediction horizon. The short-term prediction being supposedly easier and resulting in smaller MSE values.

4.2 Model selection with cross validation

In order to generalize well, the prediction must be accurate not only in the training set, but also in a data set that is not part of the training. We have left out the last 50 samples from the original data set, which is used as a test set. This simulates the competition setup. In order to estimate generalization properties of the predictors, we have used rest of the data in training and validation. All the models are trained using normalized (zero mean and unit variance) data, but all the results, for instance the errors, are shown in the original scale.

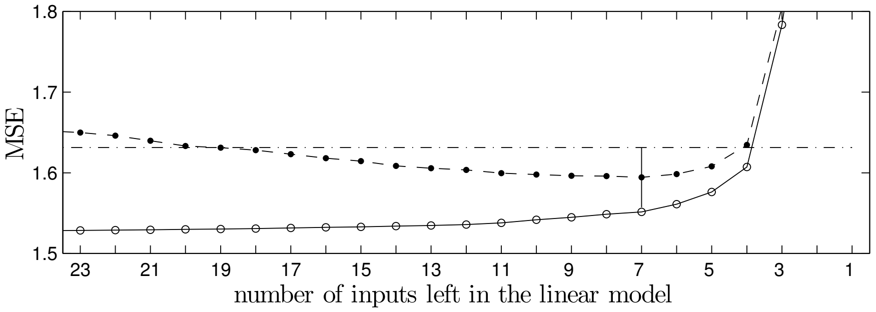

Fig. 1: Illustration of input selection in the prediction of y_{t+24}, training error (*solid line with circles*) and validation error (*dashed line with dots*) as a function of the number of inputs in the linear model. The vertical line marks the minimum validation error and the horizontal dash-dotted line represents the threshold, which is used in the selection of the final model.

In the input selection, 10-fold cross-validation repeated $M = 100$ times is used. This choice produces 1000 estimates for the coefficients β_i, which is considered to be large enough for reliably estimating the distribution of the parameters in the linear model. The LS-SVM models using selected inputs are evaluated using 15 values of the regularization parameter γ and 15 values of the kernel parameter σ^2, which are logarithmically equally spaced in the ranges $\gamma \in [10^{-2}, 10^5]$ and $\sigma^2 \in [10^{-3}, 10^4]$. The optimal values of γ and σ^2 are selected using 10-fold cross-validation repeated ten times to increase the reliability of the results.

For the final predictions used in the competition, we used all the available data to train the models with the parameters producing minimum validation errors. These models cannot obviously be validated against the available data set anymore.

4.3 Results

As a result of our approach to predict the time series, we get a parsimonious set of inputs, which provides insight to the system itself. Since this is of little value in the competition setting, we concentrate on the prediction results.

The maximum number of inputs is set to be $l = 50$, i.e. the available inputs are $y_{t-l}, l = 1, \ldots, 50$ in each of the 50 prediction cases. Figure 1 illustrates how the training and the validation errors develop as a result of our input selection procedure in the case of 25-step-ahead (y_{t+24}) prediction. It is notable that the validation error does not increase drastically until almost all the inputs are dropped out from the model. If the final model had been selected according to the minimum validation error the number of inputs would have been 7. However, even more parsimonious model is achieved when the thresholding is used. The

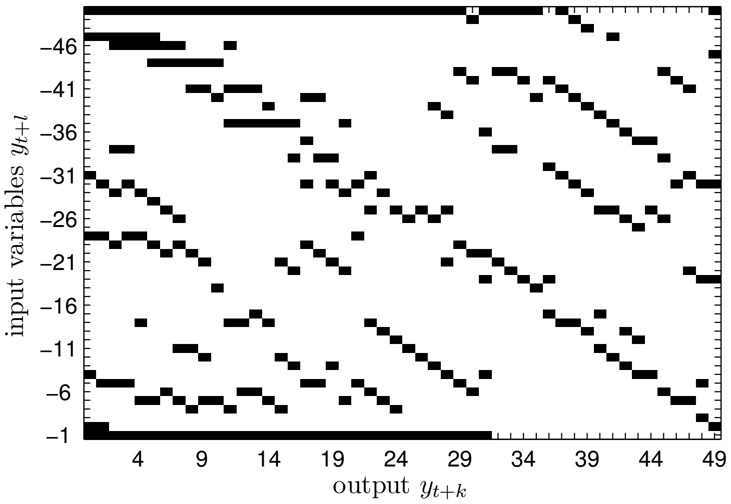

Fig. 2: The selected inputs in the final models. The outputs $y_{t+k}, k = 0, 1, \ldots, 49$ are in the x-axis and the available inputs $y_{t+l}, l = -1, -2, \ldots, -50$ are in the y-axis. The selected inputs are denoted by black rectangles in each column.

final number of inputs is 5 and the validation error does not increase significantly.

In Figure 2, we illustrate the selected inputs in our direct prediction tasks with prediction horizon varying from 1 to 50 $(y_t, \ldots, y_{t+49})$. The black squares in each of the vertical columns denote the inclusion of the variable to the prediction model. The figure shows that each of the direct predictors has a different set of variables. The first available autoregressive input variable y_{t-1} is interestingly included in all the models for predicting y_{t+k-1} upto the value of $k = 32$. Also, the last autoregressive input y_{t-50} is included in most of the models. The partial diagonal patterns indicate the inclusion of a variable with the same absolute lag difference from the value to be predicted. It is noteworthy, that the pattern of selected input variables is relatively sparse, approximately only 5 inputs are chosen from the available 50 autoregressive input variables $(y_{t-1}, \ldots, y_{t-50})$. For instance, the selected 5 inputs in 25-step-ahead prediction model (y_{t+24}) are y_{t-1}, y_{t-4}, y_{t-12}, y_{t-27}, and y_{t-50}. The direct predictor will then have the form $\hat{y}_{t+24} = f_{24}(y_{t-1}, y_{t-4}, y_{t-12}, y_{t-27}, y_{t-50})$ and the form of the prediction function f_{24} is left to be specified.

In Figure 3, the validation mean squared errors for direct k-step-ahead predictions $y_{t+k-1}, k = 1, ...50$ are shown. The dashed line with circles indicates the mean squared errors of the linear predictors using the parsimonious, or sparse pattern of autoregressive input variables. The solid line with dots indicates the mean squared errors of the LS-SVM models trained using the same inputs as the linear models. The non-linear predictors (LS-SVM) show consistently bet-

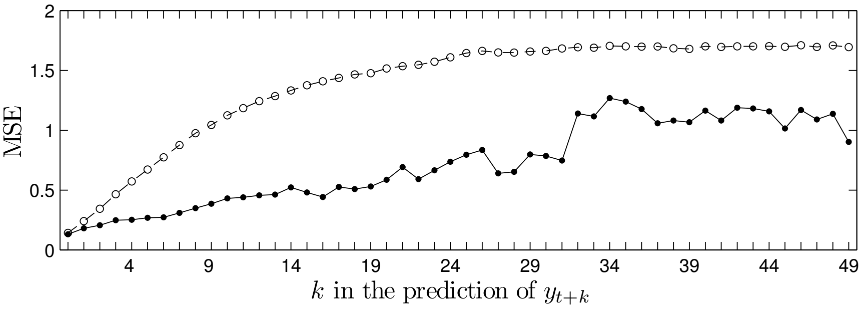

Fig. 3: The mean-square error (MSE) calculated for the validation set for different direct predictors $y_{t+k-1}, k = 1, \ldots, 50$. The MSE for the sparse linear model has been plotted with a dashed line marked with circles, the MSE for the LS-SVM has been plotted with a dotted solid line.

ter performance over whole prediction horizon in the validation set. The errors increase when the prediction horizon increases, which is expected behavior.

In Figure 4, the absolute errors of predictions $|y_{t+k-1} - \hat{y}_{t-k+1}|, k = 1, \ldots, 50$ are shown for the linear models (dashed line with circles) and for the LS-SVM models (solid line with dots) in the test set. The LS-SVM model has smaller absolute error in 19 cases out of the 50, but it only performs clearly better in the prediction of 19-22 steps-ahead. On the other hand, the linear model is clearly more accurate in the prediction of 9-11 and 34-46 steps-ahead.

In order to improve the performance of the LS-SVM models we train a committee of LS-SVM models for each k-step-ahead prediction. It is known that combining the models to form a committee can significantly improve the predictions on new data [1]. For each k-step-ahead prediction case we train a committee, which consists of 10 LS-SVM models. To get variation for the members of committees we train them as follows. Firstly, we train each member using a random subsample of data whose size is 9/10 of the original data. Secondly, the kernel parameters for the members of committee are linearly equally spaced in the range $\sigma^2 \in [\sigma^2_{opt} - \sigma^2_{opt}/2, \sigma^2_{opt} + \sigma^2_{opt}/2,]$, where σ^2_{opt} is the optimal value obtained using cross-validation. The same value of regularization parameter γ is used for each member of the committee. Obviously, the σ^2_{opt} and γ might vary in different k-step-ahead prediction models. In the end, the output of the committee is a mean of the outputs of the members of the committee.

The mean-square test errors $MSE_K = \frac{1}{K}\sum_{k=1}^{K}(y_{t+k-1} - \hat{y}_{t+k-1})^2, K = 15, 50$ for the linear models, the LS-SVM models, and committees of LS-SVMs are presented in Table 1. Although the cross-validation has been done carefully, the test set (the last 50 points not part of the validation or the training set) indicates worse performance for the LS-SVM models than for the linear models. The committees of LS-SVM do not either perform as well as the linear models. This is clearly a contrary finding with the results from the validation results.

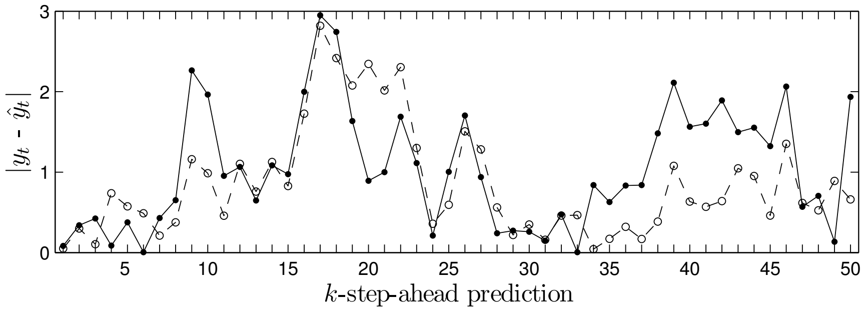

Fig. 4: The absolute errors of predictions $|y_{t+k-1} - \hat{y}_{t-k+1}|, k = 1, \ldots, 50$ in the test set. Dashed line with circles and solid line with dots indicate the errors in the linear and LS-SVM models, respectively.

MSE	sparse linear	LS-SVM	LS-SVM committee
15-step-ahead	0.51	0.98	0.90
50-step-ahead	1.18	1.63	1.60

Table 1: The averaged MSEs for linear models, LS-SVM models, and committees of LS-SVM models for 15- and 50-step-ahead prediction in the test set.

Certainly, Figure 3 indicates the superiority of the LS-SVMs over linear models in the validation set. At this time, we cannot attribute this phenomenon to any particular cause. However, we noted that the averaged MSEs are largely influenced by a small fraction of unsuccessful predictions in the evaluation of test errors in Table 1. Thus, these results can be considered as a precautionary example how well-validated methods can perform poorly in a small test set. In this setup, we have only one sample for each k-step-ahead prediction model in the test set.

The given predictions that will be our contribution to the prediction competition are illustrated in the Figure 5. The upper panel indicates the predictions of the linear predictors with the selected inputs. The lower panel illustrates the predictions using LS-SVM models using selected inputs (black dashed line) and committee of LS-SVMs (gray dashed line). The values on left from the vertical black line are the input values, i.e last 50 values of given time series. No final conclusions can be drawn about quality of the predictions, since the actual values are unknown. However, the predictions using linear model are obviously smoother than predictions using non-linear models. Especially, the time steps from 908 to 920 may be poorly predicted by the non-linear methods. It is also noteworthy, that there are almost no differences between the predictions of LS-SVMs and the committees of LS-SVMs.

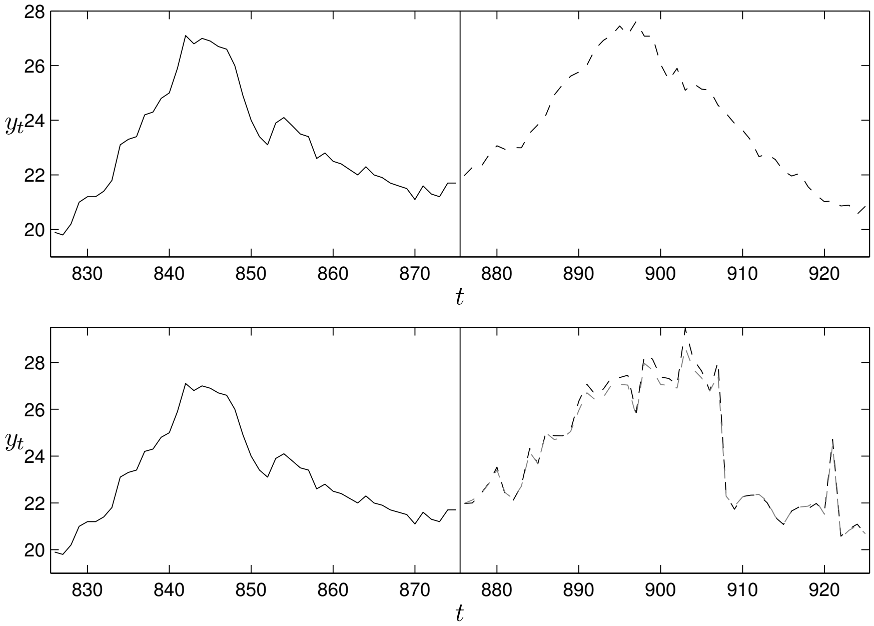

Fig. 5: The predicted values for the purpose of the competition. All 50 predicted values are shown, the predictions are given by the sparse linear model (*upper panel*), LS-SVM (*lower panel, black dashed line*) and a committee of LS-SVM models (*lower panel, gray dashed line*).

5 Summary and Conclusions

Parsimonious sets of input variables provide many advantages in time series prediction. For instance, the final model requires less parameters to be trained. Also the sparsity provides a good basis for interpretation, and highlights the relevant dependencies in time series. In a time series prediction competition setting, we have selected parsimonious inputs in the spirit of backward selection, and trained a LS-SVM prediction model based on the selected inputs. As a final attempt to improve the prediction accuracy, we have trained a set of models on slightly perturbed parameters around the optimal cross-validated model parameters and averaged the results. The goal of our approach is to produce accurate results, and at the same time provide comprehensible views into the system through the set of parsimonious regressors.

Acknowledgments

This work has been supported by the Academy of Finland, the grant number 116853.

References

[1] Christopher Bishop. *Neural Networks in Pattern Recognition.* Oxford Press, 1996.

[2] Chris Chatfield. *Time Series Forecasting.* Chapman & Hall/CRC, 2002.

[3] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap.* Chapman & Hall/CRC, 1993.

[4] Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition.* Computer Science and Scientific Computing. Academic Press, second edition edition, 1990.

[5] James D. Hamilton. *Time Series Analysis.* Princeton University Press, 1994.

[6] A. Lendasse, V.Wertz, G.Simon, and M. Verleysen. Fast bootstrap applied to LS-SVM for long term prediction of time series. In *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks*, pages 705–710, 2004.

[7] Johan A.K. Suykens, Tony Van Gestel, Jos De Brabanter, Bart De Moor, and Joos Vandewalle. *Least Squares Support Vector Machines.* World Scientifc Publishing, 2003.

[8] Jarkko Tikka, Jaakko Hollmén, and Amaury Lendasse. Input Selection for Long-Term Prediction of Time Series. In Joan Cabestany, Alberto Prieto, and Francisco Sandoval, editors, *Proceedings of the 8th International Work-Conference on Artificial Neural Networks (IWANN 2005)*, volume 3512 of *Lecture Notes in Computer Science*, pages 1002–1009. Springer-Verlag, 2005. Vilanova i la Geltú, Barcelona, Spain.

[9] Jarkko Tikka, Amaury Lendasse, and Jaakko Hollmén. Analysis of fast input selection: Application in time series prediction. In Stefanos Kollias, Andreas Stafylopatis, Wlodzislaw Duch, and Erkki Oja, editors, *Proceedings of the 16th International Conference on Artificial Neural Networks (ICANN'06)*, volume 4132 of *Lecture Notes in Computer Science*, pages 161–170. Springer-Verlag, 2006.

[10] Michel Verleysen and Damien François. The curse of dimensionality in data mining and time series prediction. In Joan Cabestany, Alberto Prieto, and Francisco Sandoval, editors, *Proceedings of the 8th International Work-Conference on Artificial Neural Networks (IWANN 2005)*, volume 3512, pages 758–770. Springer-Verlag, 2005.

[11] A. S. Weigend and N. A. Gershenfeld, editors. *Time Series Prediction: Forecasting the Future and Understanding the Past.* Addison-Wesley, 1994.