Sensitivity based selection of inputs and delays for nonlinear ARX models

Martin Macaš
Czech Institute of Informatics, Robotics and Cybernetics,
Czech Technical University in Prague
Jugoslávských partyzánů 1580/3
160 00 Prague 6, Czech Republic
macasm1@ciirc.cvut.cz

Fabio Moretti
Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA)
Italy
fabio.moretti@enea.it

ABSTRACT

In this paper, an extension of sensitivity based pruning (SBP) method for Nonlinear AutoRegressive models with eXogenous inputs (NARX) model is presented. Besides the inputs, input and output delays are simultaneously pruned in terms of the backward elimination. The concept is based on replacement of some regressors by their mean value, which corresponds to the removal of influence of the particular regressors from the network. The method is demonstrated on two datasets. Firstly, one artificial generator is used to test if the method is able to find an optimal set of inputs and delays. Further, the method is used for prediction of gas consumption of a simulated heating for an office building. It is shown that the SBP significantly reduces the complexity of the NARX network without any significant performance degradation. Moreover, it is hypothesized than SBP can be more important for NARX than for simple feedforward neural network, because NARX is more prone to overfitting and has problems with stability.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning—*Connectionism and neural nets*

General Terms

Algorithms, Experimentation.

Keywords

Time series prediction, neural networks, feature selection, pruning, NARX.

1. INTRODUCTION

In recent years, technological progress enables to measure and extract rapidly growing numbers of variables. Often multiple measurements are performed at successive time instants, which corresponds to multivariate time series. The time series are often further processed to solve a certain task like prediction or classification. In some data sets, a temporal context plays an important role. The temporal context means that the value of a variable depends on the past values of other variables. A model that can deal with such temporal dependencies is called here temporal context aware (TCA) model. An important property of such models is that their response on the same input can be different in two different times. Such models are able to predict or classify temporal patterns. Typical TCA models are recurrent neural networks. Thanks to their computational capabilities [21], Nonlinear AutoRegressive models with eXogenous inputs (NARX) are a popular family of models, which, if used in closed-loop manner, can be understood as a recurrent neural networks. NARX have been used in many system identification and time-series prediction applications. One of most important points of an application of NARX network is a proper selection of inputs, input delays and output delays. Unfortunately, not enough attention is paid to those points, although they definitely help to avoid overfitting, reduce time and training data requirements, and can even increase the prediction performance. Since NARX network is one of temporal context aware methods, we believe that also selection of inputs must be temporal context aware and should be performed simultaneously with selection of input and output delays.

In most studies, common non-TCA filter feature selection criteria have been used (e.g. entropy based [10], Fisher's criterion [3], minimal-redundancy-maximal-relevance [1]). In [1], bi-directional long short-term memory was used to recognize words in Arabic text and the minimal-redundancy-maximal-relevance (mRMR) technique was chosen among many other tested methods, because it offered the best compromise between accuracy and speed. Although the studies usually report an improvement of the performance or computational time brought by the feature selection, we hypothesize that this approach may not be suitable in some cases. The main reason is that these common filter criteria do not take temporal context into account and thus they cannot guide a search mechanism into a feature subset that corresponds to a sufficiently good performance of the TCA classification or prediction model. The non-TCA filter criteria can be useful and even better than wrapper criteria for common non-TCA classification models (e.g. Fisher discriminant or Support Vector Machines) or prediction models

BICT 2015, December 03-05, New York City, United States

DOI 10.4108/eai.3-12-2015.2262480

(e.g. feed-forward neural networks). However, their use as a performance approximation for the TCA models can seriously fail. For example, in [20], the number of inputs to a RNN was reduced by employing a binary gravitational search and binary particle swarm optimization algorithms for feature selection. However, the accuracy of optimum path forest classifier was used as the feature selection criterion. Since the optimum path forrest classifier is static model that does not take into account the temporal context of the time series, features that are optimal for this classifier can differ from features that are optimal for RNN.

One easy option is to use feature selection for non-TCA feed-forward neural networks that use so-called saliency measures. Input selection can be understood as a pruning of inputs from the network. Popular algorithms developed for feed-forward neural networks are optimal brain surgeon [5] and optimal brain damage [9] based on saliency based weight ranking. For recurrent neural network, optimal brain surgeon was adapted in [7] for pruning a general dynamic neural network.

Although the pruning mechanisms are used to remove network connections or nodes, not sufficient effort was devoted to selection of inputs. A delay damage algorithm was introduced in [8], which performs selection of model order (number of input and output lags) through a pruning. The second order derivative of the error with respect to the delayed inputs and outputs was used as the criterion. For four experimental data sets, the method significantly improved the generalization and predictive performance of NARX. Another approach is Signal-to-Noise Ratio introduced by Bauer et al. [2] for evaluation of features for feed-forward neural networks, which is weight-based method, because it uses only weights of the neural network.

On the other hand Sensitivity based Pruning [16] developed by Moody evaluates the effect of removing an input variable from the fully connected network on its training error. Such method can be understood as output-based, because it uses information from network's output. Laine and Bauer [11] compared and assessed the Signal-to-Noise Ratio approach and the Sensitivity based Pruning for Elman network on a very limited number of data sets and observed that the selection methods performed equivalently. The sensitivity based pruning was used for selection of inputs for NARX network for prediction of heating gas consumption or thermal discomfort [14]. It was observed in both cases that the feature selection brings significant benefits for recurrent neural networks in terms of 50% input dimensionality reduction without a significant increase of prediction performance.

This paper enhances the sensitivity based pruning for simultaneous selection of inputs, input delays and output delay. Section 2 describes the original Moody's sensitivity based pruning method, its extended application is described in section 3 and some practical issues are discussed in section 4. Section 5 experimentally demonstrates the method on artificial and real-world time-series. Finally, section 6 concludes the paper and provides some potential future work examples.

2. SENSITIVITY BASED PRUNING

To select proper features tailored for particular feed-forward network, one can use well known sensitivity based method developed by Moody [17]. It is called Sensitivity based Pruning (SBP) algorithm. It evaluates a change in training mean squared error (MSE) that would be obtained if ith input's influence was removed from the network. The removal of influence of the input is simply modeled by replacing it by its average value. Let $[\mathbf{x}(1), \mathbf{x}(2), \mathbf{x}(3), \ldots \mathbf{x}(N))]$ be the multidimensional time series of length N, where $\mathbf{x}(k) = [x_1(k), \ldots, x_i(k), \ldots, x_D(k)]^\top$, be the kth of N instances of the input vector. Let $[t(1), t(2), t(3), \ldots t(N))]$ be the one dimensional time series of corresponding target outputs.

A feed-forward network can be understood as a non-linear function $y(k) = f(\mathbf{x}(k))$. Input selection seeks for a good subset of inputs $\{1, 2, \ldots, i, \ldots, D\}$. The goodness of a subset can be measured using mean squared error (MSE) on some data set. For a trained network, one can eliminate an influence of ith input $x_i(k)$ by replacing it by its average value $\sum_{k=1}^{N} x_i(k)/N$. Let $\mathbf{x}^i(k)$ be the kth data instance whose ith position is replaced by such corresponding average. The sensitivity of the network to an input is defined as absolute increase of MSE caused by the input's influence removal:

$$S_{in}^i = \frac{1}{N}\sum_{k=1}^{N}[f(\mathbf{x}^i(k)) - t(k)]^2 - \frac{1}{N}\sum_{k=1}^{N}[f(\mathbf{x}(k)) - t(k)]^2$$
$$= MSE_{in}^i - MSE. \quad (1)$$

Like in Moody's original work, also in our implementation of SBP, backward elimination was used as the search mechanism. The algorithm starts with the full set of D inputs. At each step, a target neural network is trained. Further, its sensitivity is computed for all particular inputs according to the Equation 1. The input, for which the sensitivity is the smallest one is removed from the data. Note that a new neural network is trained at each backward step.

3. ENHANCEMENT FOR NARX

Compared to the feed-forward networks, NARX network computes $y(k)$ from the following regressors:

- delayed inputs $\{\mathbf{x}(k-\delta)\}_{\delta\in\Delta}$, where the delay δ is from a predefined set of input delays $\Delta \subset \{0, 1, 2, \ldots \delta_{MAX}\}$,
- delayed outputs $\{t(k-\lambda)\}_{\lambda\in\Lambda})$, where the delay λ is from a predefined set of output delays $\Lambda \subset \{1, 2, \ldots \lambda_{MAX}\}$.

Besides the inputs, also input and output delays can be eliminated. For a trained network, such elimination can be performed similarly to previous one described by Moody [17]:

- An influence of input i is removed by replacing $x_i(k-\delta)$ by $\sum_{k=\delta+1}^{N} x_i(k-\delta)/N$ for all k and δ, i.e. by replacing ith component of all inputs and their delayed versions by corresponding average value.

- An influence of input delay δ is removed by replacing $x_i(k-\delta)$ by $\sum_{k=\delta+1}^{N} x_j(k-\delta)/N$ for all k and i, i.e. by replacing all components of inputs delayed by δ by corresponding average value.

- An influence of output delay λ is removed by replacing $t(k-\lambda)$ by $\sum_{k=\lambda+1}^{N} t(k-\lambda)/N$ for all k, i.e. by replacing output delayed by λ by its corresponding average value.

Note that delayed inputs are replaced for all i, which means that the influence of the delay is removed from all inputs. This can degrade the results for systems with significantly different delays for different outputs. On the other hand, this saves computational requirements by selecting M from $D + |\Delta| + |\Lambda|$ original entities instead of $D \times |\Delta| + |\Lambda|$. Although this naive approach is used in all underlying experiments, the exhaustive approach can be easily implemented.

Further, let MSE_{in}^{i}, MSE_{din}^{δ} or MSE_{dout}^{λ} be the MSE obtained for network from which the influence of input i, input delay δ or output delay λ was removed, respectively.Then, the sensitivity of the network to input, input delay or output delay is defined as absolute increase of MSE caused by removing the input, input delay or output delay, respectively:

$$S_{in}^{i} = MSE_{in}^{i} - MSE \tag{2}$$

$$S_{din}^{\delta} = MSE_{din}^{\delta} - MSE \tag{3}$$

$$S_{dout}^{\lambda} = MSE_{dout}^{\lambda} - MSE. \tag{4}$$

SBP algorithm starts with the full set of inputs $\{1, \ldots, D\}$, input delays Δ and output delays Λ. At each step, a target neural network is trained, sensitivities defined above are computed for all remaining inputs, input delays and output delays. The input, input delay or output delay for which the sensitivity is smallest, is pruned. Those backward steps repeat until a stopping condition is met.

4. PRACTICAL ISSUES

This section describes two important implementation issues for described SBP. First, it must be pointed out that underlying implementation differs from the original Moody's approach in error estimate used for sensitivity computation. Compared to the original Moody's approach [17], which uses only training set for the sensitivity computation (resubstitution estimate), we split the training set into two parts - on the first we train the network and on the second we compute the sensitivity (hold-out estimate).

Second, an obvious question is, how many inputs and delays to select. The backward elimination described above effectively orders the inputs, input delays and output delay, but does not answer this question. One possible solution is based on validation dataset and minimum validation error principle. The validation dataset is used to test the particular models of different numbers of inputs and delays. The number of inputs and delays is then decided according to the minimal validation error.

5. EXPERIMENTS

The proposed approach is validated on two time series data. To ensure availability of sufficient testing data and validity of results, both datasets are generated by computer. First, a simple system with known analytically expressible dynamics is identified to demonstrate that the method efficiently eliminates unimportant regressors. Second, a real application on modeling of heating consumption is presented.

The predictor is the NARX network with one hidden layer whose delayed output is connected back to the input [12]. The network was simulated in Neural Network Toolbox for Matlab [15]. Three hidden units were used. The hidden and output units use the sigmoid and linear transfer function, respectively. The mean squared error was minimized by Levenberg-Marquardt algorithm, because of its relatively high speed, and because it is highly recommended as a first-choice supervised algorithm by Matlab Neural Network toolbox, although it does require more memory than other algorithms [15]. The training was stopped after 100 epochs without any improvement or after the number of training epochs exceeded 300 or if the error gradient reached 10^{-7}.

5.1 Artificial data

5.1.1 Data generation

First, 30 inputs are generated by pseudorandom methods to efficiently excite frequencies and amplitude levels. Pseudorandom binary signals are well suited for linear system identification, because they imitate white noise in discrete time with a deterministic signal and excite all frequencies equally well. However, amplitude modulated pseudo-random binary signals (APRBS) are more appropriate for non-linear identification methods [18, Chapter 17.7], because they have changing amplitude and cover more operating conditions of interest. Therefore, the dynamics of the input variables is defined here using APRBS. All the input signals are defined as maximally shifted APRBS having a clock period of 1 [13, Chapter 13.3].

The output dynamics is:

$$t(k) = \frac{x_1(k-2)x_2(k-4) + x_1(k-4)x_2(k-2)}{1 + x_2(k-2)t(k-3) + t^2(k-3)}. \tag{5}$$

Thus, the relevant regressors are $x_1(k-2)$, $x_1(k-4)$, $x_2(k-2)$, $x_2(k-4)$, $t(k-3)$ and a proper selector should select inputs $\{1, 2\}$, input delays $\{2, 4\}$, and the only relevant autoregressive delay: $\{3\}$.

5.1.2 Results

To evaluate the results, training, validation and testing data were generated. The training data consisted of 400 samples. To choose the output dimensionality, neural networks with different numbers of regressors were tested on an independent validation data set with 400 samples. The testing errors of the method were estimated on 3000 samples, which are not used in any part of the predictor design process. In original setting, there are 30 inputs, 7 candidate input delays $\{0 \ldots 6\}$ and three candidate output delays $\{1 \ldots 3\}$. The total number of original entities inputs, input delays and output delays is thus 30, 7 and 3, respectively. This corresponds to selection of M of $30 + 7 + 3 = 40$ entities corresponding to $30 \times 7 + 3 = 213$ original regressors.

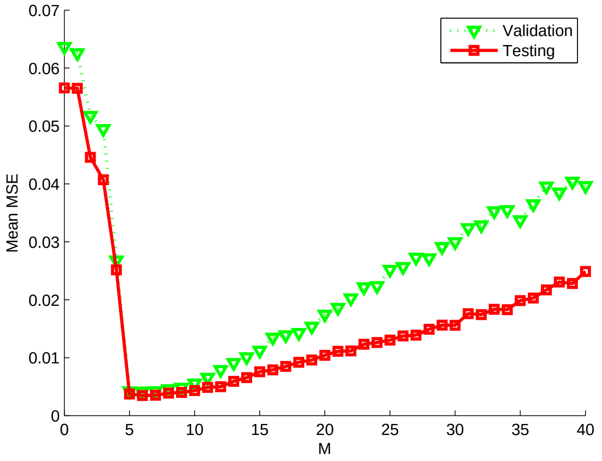

Figure 1: Dependence of average validation and testing MSE on number of selected inputs and delays M.

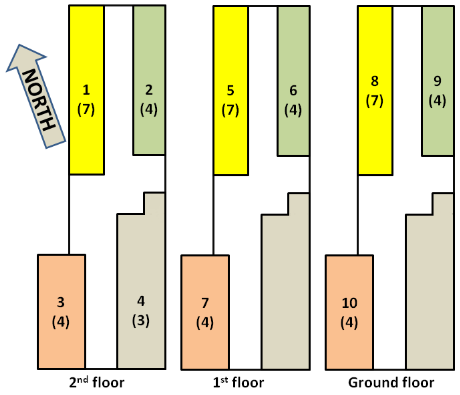

Figure 2: Partitioning of F40 building zones. Numbers denote the number of zones and numbers in brackets denote corresponding number of fan-coils.

The dependence of validation and testing MSE averaged over 100 runs on the number of selected inputs and input and output delays can be found in Figure 1. One can observe that there is an evident peak of both averaged validation and testing MSE for $M = 5$, which corresponds to size of the optimal set of relevant regressors. The peak appears in all 100 runs and in all those runs, it corresponds to selected inputs $\{1, 2\}$, input delays $\{2, 4\}$, and autoregressive delay $\{3\}$. For the artificial data set, the method always found optimal regressors that correspond exactly to the generator dynamics described in the previous section. This proves the ability of the method to find optimal set of inputs and delays.

5.2 Simulated building consumption data

5.2.1 Data generation

To demonstrate a practical importance of the proposed approach, the prediction of total gas consumption of modeled office building heating is presented. An hourly prediction for the same building is described in details in [14]. The office building is located at Casaccia Research Centre of Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA). The structure is composed of three floors and a thermal plant in the basement. There are 41 offices of different size with a floor area ranging from 14 to 36 m^2, two electronic data processing rooms each of about 20 m^2, four laboratories, one control room and two meeting rooms. Each office room has from one up to two occupants. For each room and laboratory, as thermal exchangers, there are fan-coils with on-off fan speed controlled by a proper thermostat with hysteresis equal to $1\ ^\circ\ C$. During winter, the thermal plant produces heat by a traditional natural gas boiler. This study is related only to heating during winter season. The thermal fluid circulation into fan-coil circuits is ensured by a triplet of centrifugal pumps. The building is equipped with an advanced monitoring system collecting data from sensors of environmental conditions and electrical and thermal energy consumption.

In order to simulate testing data of sufficient sample sizes, a Matlab Simulink simulator based on Heat, Air and Moisture model for Building and Systems Evaluation [6] was used. In particular, the building was divided into ten controllable thermal zones according to different thermal behavior depending on solar radiation exposure. Therefore a zone consists of a group of rooms with similar climatic conditions and the same climate control policy. Figure 2 shows the division into thermal zones. Although there are 15 zones at all, those that do not have sensors and remotely controllable fan coils are not considered.

The gas consumption is derived by integration of the natural gas mass flow which depends directly from the discharge and return water temperature at the thermal plant and from the thermal plant efficiency. The fan-coils are modeled by the ε-Number of Transfer Units (ε-NTU) method [4] which allows to derive the heat injected in the zones and the outlet water temperatures from known zone air temperatures, fan-coil inlet water flows and fan speeds. The inputs of the simulation are indoor temperature set points, current air temperatures inside the zones and external meteorological data. The summary of the inputs can be found in Table 1. The main task is to predict total gas consumption in the following 12 hours denoted as $y(t)$.

The behavior of supply water temperature set point was controlled by a simple weather compensation rule. To excite the dynamics of the system in a proper degree, we also added a random component. The value of the temperature set point is Gaussian random number with standard deviation $4^\circ\ C$ and mean equal to $85 - 2T_e$, where T_e is the mean of previous day external temperature. The behavior of air temperature set points differs for daytime and nighttime hours. Between 7 a.m. and 7 p.m., they are also Gaussian random numbers with mean $21.5^\circ\ C$ and standard deviation $1^\circ\ C$, which guarantees an acceptable level of thermal comfort. Moreover, there is a saturation under $20^\circ\ C$ and above $25^\circ\ C$. Between 7 p.m. and 7 a.m., there is a nighttime regime and air temperature set points are Gaussian random numbers with mean $20^\circ\ C$, standard deviation $1^\circ\ C$ and upper saturation level $23^\circ\ C$. A similar randomized approach was used

Table 1: Summary of the network inputs

Inputs	Names
$x_1(k)$	the value of the air temperature set point constant for the following 12 hours
$x_2(k)$	the value of the supply water temperature set point constant for the following 12 hours
$x_3(k)\dots x_{12}(k)$	ten values of instantaneous air temperature in ten zones
$x_{13}(k)\dots x_{19}(k)$	arithmetic means of weather variables computed over last 12 hours

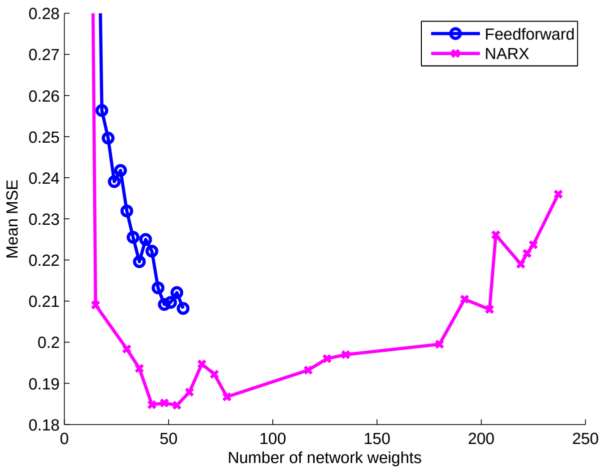

Figure 3: Comparison of input selection for feedforward network and proposed input and delay selection for NARX.

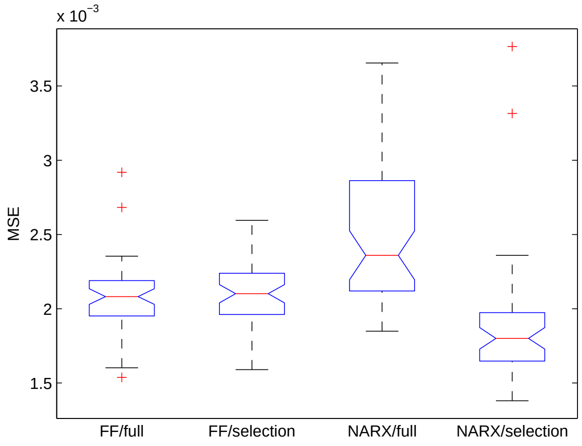

Figure 4: Boxplots of MSE for feedforward network with full set of inputs (FF/full) and with selection of inputs (FF/selection) and for NARX network with full set of delays and inputs (NARX/full) and with selection of inputs and delays (NARX/selection).

in [19].

5.2.2 Results

The training data consisted of two simulated heating seasons, 2004/2005 and 2005/2006. To choose the output dimensionality, neural networks with different numbers of regressors were tested on an independent validation data set 2006/2007. The testing errors of the method were computed on the $2008-2012$ data sets, which are not used in any part of the predictor design process and are large enough to provide valid estimate of the real prediction error.

The results are shown in Figure 3, where only testing error is depicted. The SBP of inputs and delays for NARX is compared to selection of inputs for common feed-forward network with the same number of hidden units, the same training algorithms and the other settings. To make the values on horizontal axis comparable, number of network weights is used. For three hidden units, maximum number of weights (without selection of inputs or delays) for feedforward network is 57, while maximum number of weights for NARX is 237. This is why the graphs span different ranges on horizontal axis.

One can observe that without the selection of inputs and delays, NARX network with the original settings is significantly worse than the simple feedforward network (Wilcoxon sum-rank test, $\alpha = 0.05$). On the other hand, if SBP is used, NARX network becomes better as the elimination proceeds until it permanently outperforms all feedforward networks of comparable number of weights. Moreover, one can see that the input selection for feedforward network causes a performance degradation. This demonstrates that the input and delays selection is much more important for NARX network than the input selection is for the feed-forward network. Although this result is obtained for a specific case of one particular application, it is highly probable that this will be also obtained for other data, because the NARX network has a more complex dynamics and its irrelevant connections cause higher prone to overfitting and smaller stability of the network.

Figure 3 does not consider a decision about number of inputs and delays. Therefore, the final MSE obtained by minimum validation error principle (see section 4) averaged over 100 runs is compared by boxplots in Figure 4. The figure supports the previous evidence that the pruning is more important for NARX network than for feedforward network. Although NARX network is significantly worse for the original set of inputs and delays, it significantly outperforms the feedforward network if the pruning is used.

Here we do not interpret the results of the selection. We do not describe, which inputs and delays were selected and which not. This is out of the scope of the paper, which only proposes the new method and uses the application only as a tool for proving the concept.

6. CONCLUSIONS

The paper proposes an extension of SBP method to NARX networks. It shows that the method efficiently selects inputs, simultaneously removes input and output delays and does not degrade the prediction performance. The approach is obviously not limited to NARX neural network, but is easily extendable to other networks. The extension would be straightforward. The network must be simulated on the data and the signal of connection, which is to be removed is collected for all data samples. Than the signal is replaced by its mean value computed from the collected signal.

For future, some practical implementation issues described in section 4 should be focused in more details. First, it should be examined, if and when the sensitivity based on hold-out error estimate (computed on validation data) brings some benefits to the resubstitution error estimate (computed on training data). Further, different methods for final decision about number of inputs and delays or different stopping condition for the backward search should be proposed. Finally, the influence of the use of the naive approach (considering the same delays for all inputs) instead of computationally more intensive exhaustive approach (considering and eliminating different delays from different inputs) should be examined.

7. REFERENCES

[1] Abandah, G. A., Jamour, F. T., and Qaralleh, E. A. Recognizing handwritten Arabic words using grapheme segmentation and recurrent neural networks. *International Journal on Document Analysis and Reecognition 17*, 3 (SEP 2014), 275–291.

[2] Bauer, K. W., Alsing, S. G., and Greene, K. A. Feature screening using signal-to-noise ratios. *Neurocomputing 31*, 1âĂŞ4 (2000), 29 – 44.

[3] Beghdad, R. Applying Fisher's filter to select KDD connections' features and using neural networks to classify and detect attacks. *Neural Network World 17*, 1 (2007), 1–16.

[4] Bergman, T., Lavine, A., and Incropera, F. *Fundamentals of Heat and Mass Transfer, 7th Edition.* John Wiley & Sons, Incorporated, 2011.

[5] Cun, Y. L., Denker, J. S., and Solla, S. A. Advances in neural information processing systems 2. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990, ch. Optimal Brain Damage, pp. 598–605.

[6] De Wit, M. *HAMBASE: Heat, Air and Moisture Model for Building And Systems Evaluation.* Technische Universiteit Eindhoven, Faculteit Bouwkunde, 2006.

[7] Endisch, C., Hackl, C., and Schroeder, D. Optimal brain surgeon for general dynamic neural networks. In *Progress In Artificial Intelligence, Proceedings* (2007), Neves, J and Santos, MF and Machado, JM, Ed., vol. 4874 of *Lecture Notes in Artificial Intelligence*, pp. 15–28. 13th Portuguese Conference on Artificial Intelligence, Guimaraes, Portugal, DEC 03-07, 2007.

[8] Giles, C., Lin, T., Horne, B., and Kung, S. The past is important: A method for determining memory structure in NARX neural networks. In *IEEE World Congress On Computational Intelligence* (1998), pp. 1834–1839.

[9] Hassibi, B., Stork, D., and Wolff, G. Optimal brain surgeon and general network pruning. In *Neural Networks, 1993., IEEE International Conference on* (1993), pp. 293–299 vol.1.

[10] He, Y.-J., Zhu, Y.-C., Duan, D.-X., and Sun, W. Application of neural network model based on combination of fuzzy classification and input selection in short term load forecasting. In *Proceedings of 2006 International Conference on Machine Learning and Cybernetics, Vols 1-7* (2006), IEEE, pp. 3152–3156.

[11] Laine, T. I., and Bauer, K. W. Input feature selection for automatic target recognition of temporal data. *Military Operations Research 10*, 2 (2005), 51–65.

[12] Leontaritis, I., and Billings, S. Input-output parametric models for non-linear systems part I: deterministic non-linear systems. *International Journal of Control 41*, 2 (1985), 303–328.

[13] Ljung, L., Ed. *System Identification (2Nd Ed.): Theory for the User.* Prentice Hall PTR, 1999.

[14] Macas, M., Lauro, F., Moretti, F., Pizzuti, S., Annunziato, M., Fonti, A., Comodi, G., and Giantomassi, A. Sensitivity based feature selection for recurrent neural network applied to forecasting of heating gas consumption. In *International Joint Conference SOCO2014-CISIS2014-ICEUTE2014*, vol. 299 of *Advances in Intelligent Systems and Computing.* Springer International Publishing, 2014, pp. 259–268.

[15] Mathworks. Neural Network Toolbox for Matlab ver. 2012b, 2012.

[16] Moody, J. *From Statistics to Neural Networks: Theory and Pattern Recognition Applications.* Springer-Verlag, 1994, ch. Prediction risk and neural network architecture selection.

[17] Moody, J. E. The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. In *NIPS* (1991), Morgan Kaufmann, pp. 847–854.

[18] Nelles, O. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models.* Engineering online library. Springer, 2001.

[19] P.M.Ferreira, A.E.Ruano, S.Silva, and E.Z.E.Conceição. Neural networks based predictive control for thermal comfort and energy savings in public buildings. *Energy and Buildings 55*, 0 (2012), 238 – 251.

[20] Sheikhan, M. Generation of suprasegmental information for speech using a recurrent neural network and binary gravitational search algorithm for feature selection. *Applied Intelligence 40*, 4 (2014), 772–790.

[21] Siegelmann, H., Horne, B., and Giles, C. Computational capabilities of recurrent narx neural networks. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 27*, 2 (Apr 1997), 208–215.