Variable Selection in the Kernel Regression Based Short-Term Load Forecasting Model

Grzegorz Dudek

Department of Electrical Engineering, Czestochowa University of Technology,
Al. Armii Krajowej 17, 42-200 Czestochowa, Poland
dudek@el.pcz.czest.pl

Abstract. The short-term load forecasting is an essential problem in energy system planning and operation. The accuracy of the forecasting models depends on the quality of the input information. The input variable selection allows to chose the most informative inputs which ensure the best forecasts. To improve the short-term load forecasting model based on the kernel regression four variable selection wrapper methods are applied. Two of them are deterministic: sequential forward and backward selection and the other two are stochastic: genetic algorithm and tournament searching. The proposed variable selection procedures are local: the separate subset of relevant variables is determined for each test pattern. Simulations indicate the better results for the stochastic methods in relation to the deterministic ones, because of their global search property. The number of input variables was reduced by more than half depending on the feature selection method.

Keywords: feature selection, kernel regression, genetic algorithm, tournament feature selection, short-term load forecasting.

1 Introduction

The short-term load forecasting (STLF) is extremely important to balance the electricity generated and consumed at any moment. Precise load forecasts are necessary for electric companies to make important decisions connected with electric power production and transmission planning, such as unit commitment, generation dispatch, hydro scheduling, hydro-thermal coordination, spinning reserve allocation and interchange evaluation.

Many STLF models have been proposed. Conventional STLF models use smoothing techniques, regression methods and statistical analysis. In recent years artificial intelligence methods have been widely used to STLF: neural networks, fuzzy systems and expert systems.

In this article nonparametric regression method is applied to STLF. The regression relationship can be modelled as [1]:

$$y = m(x) + \varepsilon \tag{1}$$

L. Rutkowski et al. (Eds.): ICAISC 2012, Part II, LNCS 7268, pp. 557–563, 2012.

where: y is the response variable, x – the predictor, ε – the error, which is assumed to be normally and independently distributed with zero mean and constant variance, $m(x) = \mathrm{E}(Y \mid X = x)$ is a regression curve.

The aim of regression is to estimate the function m. This task can be done essentially in two ways. The first approach to analyze a regression relationship is called parametric since it is assumed that the mean curve m has some prespecified functional form and is fully described by a finite set of parameters (e.g. a polynomial regression equation). In the alternative nonparametric approach the regression curve does not take a predetermined form but is constructed according to information derived from the data. The regression function is estimated directly rather than to estimate parameters. Most methods of nonparametric regression implicitly assume that m is a smooth and continuous function. The most popular nonparametric regression models are [1]: kernel estimators, k-nearest neighbour estimators, orthogonal series estimators and spline smoothing.

In [2] to STLF the multivariate generalization of the kernel Nadaraya-Watson estimator was described:

$$\hat{m}(\mathbf{x}) = \frac{\sum_{j=1}^{n} \prod_{k=1}^{d} K\left(\frac{x_k - x_{j,k}}{h_k}\right) y_j}{\sum_{j=1}^{n} \prod_{k=1}^{d} K\left(\frac{x_k - x_{j,k}}{h_k}\right)}, \tag{2}$$

where n is the size of the random sample:

$$\begin{bmatrix} y_1 \\ \mathbf{x}_1 \end{bmatrix}, \begin{bmatrix} y_2 \\ \mathbf{x}_2 \end{bmatrix}, \ldots, \begin{bmatrix} y_n \\ \mathbf{x}_n \end{bmatrix}, \tag{3}$$

d is the dimension of the input pattern vector $\mathbf{x}_j = [x_{j,1}\ x_{j,2}\ \ldots\ x_{j,d}]$, which represents a vector of hourly power system loads in the following hours of the day preceding the day of forecast $\mathbf{L}_j = [L_{j,1}\ L_{j,2}\ \ldots\ L_{j,d}]$:

$$x_{j,k} = \frac{L_{j,k} - \overline{L}_j}{\sqrt{\sum_{l=1}^{d} (L_{j,l} - \overline{L}_j)^2}}, \tag{4}$$

and y_j is the encoded value the forecasted system load $L_{j+\tau,k}$ at the kth hour of the day $j+\tau$ ($\tau = 1, 2, \ldots$ is the forecast horizon):

$$y_j = \frac{L_{j+\tau,k} - \overline{L}_j}{\sqrt{\sum_{l=1}^{d} (L_{j,l} - \overline{L}_j)^2}}. \tag{5}$$

$\overline{L}_j$ in (4) and (5) is the mean load of day j.

The Gaussian kernel function used in (2) is of the form:

$$K(t)=\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{t^2}{2}\right), \tag{6}$$

and $h \in \Re^+$ is a bandwidth (smoothing parameter).

The choice of a kernel is not as important as the choice of a bandwidth value. The bandwidth values decide about the bias-variance tradeoff of the estimator. The small bandwidth value results in undersmoothing, whereas the large value results in oversmoothing. In [3] there was shown that good results are obtained when h_k is calculated using the Scott's rule:

$$h_k=\hat{\sigma}_k n^{-1/(d+4)}, \tag{7}$$

where $\hat{\sigma}_k$ is the sample standard deviation of x_k.

Estimator (2) depends on how many and which variables x_k are inputs of the model. In this article some wrapper methods of variable selection (VS) are tested: sequential forward selection (SFS), sequential backward selection (SBS), genetic algorithm (GA) and tournament feature selection (TFS) [4].

2 Methods of Variable Selection to the Kernel Regression Model

The proposed methods of VS can be divided on deterministic and stochastic ones. SFS and SBS [5], which are suboptimal strategies, belong to the first group. They based on simple greedy heuristics. SFS adds one new feature to the current set of selected features in each step. SBS starts with all the possible features and discards one at the time. The main drawback of these algorithms is that when a feature is selected or removed this decision cannot be changed. This is called the nesting problem. The extension of these strategies is plus *l*-take away *r* method and floating search method, where forward and backward selection algorithms are used alternately.

More effective, global optimization of the input variable space provide stochastic methods, such as GA. GA with binary representation is naturally adapted to solve problems of combinatorial optimization with binary variables, which include the VS problem. The GA, as the method independent on domain, has been applied to many optimization problems because of their robustness in search for large spaces and mechanism of escaping from the local minima. Search for the solution space in GA is conducted in parallel by population of chromosomes which encode the solutions. GA for VS was applied to various models: classifiers, clustering and approximation models.

In the GA approach, the given variable subset is represented as a binary string (chromosome) with a zero or one in position *i* denoting the absence or presence of feature *i*: $\mathbf{b} = (b_1, b_2, \ldots, b_d) \in \{0, 1\}^d$. Each chromosome is evaluated taking into account the model error (the forecast error here). It may survive into the next generation and reproduce in dependence on this evaluation (fitness). New chromosomes are created from old ones by the process of their crossover and mutation. One-point crossover and classical binary mutation are applied in this approach. Binary tournament is used as a chromosome selection method.

The TFS method was introduced in [4] as an alternative to other stochastic VS methods such as GA and simulated annealing. In comparison to other combinatorial optimization stochastic methods TFS is distinguished by simplicity. There is only one parameter in TFS controlling the global-local search properties which makes this algorithm easy to use.

Data representation in TFS is the same as in GA. TFS explores the solution space starting from an initial solution and generating new ones by perturbing it using a mutation operator. This operator switches the value of one randomly chosen bit (but different for each candidate solution) of the parent solution. When the set of new l candidate solutions is generated (l represents the tournament size), their evaluations are calculated. The best candidate solution (the tournament winner), with the lowest value of the error function (MAPE here), is selected and it replaces the parent solution, even if it is worse than the parent solution. This allows us to escape from local minima of the error function. If l is equal to 1, this procedure comes down to a random search process. On the other hand, when $l = d$ this method becomes a hill climbing method where there is no escape from the local maxima.

This algorithm turned out to be very promising in the feature selection problem, better than a GA and simulated annealing, as well as deterministic SFS and SBS algorithms [4].

The flowchart of GA and TFS are shown in Fig. 1.

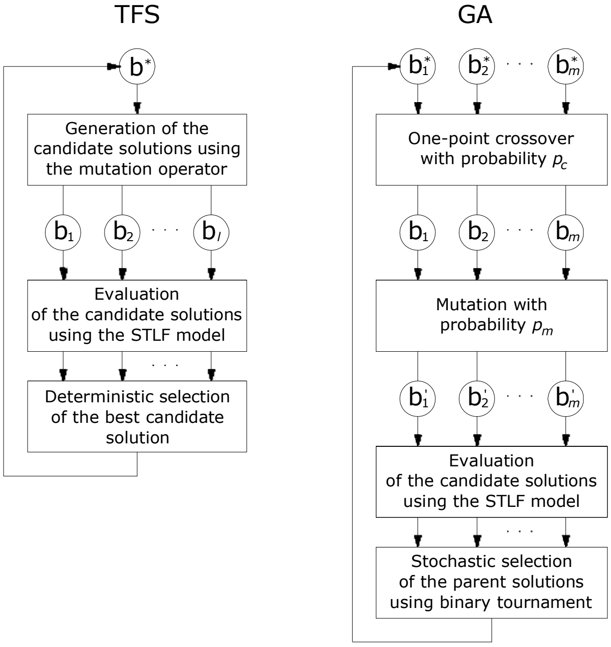
TFS
b*
Generation of the candidate solutions using the mutation operator
b1
b2
bl
Evaluation of the candidate solutions using the STLF model
Deterministic selection of the best candidate solution
GA
b1*
b2*
bm*
One-point crossover with probability pc
b1
b2
bm
Mutation with probability pm
b1'
b2'
bm'
Evaluation of the candidate solutions using the STLF model
Stochastic selection of the parent solutions using binary tournament

Fig. 1. Flowcharts of TFS and GA to variable selection in the kernel regression based STLF model

3 Application Example

The described above variable selection methods were applied to the forecasting model based on the Nadaraya-Watson estimator. The task of the model is to forecast the next day power system load (τ = 1) at hour k = 1, 6, 12, 18 and 24. Time series studied in this paper represents the hourly electrical load of the Polish power system from the period 2002-2004. This series is shown in Fig. 2.

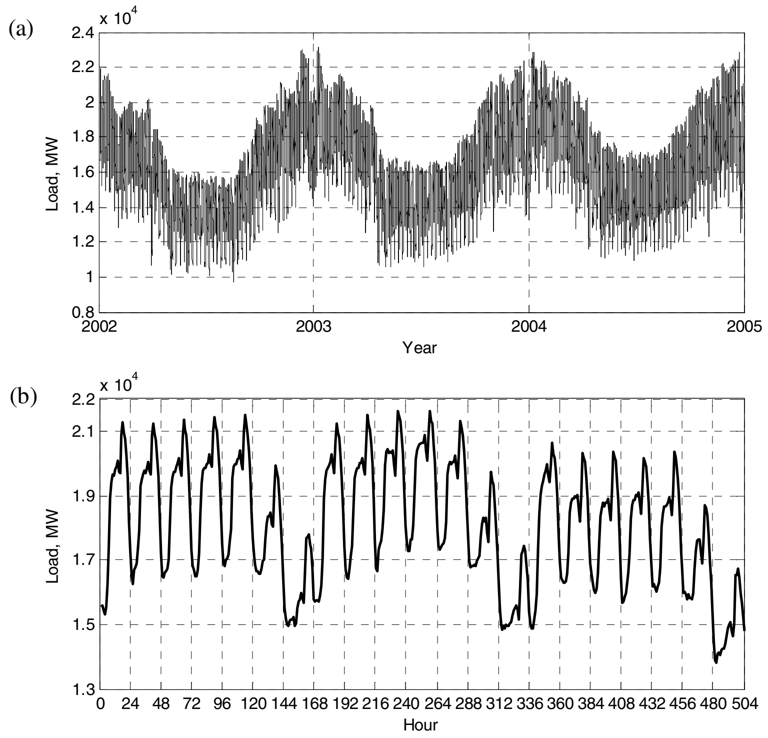

Fig. 2. The load time series of Polish power system in three year (a) and three week (b) intervals

The time series were divided into training and test parts. The test set contained 30 patterns from January 2004 (from 2 to 31 January) and 31 patterns from July 2004. The training set contained patterns from the period from 1 January 2002 to the day preceding the day of forecast.

For each forecasting task (the forecast of system load at the kth hour of the day $j+\tau$) the separate model was created using the training subset containing y-values representing loads from the days of the same type (Monday, …, Sunday) as the day of forecast and paired with them x-patterns representing the load vector of preceding days (e.g. for forecasting the Sunday load at hour k, model learns from x-patterns representing the Saturday patterns and y-values representing the loads at hour k on Sundays). This routine of model learning provides fine-tuning its parameters to the changes observed in the current behavior of the time series.

The parameters of the stochastic variable selection methods were as follows:

- GA: number of generations – 100, population size – 8, probability of mutation – 0.05, probability of crossover – 0.9,
- TFS: number of iterations – 100, tournament size $l = 8$.

The best subsets of the relevant variables were determined in leave-one-out cross-validation procedure.

The training and test errors of the Nadaraya-Watson STLF model using different methods of VS are shown in Table 1. The selected variables of the input patterns and the bandwidth values corresponding to these variables for one of the forecasting task are shown in Table 2.

Table 1. Errors of the Nadaraya-Watson STLF model using different VS methods

VS metod	January		July		Mean	
	$MAPE_{trn}$	$MAPE_{tst}$	$MAPE_{trn}$	$MAPE_{tst}$	$MAPE_{trn}$	$MAPE_{tst}$
Without VS	1.62	1.20	1.54	0.92	1.58	1.05
SFS	1.37	1.25	1.32	0.90	1.34	1.07
SBS	1.37	1.20	1.35	0.90	1.36	1.05
GA	1.38	1.17	1.34	0.90	1.36	1.03
TFS	1.34	1.17	1.30	0.90	1.32	1.03

Table 2. The bandwidth values h_k corresponding to the selected components of the input patterns in the model for hour 12 on 1 July 2004

k	1	2	3	4	5	6	7	8	9	10	11	12
SFS	0.037	-	-	-	-	-	-	-	-	-	0.041	0.040
SBS	0.045	0.035	-	-	-	0.036	-	0.058	0.048	0.041	0.051	-
GA	0.044	0.034	-	-	-	0.035	0.064	0.057	0.047	0.040	-	-
TFS	0.044	0.034	-	-	-	0.035	-	0.057	0.047	0.040	-	-
k	13	14	15	16	17	18	19	20	21	22	23	24
SFS	-	-	-	0.037	-	-	-	-	-	-	0.039	-
SBS	-	-	0.041	0.046	-	-	-	-	-	0.050	0.049	-
GA	-	-	-	-	0.076	-	-	0.092	-	-	0.048	-
TFS	-	0.037	-	0.045	-	-	-	-	-	0.049	0.048	-

All VS methods ensure the training error reduction, but only stochastic methods ensure the test error decreasing. However, the difference between the test errors in two cases: (i) using GA or TFS to VS and (ii) without VS turned out to be not statistically significant. This was proved using the Wilcoxon rank sum test for equality of medians. The 5% significance level is applied in this study. In the case of the training errors the Wilcoxon test in all cases indicates the statistically significant difference between errors.

The average reduction in the number of input pattern components was: 76% for SFS, 52% for SBS, 60% for GA and 67% for TFS, which means that filtering more than half of the x-vector components should not adversely affect the accuracy of the model. The frequency of variable selection is shown in Fig. 3. Most information about the forecast are contained in the ending components of **x** representing the system load at hour 23 and 24.

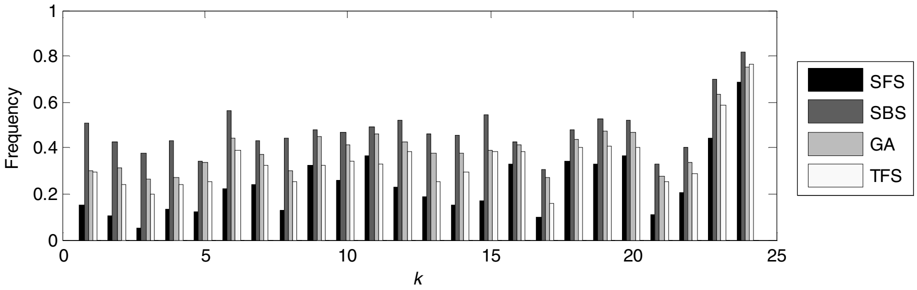

Fig. 3. The frequency of variable selection

4 Conclusion

The article describes an attempt to improve the performance of the kernel regression based STLF model by the selection of input variables. Four methods of variable selection were tested: sequential forward and backward selection, genetic algorithm and tournament feature selection. The first two are deterministic and local search methods, while the last two are stochastic and global search methods.

The empirical comparison between all of the presented variable selection method showed that the tournament feature selection provides the best performance of the forecasting model based on Nadaraya-Watson estimator. Both the training and test forecasting errors were the lowest when using this method. The global search property and simplicity (only one parameter controlling the balance between exploration and exploitation of the solution space) make the tournament feature selection easy to use and fast.

It is worth noting that the proposed routine of variable selection is local: for each test pattern a separate selection procedure and model learning is performed. Usually the feature selection methods are global, i.e. they determine one feature set for all test data. But in practice different features can be important in different regions of the input pattern space. The proposed approach enables the construction of an optimal model for the current test sample. Such a local model loses its generality but leads to the more accurate estimation of the regression curve in the neighborhood of the test point.

Acknowledgments. The study was supported by the Research Project N N516 415338 financed by the Polish Ministry of Science and Higher Education.

References

1. Härdle, W.K., Müller, M., Sperlich, S., Werwatz, A.: Nonparametric and Semiparametric Models. Springer (2004)
2. Dudek, G.: Short-term Load Forecasting Based on Kernel Conditional Density Estimation. Przegląd Elektrotechniczny 86(8), 164–167 (2010)
3. Dudek, G.: Optimization of the Kernel Regression Model to Short-term Load Forecasting. Przegląd Elektrotechniczny 87(9a), 222–225 (2011) (in polish)
4. Dudek, G.: Tournament Searching Method to Feature Selection Problem. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010. LNCS (LNAI), vol. 6114, pp. 437–444. Springer, Heidelberg (2010)
5. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 4th edn. Elsevier Academic Press (2009)