Long-Term Prediction of Time Series using NNE-based Projection and OP-ELM

Antti Sorjamaa, Yoan Miche, Robert Weiss and Amaury Lendasse

***Abstract*— This paper proposes a combination of methodologies based on a recent development –called Extreme Learning Machine (ELM)– decreasing drastically the training time of nonlinear models. Variable selection is beforehand performed on the original dataset, using the Partial Least Squares (PLS) and a projection based on Nonparametric Noise Estimation (NNE), to ensure proper results by the ELM method. Then, after the network is first created using the original ELM, the selection of the most relevant nodes is performed by using a Least Angle Regression (LARS) ranking of the nodes and a Leave-One-Out estimation of the performances, leading to an Optimally-Pruned ELM (OP-ELM). Finally, the prediction accuracy of the global methodology is demonstrated using the ESTSP 2008 Competition and Poland Electricity Load datasets.**

I. Introduction

Time series forecasting is a challenge in many fields. In finance, experts forecast stock exchange courses or stock market indices; data processing specialists forecast the flow of information on their networks; producers of electricity forecast the load of the following day. The common point to their problems is the following: how can one analyse and use the past to predict the future?

Many techniques exist for the approximation of the underlying process of a time series: linear methods such as ARX, ARMA, etc. [1], and nonlinear ones such as artificial neural networks [2]. In general, these methods try to build a model of the process. The model is then used on the last values of the series to predict the future values. The common difficulty to all the methods is the determination of sufficient and necessary information for an accurate prediction.

A new challenge in the field of time series prediction is the Long-Term Prediction: several steps ahead have to be predicted. Long-Term Prediction has to face growing uncertainties arising from various sources, for instance, accumulation of errors and the lack of information [2].

In this paper, we use a Direct prediction strategy to ensure the accuracy of the prediction, even in long-term. The Direct strategy does not suffer from the accumulation of errors as does the Recursive one. On the other hand, the Direct strategy needs reliable variable selection and model training for every prediction step.

In this paper, two projection methods, Partial Least Squares (PLS) and Nonparametric Noise Estimation (NNE) are used to create a set of inputs, which contain the relevant information for the prediction purpose. The combination of the methods project the high-dimensional input regressor into low-dimensional latent space maximizing the prediction ability of any nonlinear approximator.

The approximator used in this paper is a Feed-Forward Neural Network. The reason why these type of networks are not widely used in industry for data mining purposes, is that they are very slow to train. This is due to the many parameters to be properly tuned by slow (often gradient-based) algorithms, in order to obtain a good enough model. Furthermore, the training phase has to be repeated in order to perform model structure selection, for example the selection of the number of hidden neurons or the selection of some regularization parameter.

In [3], Guang-Bin Huang *et al.* propose an original algorithm for hidden nodes determination and weights selection called Extreme Learning Machine (ELM). The main advantage of this algorithm is in dividing the computational time by hundreds and making the learning process of the neural network rather simplistic. In this paper, a methodology based on ELM, called OP-ELM (for Optimally-Pruned ELM) with two main goals is proposed:

- being able to construct/select a nonlinear model in computational times close to these of linear models,
- this while keeping roughly the same performances as with the possibly best current algorithms.

For this purpose, we go through four main techniques, integrated in the OP-ELM methodology as four necessary steps, namely: variable selection [4], [5], [6], the mentioned Extreme Learning Machine [3], Least Angle Regression model selection [7] and finally a fast and exact estimation of the Leave-One-Out validation error in the training process, using PRESS statistics [8], [9].

In the next section, the two prediction strategies, Direct and Recursive, are explained. In Section III the global methodology is summarized. The projection methods, PLS and NNE, are more deeply explained in Section IV and the nonlinear approximator OP-ELM in Section V. Finally, Section VI provides experimental results and conclusions based on ESTSP 2007 Competition dataset and Poland Electricity Load dataset.

Antti Sorjamaa, Yoan Miche and Amaury Lendasse are with the Adaptive Informatics Research Centre in Helsinki University of Technology, Espoo, Finland. Yoan Miche is also with the Gipsa-lab in CNRS, INPG, Grenoble, France, (email: {Antti.Sorjamaa, ymiche, Lendasse}@hut.fi). Robert Weiss is with Process Vision OY, Helsinki, Finland, (email: robert.weiss@processvision.fi).

This work was supported by the project of New Information Processing Principles, 44886, of the Academy of Finland, and by the NoTeS project of Tekes, Finland, and by the Department of Computer and Information Science, Helsinki University of Technology, Finland.

II. Time Series Prediction Strategies

The time series prediction problem is the prediction of future values based on the previous values and the current value of the time series (see Equation 1). The previous values and the current value of the time series are used as inputs for the prediction model. One-step ahead prediction is needed in general and is referred to as Short-Term Prediction. But when multi-step ahead predictions are needed, it is called a Long-Term Prediction problem.

Unlike the Short-Term time series prediction, the Long-Term Prediction is typically faced with growing uncertainties arising from various sources. For instance, the accumulation of errors and the lack of information make the prediction more difficult. In Long-Term Prediction, performing multiple step ahead prediction, there are several alternatives to build models. In the following sections, two variants of prediction strategies are introduced and compared: the Direct and the Recursive Prediction Strategies.

A. Recursive Prediction Strategy

To predict several steps ahead values of a time series, Recursive Strategy seems to be the most intuitive and simple method. It uses the predicted values as known data to predict the next ones. In more detail, the model can be constructed by first making one-step ahead prediction:

$$\hat{y}_{t+1} = f_1(y_t, y_{t-1}, ..., y_{t-M+1}), \tag{1}$$

where M denotes the number input variables. The regressor of the model is defined as the vector of inputs: $y_t, y_{t-1}, ..., y_{t-M+1}$. It is possible to use also exogenous variables as inputs in the regressor, but they are not considered here in order to simplify the notation. Nevertheless, the presented global methodology can also be used with exogenous variables.

To predict the next value, the same model is used:

$$\hat{y}_{t+2} = f_1(\hat{y}_{t+1}, y_t, y_{t-1}, ..., y_{t-M+2}). \tag{2}$$

In Equation 2, the predicted value of $\hat{y}_{t+1}$ is used instead of the true value, which is unknown. Then, for the H-steps ahead prediction, $\hat{y}_{t+2}$ to $\hat{y}_{t+H}$ are predicted iteratively. So, when the regressor length M is larger than H, there are $M - H$ real data in the regressor to predict the H^{th} step. But when H exceeds M, all the inputs are the predicted values. The use of the predicted values as input variables deteriorate the accuracy of the prediction.

B. Direct Prediction Strategy

Another strategy for the Long-Term Prediction is the Direct Strategy. For the H-steps ahead prediction, the model is

$$\hat{y}_{t+h} = f_h(y_t, y_{t-1}, ..., y_{t-M+1}) \text{ with } 1 \leq h \leq H. \tag{3}$$

This strategy estimates H direct models between the regressor (which does not contain any predicted values) and the H outputs. The errors in the predicted values are not accumulated in the next prediction. When all the values, from $\hat{y}_{t+1}$ to $\hat{y}_{t+H}$, need to be predicted, H different models must be built. The direct strategy increases the complexity of the prediction, but is shown to be more accurate in the long-term [10].

III. Global Methodology

Figure 1 summarizes the methodology used in this paper.

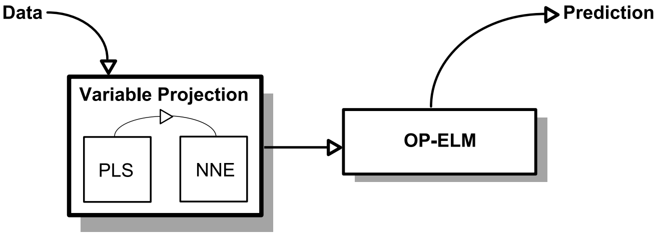

Fig. 1. Global methodology summarized.

The input variable projection consists of a combination of two projection methods: Partial Least Squares (PLS) and a projection method based on Nonparametric Noise Estimation (NNE). The projection methods are explained in the Sections IV-A and IV-C.

The nonlinear approximator used is an Optimally-Pruned Extreme Learning Machine (OP-ELM). This approximator is a Feed-Forward Neural Network with a random selection of weights as its training. The network is combined with Least-Angle Regression (LARS) and Leave-one-out (LOO), which rank the outputs of the randomly initialized neurons and validate the optimal selection of them, respectively.

IV. Projection Methods

This section briefly overviews the projection methods and related tools used in this paper. The methods are the Partial Least Squares (PLS), the Delta Test (DT) for Nonlinear Noise Estimation (NNE) and the the Extended Forward-Backward variable selection method (EFB). The projection methodology is summarized in Figure 2 and in Table I.

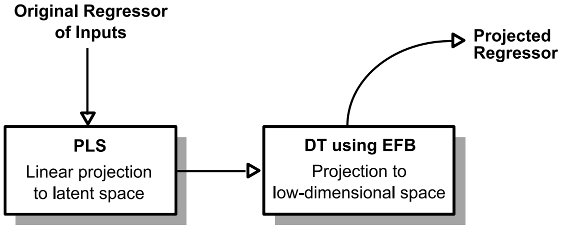

Fig. 2. Projection methodology summarized.

A. Partial Least Squares

The Forward-Backward selection has been previously tested in order to optimize a projection matrix when the criterion is the minimization of the Delta Test. Unfortunately, the method is converging and the results are satisfactory only if the number of variables is small.

TABLE I

PROJECTION METHODOLOGY SUMMARIZED.

1) Build a PLS-R model between the inputs and the output. For a model cross-validated for k_1 latent directions retain as many as $2k_1$

2) Project the data $\mathbf{x}$ onto the space spanned by the first $2k_1$ PLS-R directions

$$\mathbf{z}_1 = \mathbf{x}\mathbf{P}_1. \tag{4}$$

Here, $\mathbf{P}_1$ denotes the the projection matrix associated to PLS-R

3) Perform EFB in order to find a second projection matrix $\mathbf{P}_2$ such that the DT between the final set of latent inputs and the output is minimized

$$\mathbf{P}_2 = \min_{\mathbf{P}} \frac{1}{2N} \sum_{i=1}^{N} ||y_{NN(\mathbf{z}_{1,i})} - y_i||^2, \tag{5}$$

4) Project $\mathbf{z}_1$ onto the space spanned by the directions optimized with DT

$$\mathbf{z}_2 = \mathbf{z}_1\mathbf{P}_2. \tag{6}$$

In order to approach such a restriction, we suggest to use the PLS as a preprocessing step; thus, allowing a preliminary reduction in the dimensionality of the original problem. The number of latent variables to be retained after performing the PLS should be a compromise capable of conserving most of the information exploitable by a nonlinear method, but also small enough in order to be able to perform the minimization of the Delta Test. Notice that the number of variables retained from the PLS is, however, not critical when the choice is conservative; in our experiments, we found that retaining roughly twice the number of latent variables obtained from a cross-validated PLS is typically appropriate.

The linearly projected data $\mathbf{z}_2 = \mathbf{x}\mathbf{P}_1\mathbf{P}_2 = \mathbf{x}\mathbf{P}$ are then used to calibrate any nonlinear model to estimating the output y.

B. Nonparametric Noise Estimation with the Delta Test

Delta Test (DT) is a technique for estimating the variance of the noise or, equivalently, the Mean Square Error (MSE), that can be achieved by a regression model without overfitting; see [11] and references therein. As such, the DT is useful for evaluating the nonlinear correlation between two random variables and can be included in variable selection schemes: the set of inputs minimizing the DT is the one that is to be retained.

Given N input-output pairs: $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, the relationship between $\mathbf{x}_i$ and y_i is modeled as $y_i = f(\mathbf{x}_i) + r_i$ where f is the unknown function to be estimated and r_i is the noise. The Delta Test is a data-derived method for estimating the variance of the noise in such a setting. Denoting by $\mathbf{x}_{NN(\mathbf{x}_i)}$ the first nearest neighbor of point $\mathbf{x}_i$ in the set $\{\mathbf{x}_i\}_{i=1}^{N}$ and $y_{NN(\mathbf{x}_i)}$ the associated output, the Delta Test, δ, formulates as:

$$\delta = \frac{1}{2N} \sum_{i=1}^{N} ||y_{NN(\mathbf{x}_i)} - y_i||^2. \tag{7}$$

C. Projection based on the Delta Test

Linear projection is a common preprocessing step in both function approximation and classification tasks. When regression is to be performed, the aforementioned PLS, as well as other methods like Principal Components Regression (PCR), are standard approaches based on the idea of combining the original variables by projection. The methods project the original input variables onto a latent space with reduced dimensionality; in PCR, the projection is constructed in order to keep a maximum of information from the input variables, whereas PLS builds new inputs that are also suitable to approximate the output, [12].

This subsection illustrate an efficient strategy to use the Delta Test as a tool to select an optimal linear projection of the input variables. Being based on the DT, the strategy is mostly suitable when a nonlinear model is used to reconstruct the relationship between the new latent inputs and the output.

For N input-output pairs, $(\mathbf{x}, y) \in \mathbb{R}^d \times \mathbb{R}$, a new set of inputs $\mathbf{z}$ is given as:

$$\mathbf{z} = \mathbf{x}\mathbf{P}, \tag{8}$$

where $\mathbf{P}$ is the projection matrix. According to Delta Test, the best set of latent variables $\mathbf{z}$ is found as the one that minimizes:

$$\delta = \frac{1}{2N} \sum_{i=1}^{N} ||y_{NN(\mathbf{z}_i)} - y_i||^2, \tag{9}$$

where $y_{NN(\mathbf{z}_i)}$ is now the output for $\mathbf{z}_{NN(\mathbf{z}_i)}$. Thus, we define an optimal $\mathbf{P}$ as:

$$\mathbf{P}^{opt} = \min_{\mathbf{P}} \frac{1}{2N} \sum_{i=1}^{N} ||y_{NN(\mathbf{z}_i)} - y_i||^2, \tag{10}$$

Unfortunately, the optimization for $\mathbf{P}^{opt}$ is difficult because the Delta Test is not differentiable with respect to $\mathbf{P}$; the discontinuity is due to the fact that the Delta Test estimates the variance of the noise based on nearest neighbors.

In order to optimize for $\mathbf{P}^{opt}$, an Extended Forward-Backward optimization technique can be used.

The Forward-Backward Selection (FB) is a commonly used strategy for variable selection. The method is fast but there is no guarantee that the optimal set of variables is found [13]. In FB, each variable can be in two states: ”1”, meaning that it belongs to the set of selected variables or ”0” meaning that it does not and it is temporarily discarded. Given a certain initial state for all variables, the procedure flips the state of each variable at a time and computes a predefined criterion (for example, the Delta Test). The flipping operation that improves performances the most is accepted, and the states are flipped again (excluding the previously accepted change). The process is continued until no improvement is found.

The FB can be extended to any optimization problem for which the importance (or level) of a variable is searched; that is, instead of switching scalars from 0 to 1 or vice versa, by increasing (in case of forward selection) or decreasing (for backward selection) by regular steps $1/h$ from 0 to 1. In this study, we suggest an application of Extended FB (EFB) schemes to the problem of optimizing the projection matrix $\mathbf{P}$.

In general, we assume that the initial variables have been normalized and that the values of $\mathbf{P}$ can be bounded by -1 and 1. In practice, a degree of discretization $h = 10$ is found to be accurate enough and that leaves us 21 possible values for each variable. For a projection onto a 2-dimensional space, the procedure can be summarized as:

1) initialize the first column of $\mathbf{P}$
2) optimize the first column of $\mathbf{P}$ by EFB and DT in the projected space
3) initialize the second column of $\mathbf{P}$
4) optimize the second column of $\mathbf{P}$ by FBS with the first column unchanged

The data projected onto a bi-dimensional latent space are easily displayed and initially used to investigate their structure in the input space, being the visualization supervised by the output. If the visualization is not the main concern, the procedure can be extended to additional columns of $\mathbf{P}$ (e.g., until no significant decrease of the Delta Test is observed) and then used to estimate the output.

V. OP-ELM

Figure 3 sums up the four main steps of OP-ELM [14].

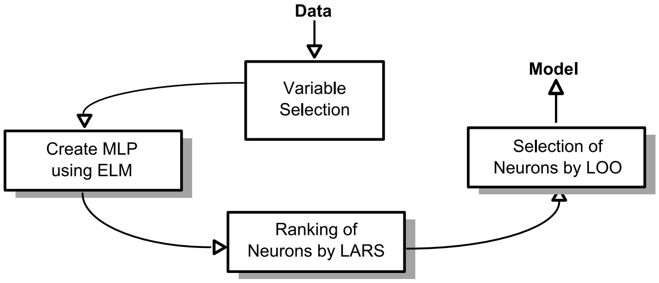

Fig. 3. The four steps of the proposed OP-ELM

A. Variable selection

An *a priori* variable selection has to be performed on the data set in order to remove the possibly irrelevant variables (not necessarily the redundant ones), for the problem.

Generally, the variable selection can be achieved by any well-known technique. Since computational speed is the main advantage of OP-ELM, fast methods, such as Forward Selection [4], is one of the most recommended ones; more elaborated techniques for selection using Markov blanket [5], typical mutual information [4] or a combination of mutual information with Forward selection and other sampling methods as proposed in [6] can also be used, at the possible penalty of a longer computational time for this step.

In this paper, the variable selection is actually performed by variable projection, which creates new input variables from the original inputs. The projection methods described in Section IV are fast and fit well with the nonlinear approximation technique of the OP-ELM.

B. Extreme Learning Machine (ELM)

Once the dataset has been pruned of its irrelevant variables, the actual feed-forward neural network is built, with only one hidden layer as proposed in the ELM algorithm. This algorithm has been presented by Guang-Bin Huang *et al.* in [3], although a common idea existed already in [15]. In ELM, traditional multilayer perceptrons with one hidden layer is used. The weight between the input data and the hidden-layer are denoted $\mathbf{w}_i$. The weights between the hidden-layer and the output are denoted $\mathbf{b}$. The activation functions used are sigmoids in the hidden-layer and a linear function for the output layer. The novelty is in the determination of the input weights $\mathbf{w}_i$, which are randomly determined from a uniformly distributed distribution (for example between -10 and 10). Indeed, with this done and following the mandatory hypothesis that the activation functions f of the hidden layer are indefinitely differentiable in any interval of their domain, the output weights $\mathbf{b}$ can be simply calculated from the hidden layer output matrix $\mathbf{H}$. Each column of $\mathbf{H}$ is given by the product of the weight vector and the input vectors: $\mathbf{h_i} = \text{sigmoid}(\mathbf{x_i}^T\mathbf{w_i})$. The output weights are calculated by $\mathbf{b} = \mathbf{H}^\dagger\mathbf{y}$, where $\mathbf{H}^\dagger$ stands for the Moore-Penrose inverse [16] and $\mathbf{y} = (y_1, \ldots, y_M)^T$ is the output. The choice of the number of neurons N to be used in the hidden layer remains the only arbitrary parameter; since the next step of the methodology is meant to prune the unuseful neurons of the hidden layer, it is wise to have sufficient number of neurons for the ELM part.

C. Least Angle Regression (LARS)

The LARS algorithm was proposed by Efron *et al.* in [7] and implemented in [17]. The basic underlying idea of this selection algorithm, is following the Forward selection one:

1) Select predictor x_{j_1} giving best results alone
2) The second, x_{j_2}, is selected by looking for the best x_{j_2} along x_{j_1}
3) Third, x_{j_3}, searched for in equiangular direction between x_{j_1} and x_{j_2}
4) All remaining are searched for in the equiangular direction between all selected predictors
5) In the end, a ranking of the predictors is obtained.

In the case of our neural network built in the previous stage of the methodology, we are going to rank the hidden layer neurons by the LARS algorithm. Since the part between the hidden and the output layer of the neural network is linear, LARS is guaranteed to find the best possible ranking of predictors.

Finally, the selection of the final model structure is achieved through Leave-One-Out validation in the last step of the methodology.

D. Leave-One-Out (LOO)

For the estimation of the validation error and the actual selection of the best neurons for the problem, a Leave-One-Out is used.

Calculating the LOO error ϵ can be very time consuming when datasets tend to have a large number of samples. Fortunately, the PRESS (or PREdiction Sum of Squares) statistics provide a direct and exact formula for the calculation of the LOO error for linear models (see [8], [9] for details on this formula and implementations):

$$\epsilon^{\text{PRESS}} = \frac{y_i - \mathbf{x}_i \mathbf{b}}{1 - \mathbf{x}_i \mathbf{P} \mathbf{x}_i'}, \tag{11}$$

where $\mathbf{P}$ is defined as $\mathbf{P} = (\mathbf{X}'\mathbf{X})^{-1}$ and $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$.

Finally, evaluating the LOO error versus the number of neurons used (which have been previously properly ranked by the LARS algorithm) enables to select the best model for the problem, that is, with a low enough number of neurons to avoid over-fitting, but still large enough to ensure proper generalization.

E. Toy Example

This toy example shows the application of the OP-ELM method to a sum of two sines. A set of 1000 training points are generated, which gives a one-dimensional example where no feature selection has to be performed beforehand. Figure 4 plots the obtained model on top of the training data.

For the test, a set of 10 000 samples is used to check the validity of the selected model.

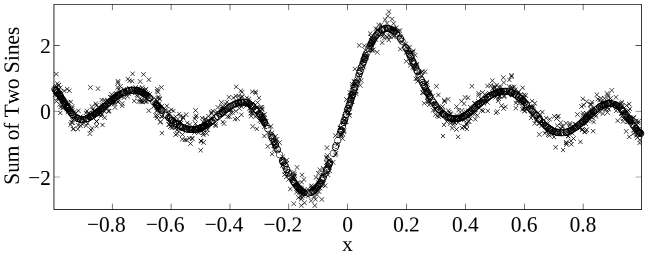

Fig. 4. Plot of a one-dimensional sum of two sines. The data is shown as crosses and the model obtained by OP-ELM using circles.

The model seems to approximate the data very nicely, and using a number of neurons around 20, one reaches an error already equal to the noise introduced in the dataset (0.0625) as can be seen on Figure 5.

VI. Experiments

In this section the global methodology is applied to ESTSP 2007 Competition dataset and to the Poland Electricity Load dataset.

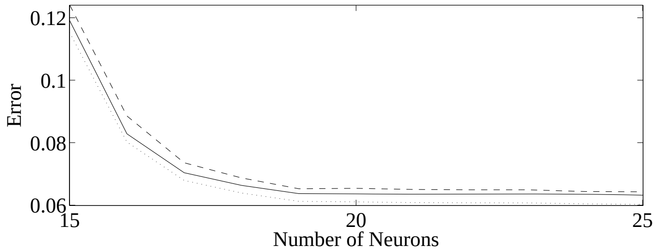

Fig. 5. Mean Square Error shown as dotted line, the Leave-One-Out error as solid line and the Test error as dashed line.

A. ESTSP 2007 Competition Dataset

This time series prediction benchmark includes a total of 875 values from an unknown origin. The dataset is shown in Figure 6. More information and the dataset can be found from the ESTSP 2007 conference website and the proceedings [18], [19].

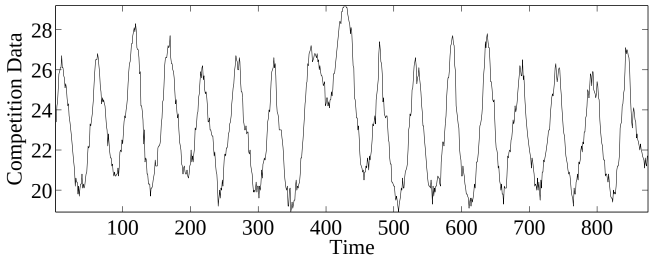

Fig. 6. ESTSP 2007 Competition dataset.

The presented methodology is applied to the dataset. Original size of the regressor is selected to be 55 [19], which means that we have originally samples in 55-dimensional space. Then, the PLS is used to decrease the dimension to 7 and finally, the DT decreases the dimension to maximum of 6 dimensions. The selections of the regressor sizes is based on earlier experience and several test runs with the projection methods.

The projection is done for each prediction time step from 1 to 50, as described by the prediction. The result from the Delta Test is shown in Figure 7 using Equation 9.

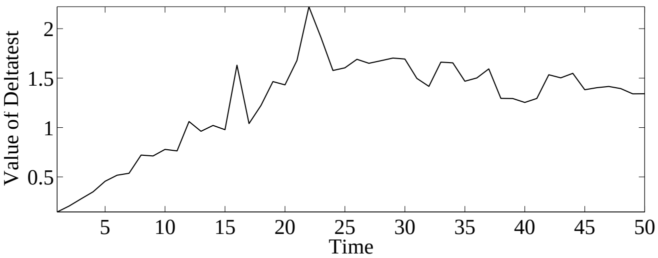

Fig. 7. ESTSP Competition dataset, noise approximation by the Delta Test with respect to prediction steps.

From Figure 7, we can see, as expected, that the Delta Test approximation of noise increases the further we go into the future with our prediction. But at a certain point, the value levels out and does not increase significantly.

The methodology is repeated 100 times using a Monte-Carlo scheme. The average is used to compute the final prediction and confidence interval is calculated using a sum of the mean and the standard deviation multiplied by ± 1.96, in order to get a 95 percent confidence interval. In Figure 8, the final prediction of the ESTSP dataset is shown and the confidence interval is shown in Figure 9.

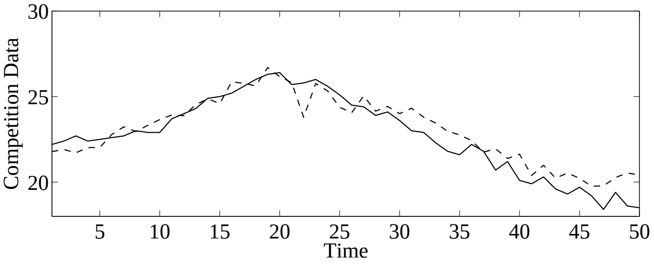

Fig. 8. ESTSP Competition dataset, prediction of 50 values. Solid line represents the real value and the dashed one the prediction.

Figure 8 shows that the prediction is good in terms of visual inspection. For 15 steps ahead, the Mean Square Error (MSE) is 0.206 and for 50 steps ahead 0.686. The prediction performances are the 6^{th} and the 2^{nd} places, respectively, according to the results of the ESTSP 2007 prediction competition [20].

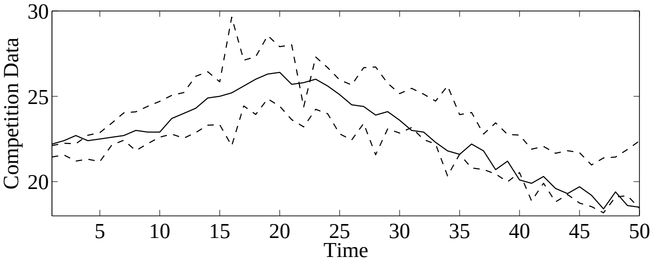

Fig. 9. Confidence intervals of 50 predicted values. The solid line represents the real value and the dashed lines the confidence intervals.

From Figure 9, we can see that the confidence interval is almost everywhere containing the real value. Only a few places are outside the bounds, mainly in the place of furthest prediction horizon.

B. Poland Electricity Load Dataset

The dataset is called Poland Electricity Load and it represents two periods of the daily electricity load of Poland during around 1500 days in the 90's [21]. The quasi-sinusoidal seasonal variation is clearly visible from the dataset.

The first part, 1400 values, is used for training and the first 30 values of the second part for testing. The learning part of the dataset is shown in Figure 10.

We follow the same procedure than in previous experiment, but this time the original input regressor size is selected to be 15 [10]. The PLS is still projecting to 7 latent variables and after that the DT maximum to 6 dimensions. In Figure 11, the final prediction of the Poland Electricity Load dataset is shown and the confidence intervals is shown in Figure 12.

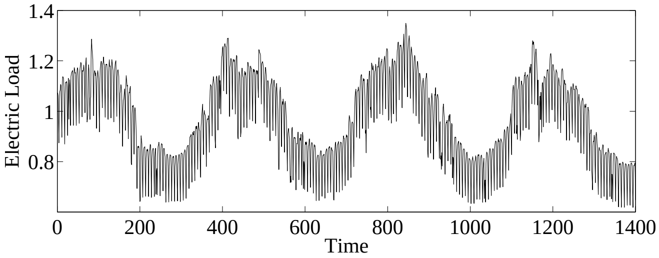

Fig. 10. Learning set of the Poland Electricity Load dataset.

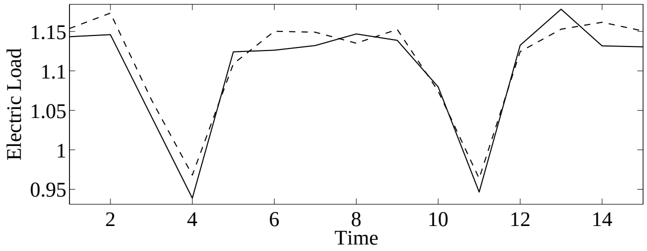

Fig. 11. Poland Electricity Load, 15 predicted values of the test set. Solid line represents the real value and the dashed one the prediction.

From Figure 11, we can see again that the prediction accuracy is very good. For 15 steps ahead, the test MSE is 0.0004.

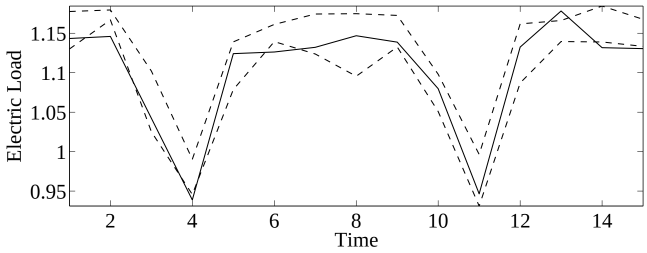

Fig. 12. Confidence intervals of 15 predicted values of the test set. The solid line represents the real value and the dashed lines the confidence intervals.

The confidence interval is not as consistent with this dataset than with the ESTSP Competition one. Still the prediction accuracy is good, even more measurements are outside the bounds.

VII. Conclusions

The proposed methodology of sophisticated variable projection combined with the OP-ELM, based on the Extreme Learning Machine, performs better than the original version of ELM. Big part of the performance improvement comes from the variable projection that provides suitable input for the OP-ELM method.

The calculation time of the whole methodology is surprisingly fast, only a couple of minutes for each prediction. Starting from the PLS projection and ending to the selection of neurons based on the combination of LARS and LOO, each part is very fast and, as the results demonstrate, provide accurate results. Furthermore, the low computational load enables the researcher to compute very long-term predictions using the Direct prediction strategy.

For further work, the proposed methodology is applied to other datasets and obtained performances are compared to the existing ones. Improvements to the projection and selection techniques are also researched.

References

[1] L. Ljung, *System identification theory for User*. Prentice-Hall, Englewood CliPs, NJ, 1987.

[2] A. Weigend and N. Gershenfeld, *Times Series Prediction: Forecasting the Future and Understanding the Past*. Addison-Wesley Publishing Company, 1994.

[3] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, December 2006.

[4] F. Rossi, A. Lendasse, D. François, V. Wertz, and M. Verleysen, "Mutual information for the selection of relevant variables in spectrometric nonlinear modelling," *Chemometrics and Intelligent Laboratory Systems*, vol. 80, pp. 215–226, 2006.

[5] L. J. Herrera, H. Pomares, I. Rojas, M. Verleysen, and A. Guilén, "Effective input variable selection for function approximation," in *Artificial Neural Networks: ICANN 2006*, ser. Lecture Notes in Computer Science, S. B. . Heidelberg, Ed., vol. 4131/2006, 2006, pp. 41–50.

[6] D. François, F. Rossi, V. Wertz, and M. Verleysen, "Resampling methods for parameter-free and robust feature selection with mutual information," *Neurocomput.*, vol. 70, no. 7-9, pp. 1276–1288, 2007.

[7] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," in *Annals of Statistics*, 2004, vol. 32, no. 2, pp. 407–499.

[8] R. H. Myers, *Classical and Modern Regression with Applications, 2nd edition*. Pacific Grove, CA, USA: Duxbury, 1990.

[9] G. Bontempi, M. Birattari, and H. Bersini, "Recursive lazy learning for modeling and control," in *European Conference on Machine Learning*, 1998, pp. 292–303.

[10] A. Sorjamaa, J. Hao, N. Reyhani, Y. Ji, and A. Lendasse, "Methodology for long-term prediction of time series," *Neurocomputing*, vol. 70, no. 16-18, pp. 2861–2869, October 2007.

[11] A. Jones, "New tools in non-linear modeling and prediction," *Computational Management Science*, vol. 1, pp. 109–149, 2004.

[12] H. Wold, "Partial least squares," in *Encyclopedia of Statistical Sciences*. New York: Wiley, 1985, vol. 6, pp. 581–591.

[13] S. Haykin, *Neural Networks - A Comprehensive Foundation, 2nd edition*. Upper Saddle River, New Jersey 07458: Prentice Hall, 1999.

[14] Y. Miche, P. Bas, C. Jutten, O. Simula, and A. Lendasse, "A methodology for building regression models using extreme learning machine: Op-elm." Accepted for publication in ESANN 2008 conference, 2008.

[15] W. T. Miller, F. H. Glanz, and L. G. Kraft, "Cmac: An associative neural network alternative to backpropagation," in *Proceedings of the IEEE*, October 1990, vol. 70, no. 10, pp. 1561–1567.

[16] C. R. Rao and S. K. Mitra, *Generalized Inverse of Matrices and Its Applications*. John Wiley & Sons Inc, January 1972.

[17] T. Similä and J. Tikka, "Multiresponse sparse regression with application to multidimensional scaling," in *Lecture Notes in Computer Science*, vol. 3697. International Conference on Artificial Neural Networks (ICANN), Warsaw, Poland, September 11-15, 2005, pp. 97–102.

[18] ESTSP2007 Conference: http://www.estsp.org.

[19] A. Lendasse, Ed., *Proceedings of ESTSP 2007*. P.O. Box 5400, 02015 HUT, Finland: Helsinki University of Technology, Laboratory of Computer and Information Science, ISBN: 978-951-22-8601-0, 2007.

[20] Http://www.cis.hut.fi/projects/tsp/ESTSP.

[21] Http://www.cis.hut.fi/projects/tsp/?page=Timeseries.