Support Vector Regression based on Grid Search method of Hyperparameters for Load Forecasting

Tran Thanh Ngoc, Le Van Dai, Chau Minh Thuyen

Faculty of Electrical Engineering Technology
Industrial University of Ho Chi Minh City
12 Nguyen Van Bao, Ward 4, Go Vap District, Ho Chi Minh City, Vietnam
tranthanhngoc@iuh.edu.vn, levandai@iuh.edu.vn, chauminhthuyen@iuh.edu.vn

Abstract: Support Vector Regression is becoming one of the most attractive models for load forecasting, in recent years. The performance of Support Vector Regression deeply depends on its hyperparameters, such as, Kernel function, Kernel function parameters and a penalty factor. This paper proposes a methodology for the Grid Search hyperparameters of the Support Vector Regression model. In the training process, the optimal hyperparameters will specify conditions that satisfy requirements for minimizing evaluation indexes of Root Mean Square Error, Mean Absolute Percentage Error, Symmetric Mean Absolute Percentage Error and Mean Absolute Error. In the testing process, the optimal models will be used to evaluate the obtained results along with all other ones. It is indicated that the evaluation indexes of these optimal models are close to the minimum values of all models. Load demand data of Tasmania State, Australia, and Ho Chi Minh City, Vietnam were utilized to verify the accuracy and reliability of the Grid Search methodology.

Keywords: Load forecasting; Grid Search; Support Vector Regression; evaluation index

1 Introduction

Electrical load forecasting is an important element of any electrical power system, including, generation, transmission, distribution and the retail sale of electricity. According to the period of prediction time, load forecasting can be divided into four categories: Very Short Term, Short Term, Medium Term, and Long Term [1, 2]. In recent years, Support Vector Regression (SVR) has been becoming an attractive tool for time series forecasting, especially for load forecasting [3-13]. Generally, SVR shows better generalization performance with the rule of Structural Risk Minimization in comparison with other learning methods such as Neural Networks that are based on Empirical Risk Minimization [3-5, 12]. However, the performance of SVR strongly depends on its hyperparameters. The

more different the hyperparameters, the more different the output values of the SVR model. In this regard, the hyperparameters chosen for an SVR model is essentially important to get reliable forecasting results [14-16]. A variety of methods, such as Genetic Algorithms or Grid Search, have been considered to optimize the hyperparameters. Grid Search is a tuning technique that is performed to obtain the optimum values of hyperparameters by going through all combinations of hyperparameters in the search range. To evaluate the performance of hyperparameters in case of load forecasting, different evaluation indexes can be used, such as Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Absolute Error (MAE), etc. According to recent studies, most Grid Search methods have only focused on the parameters of the SVR model, such as penalty coefficient **C**, kernel function **K**, and kernel parameter γ [17-22]. In addition, the criterion to evaluate the performance of the model is mostly based on one unique evaluation index, which is usually either MAPE or MSE [17, 19-22]. In this study, the SVR Grid Search methodology is proposed with the consideration of the hyperparameters that not only determine the SVR network structure, like conventional Grid Search methods, but also represent the characteristics of input data. The first proposed hyperparameter is the number of lag observation of data used as input of the model, and the second hyperparameter is the differencing order of data. In addition, the study investigates simultaneously four most common evaluation indexes, including RMSE, MAPE, SMAPE, and MAE, to evaluate the performance of the SVR model during the training and testing processes. In the training process, optimal hyperparameters are obtained to meet the minimum requirement of evaluation indexes. In the testing process, these optimal models will be compared to all other ones to evaluate the Grid Search methodology. Load demand data of Tasmania State, Australia, and Ho Chi Minh City, Vietnam, will be analyzed through the training and testing processes to verify the accuracy and the reliability of the Grid Search methodology. The experiments were implemented using the scikit-learn library in the Python environment with Google Colab, a free GPU on the cloud for running large-scale machine learning projects.

This paper is organized as follows, in Section 2, a brief introduction to an SVR model along with proposing the SVR Grid Search methodology is presented. In Section 3, experiments and an analysis of the obtained results are discussed. The conclusions are shared in Section 4.

2   Research Method

2.1   Supporting Vector Regression

Considering a set of data $\{x_i, y_i\}$, i =1, …, N; where $x_i \in R^n$ is the input vector with n dimension, $y_i \in R$ is the target value, and N is the total number of data patterns. The basic concept of the SVR is to map nonlinearly the original input data x into a higher dimensional feature space R^h by using a non-linear mapping function $\varphi() : R^n \rightarrow R^h$. Hence, the SVR function is formulated as follows [3-7, 24, 33]:

$$f(x) = \omega^T \varphi(x) + b \tag{1}$$

where ω represents the weight vector and b is a bias term. The coefficients ω and b are estimated by minimizing the regularized risk function R as shown in (2):

$$R = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N} L_\varepsilon(y_i - f(x_i), x_i) \tag{2}$$

where C is the regularized constant determining the trade-off between the regularization term (the first term) and the empirical error (the second term). The term ε defines the ε zone as shown in Figure 1 [4, 7], and L_ε is called the ε insensitive loss function as follows:

$$L_\varepsilon(y - f(x), x) = \begin{cases} 0, |y - f(x)| \le \varepsilon \\ |y - f(x)| - \varepsilon, otherwise \end{cases} \tag{3}$$

Figure 1
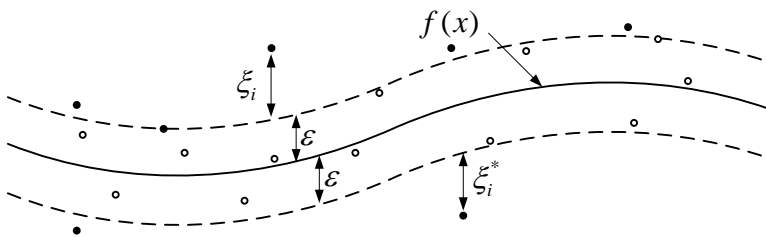Definition of ε, ξ_i, ξ_i^* of SVR

ξ_i, ξ_i^* are slack variables that quantify how far data is above or below the ε tube. As can be seen from Figure 1, the training data points that lie outside of the ε insensitive tube can be obtained by:

$$|y - f(x)| - \varepsilon = \xi, \text{ points above the tube}$$

$$|y - f(x)| - \varepsilon = \xi^*, \text{ points below the tube}$$

$$(4)$$

By substituting the equations (4) and (3) for (2), the equation (2) can be expressed by (5) and subject to constraints expressed in (6):

$$R = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{5}$$

$$y_i - (\omega^T \varphi(x_i) + b) \leq \varepsilon + \xi_i$$

$$(\omega^T \varphi(x_i) + b) - y_i \leq \varepsilon + \xi_i^* \tag{6}$$

$$\xi_i, \xi_i^* \geq 0; i = 1, 2, ..., N$$

The parameters of the f(x) can be found by using the Lagrange function as shown in (7), where α_i^*, α_i are Lagrange multipliers and K(x_i, x) are Kernel function, defined as the dot product between $\varphi(x_i)^T$ and $\varphi(x)$:

$$f(x) = \omega^T \varphi(x) + b = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)K(x_i, x) + b \tag{7}$$

2.2   Hyperparameters Tuning of Supporting Vector Regression

The selection of several hyperparameters of the SVR model that can be listed as a type of Kernel function **K**, parameters of Kernel function **γ**, as well as the regularized constant **C**, is important to the prediction accuracy and have been considered in many recent works of literature [21-24]. In order to improve the accuracy of the SVR model, some hyperparameters of the characteristic of input data such as a number of lag observations that are used as input data, and differencing order of input data are proposed to be in combination with the above-mentioned hyperparameters [25]. Hence, we focus on the following hyperparameters in this section:

Number of input data N:

Usually, in regression and time series forecasting problems, a large amount of data is stored in the form of time series: stock indices, weather measurements, electricity load, etc. Time series data is a set of observations on the values that a variable takes at different times according to a uniform time-frequency and can be defined by (8), where N is the number of observation values:

$$x(t) = \left\{ x(t_1), x(t_2), ..., x(t_N) \right\} \tag{8}$$

The input variables of the SVR model are named by the lag time series values $x(t_1), x(t_2), ..., x(t_N)$ and the predicted output values are referred to the next values $x(t_{N+1})$. The number of input N is pre-specified by the available data that represents the number of lag observations used as input as shown in Table 1. Because of the seasonality of load times series, the number of input N can be chosen as the multiples of the period of time series. For instance, with the period of half-hourly time series of 48, the period of hourly time series is 24, …, and so on.

Table 1
The input and target values for times series

Input	Target
$x(t_1), x(t_2), ..., x(t_N)$	$x(t_{N+1})$
$x(t_2), x(t_3), ..., x(t_{N+1})$	$x(t_{N+2})$
…	…
$x(t_h), x(t_{h+1}), ..., x(t_{N+h-1})$	$x(t_{N+h})$

Data differencing order: *d*

Differencing is a method of transforming a time series dataset that can be used to remove the series dependence on time. Differencing is performed by subtracting the previous observation from the current observation as shown by (9), where d=0: no differencing, d=1: first differencing, and d>1: seasonal differencing.

$$dy(t) = y(t) - y(t - d) \qquad (9)$$

Kernel functions and their parameters

The main function of the Kernel functions is to take low dimensional input space and transform it into a higher-dimensional space. Some basic Kernel functions are linear, Radial Basis Function (RBF), and sigmoid. The mathematical formulas are presented as follows:

Linear:

$$K(x, y) = x^T y \qquad (10)$$

RBF:

$$K(x, y) = e^{-\gamma \|x-y\|^2} \qquad (11)$$

Sigmoid:

$$K(x, y) = \tanh(\gamma x^T y + r) \qquad (12)$$

where x and y are input vectors, $r \geq 0$ is a free parameter, $\gamma > 0$ is the parameters of Kernel function.

Regularized constant C

C is the penalty parameter, which represents the error term between forecast values and observation values. C specifies the trade-off between errors on training data set and margin maximization. A smaller C will lead to more errors and usually produce a larger margin. When C increases to infinity, SVR becomes a hard-margin.

2.3   Methodology of SVR Model

According to Section 2.2, the hyperparameters of the SVR model considered in this paper consist of the following ones: number of input data **N**, data differencing order **d**, Kernel functions **K**, a typical Kernel function parameter γ, and Regularized constant **C**. Based on the SVR structure and these hyperparameters, the methodology of the SVR model is proposed in Figure 2.
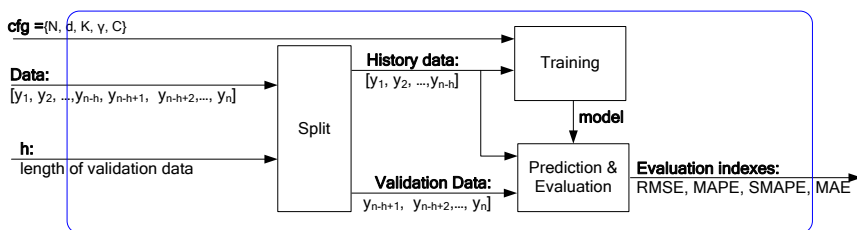

Figure 2

The methodology of the SVR model

The input of SVR model in the paper consists of the hyperparameter **cfg**={N, d, K, C} that was predefined at certain values as mentioned in section 2.2, the **Data,** and the length (dimension) of the validation data **h**. The function of the SVR model is calculating evaluation indexes at the output as shown in Figure 2 which following three steps:

Step 1: The original data ($[y_1, y_2, …, y_n]$) are split into history data ($y_1, y_2, …,y_{n-h}$) and validation data ($y_{n-h+1}, …, y_n$) using split function. The validation and history data have lengths of **h** and **n-h**, respectively. The history data used for the training process to build the SVR model, the validation data used for calculating evaluation indexes.

Step 2: In the training stage, the SVR model is obtained after the training process by using history data and certain values of hyperparameter of N, d, K, $γ$, C.

Step 3: In the prediction and evaluation stage, prediction values are obtained based on the SVR model in step 2, and evaluation indexes (RMSE, MAPE, SMAPE, and MAE) between the prediction and validation values are calculated as follows equations [19-22, 27-32]:

$$RMSE = \sqrt{\frac{1}{h}\sum_{i=1}^{h}\left|y_{n-h+i} - \hat{y}_i\right|^2}, \; MAPE = \frac{1}{h}\sum_{i=1}^{h}\left|\frac{y_{n-h+i} - \hat{y}_i}{y_{n-h+i}}\right| \times 100$$

$$SMAPE = \frac{1}{h}\sum_{i=1}^{h}\left|\frac{y_{n-h+i} - \hat{y}_i}{y_{n-h+i} + \hat{y}_i}\right| \times 200, \; MAE = \frac{1}{h}\sum_{i=1}^{h}\left|y_{n-h+i} - \hat{y}_i\right|$$

(13)

where: $[y_{n-h+1}, y_{n-h+2}, …, y_n]$ is the validation values (real values), and $[\hat{y}_1, \hat{y}_2, ..., \hat{y}_h]$ is the prediction values.

In the paper, the scikit-learn library in Python was used to build the SVR model [26]. The pseudocode of the training and prediction & evaluation stages is shown in Figure 3.

Pseudocode of training stage		Pseudocode of prediction & evaluation stage	
Input: - History data $[y_1, y_2, …, y_{n-h}]$ - cfg: combination of tuning hyperparameters (N, d, K, γ, C)		Input: - History data: $[y_1, y_2, …, y_{n-h}]$ - Validation data: $[y_{n-h+1}, y_{n-h+2}, …, y_n]$ - model: model from training stage	
1:	Differencing data: *If d > 0:* *History data = History data – History data [-d]*	1:	t = 1 **Repeat**

2:	Transform data into supervised format
	Transform History data into input X_train and output Y_train according the number of input data N
		a:	Difference History data and calculate offset
			If d > 0:
			  History data = History data – History data [-d]
			  Offset = History data [-d]

3:	Define SVR model: *model = svm.SVR(kernel=K, gamma = γ, C=C)*	b:	Obtain the input X_test, the length of X_test is the number of input *i*. *X_test = History data [-N:]*
4:	Training model *model.fit(X_train,Y_train)*	c:	Predict for X_test and add offset $\hat{y}_1 = model.predict(X_{test})$ $\hat{y}_1 = \hat{y}_1 + Offset$
Output: model		d:	Add actual observation to history for the next prediction value *History data : [y_1, y_2, …, y_{n-h,} y_{n-h+1}]* *t = t+1* ***Until t = h***
(a)		*2:*	***Calculate evaluation indexes*** Calculate evaluation indexes between the validation values $[y_{n-h+1}, y_{n-h+2}, …, y_n]$ and the prediction values $[\hat{y}_1, \hat{y}_2, ..., \hat{y}_h]$
		Output: RMSE, MAPE, SMAPE, MAE	
		(b)	

Figure 3

The SVR pseudocode: training stage (a); prediction & evaluation stage

2.4   SVR Grid Search Methodology

Based on the hyperparameters tuning and the SVR model as earlier described, the methodology of the SVR Grid Search is proposed as shown in Figure 4. Each hyperparameters take a range of values, such as: $N = \{N_{min}, \ldots, N_{max}\}, \ldots, C = \{C_{min}, \ldots, C_{max}\}$. The combination for certain values of hyperparameter is $\mathbf{cfg_i} = \{N_i, d_i, K_i, \gamma_i, C_i\}$ and the total combination of hyperparameters is $\mathbf{CFG} = \{\mathbf{cfg_i}\}$. The training and the testing processes have the same combination of tuning hyperparameters $\mathbf{CFG}$ and the same length of validation data $\mathbf{h}$. In the training and the testing processes, each component hyperparameter $\mathbf{cfg_i}$ of CFG is inputted into the SVR model (Section 2.3) and the evaluation indexes are calculated at the output layer. The $\mathbf{Min}$ stage in Figure 4 takes a role of selecting optimal hyperparameters ($\mathbf{cfg_{RMSE}}$, $\mathbf{cfg_{MAPE}}$, $\mathbf{cfg_{SMAPE}}$, $\mathbf{cfg_{MAE}}$) that satisfies a minimum of the evaluation indexes ($RMSE_{min}$, $MAPE_{min}$, $SMAPE_{min}$, and MAE_{min}) correspondingly. In the Evaluating stage, these optimal hyperparameter models are compared to all other ones according to their evaluation indexes that were obtained in the testing process.
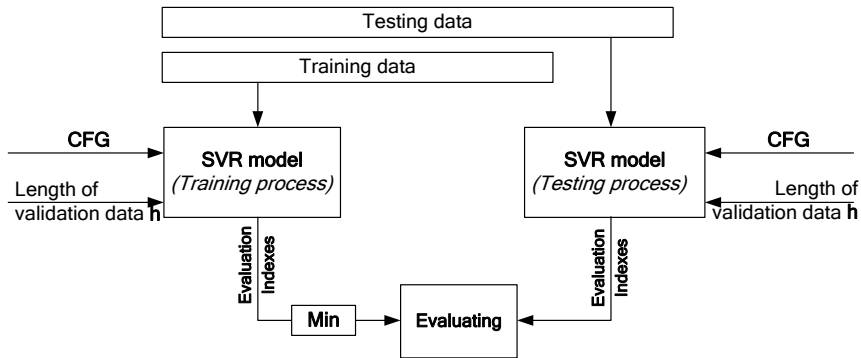

Figure 4

The SVR Grid Search methodology

3   Experimental Results and Analysis

3.1   Data Description and Hyperparameters Tuning

In order to enhance the reliability of results, load demand data of Tasmania State, Australia, and Ho Chi Minh City, Vietnam were studied in our experiments. Tasmania State's load demand dataset provides peak daily electricity demand in MW from 2013-03-10 to 2014-05-31, and so does Ho Chi Minh city load demand dataset - from 2017-10-09 to 2018-12-30. The characteristics of these load demand data, are presented in Table 2 and Figure 5.

Table 2

Characteristics of load demand data

Describe	Tasmania state	Ho Chi Minh City
mean	1304.67	3452.78
std	136.97	475.62
min	987.26	1594.10
max	1650.34	4138.50
count	448	448

Table 3

The values of tuning hyperparameters

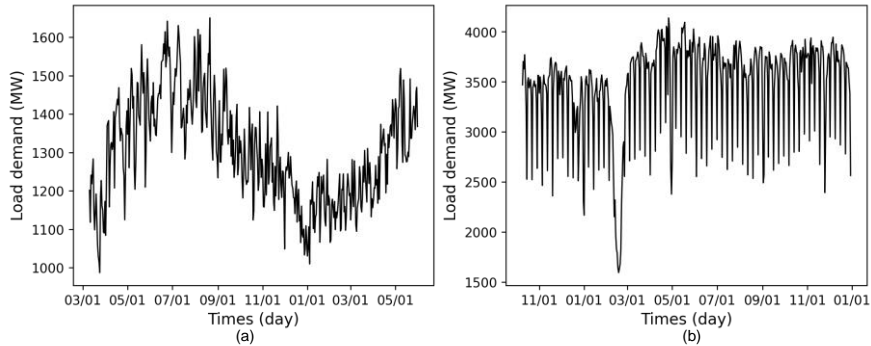| Items	Values
Number of input data **N**	7, 28
Data differencing order **d**	0, 1, 7
Kernel functions **K**	linear, RBF, sigmoid
Kernel function parameter γ	0.01, 0.1, 1, 10, 100, 1000
Regularized constant **C**	0.01, 0.1, 1, 10
Number of combination **CFG**	432

Figure 5

Load demand data: (a) Tasmania state, (b) Ho Chi Minh city

The setup of the value ranges of the hyperparameters tuning for Tasmania state and Ho Chi Minh city load demand data is listed in Table 3. Because of weekly seasonality data, there is a numeric value of 7 assigned to number of input data **N** and the value of differencing order **d**. With N=7, the SVR model used 7 days (one week) before to predict the data of one day ahead; with N=28, the SVR model used 28 days (one typical month) before to predict the data of one day ahead. Combining all tuned hyperparameters gives 432 cases corresponding to 432 possible models of SVR. The values of the length of validation data **h** is 28 (four weeks).

T. N. Tran *et al.*
Support Vector Regression based on Grid Search method of
Hyperparameters for Load Forecasting

3.2 Experimental Results

Table 4 shows the results of training and testing processes using Tasmania state load demand data. For the training process, the optimal model was selected according to the minimum values of evaluation indexes of RMSE, MAPE, SMAPE, and MAE. Obviously, we have the same optimal models in the case of RMSE and MAE, as well as in the case of MAPE and SMAPE. For the testing process, the column 'Optimal' shows evaluation indexes for the optimal model which was obtained from the training process, and the columns 'Min', 'Average' and 'Max' - the min, the average, and the max values for all possible models that can be generated by the combination of tuned hyperparameters **CFG** (432 models). Analyzing Ho Chi Minh City load demand data also gave similar results shown in Table 5.

Table 4

The results of training and testing process in case of Tasmania state load demand data

Evaluation indexes	Optimal models of training process	Evaluation of testing process			
		Optimal	Min	Average	Max
RMSE (MW)	[7, 0, 'linear', 0.01, 10]	74.66	69.75	102.74	386.62
MAPE (%)	[28, 1, 'rbf', 0.01, 0.01]	4.33	4.03	6.05	24.32
SMAPE (%)	[28, 1, 'rbf', 0.01, 0.01]	4.33	4.02	6.14	26.02
MAE (MW)	[7, 0, 'linear', 0.01, 10]	60.87	55.72	81.99	334.50

Table 5

The results of training and testing process in case of Ho Chi Minh city load demand data

Evaluation indexes	Optimal models of training process	Evaluation of testing process			
		Optimal	Min	Average	Max
RMSE (MW)	[28, 7, 'linear', 0.01, 10]	115.25	111.81	283.59	632.74
MAPE (%)	[28, 7, 'linear', 0.01, 10]	2.47	2.43	6.67	18.72
SMAPE (%)	[28, 7, 'linear', 0.01, 10]	2.44	2.41	6.54	19.74
MAE (MW)	[28, 7, 'linear', 0.01, 10]	89.81	86.99	204.83	515.36

Figure 6 indicates the distribution of difference evaluation indexes for the testing process toward Tasmania State load demand data. Figure 6a presents the box plot for the RMSE component with the first column for the distribution of all possible models and the second column for the optimal model. The same distributed data are plotted in Figures 6b, 6c, and 6d. In the same manner, results in the case of Ho Chi Minh City load demand data are shown in Figure 7.
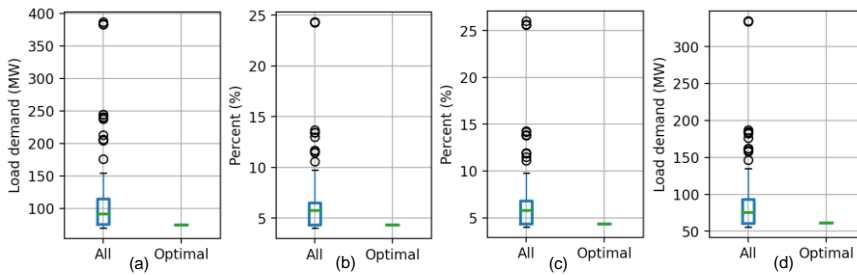
Figure 6

The box plot of evaluation indexes for testing process in case of Tasmania state load demand data:
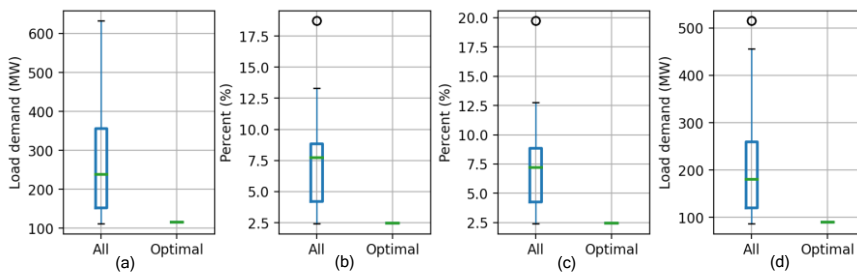(a) RMSE, (b) MAPE, (c) SMAPE, (d) MAE

Figure 7

The box plot of evaluation index for testing process in case of Ho Chi Minh city load demand data:
(a) RMSE, (b) MAPE, (c) SMAPE, (d) MAE

Figure 8 shows the prediction series and the validation series for Tasmania State load demand data. Figures 8a, 8b, 8c, and 8d give the prediction series of the optimal, minimum, and maximum models and the validation series for RMSE, MAPE, SMAPE, and MAE in the testing process, respectively. With regard to Ho Chi Minh City load demand data, similar results were also obtained as shown in Figure 9.
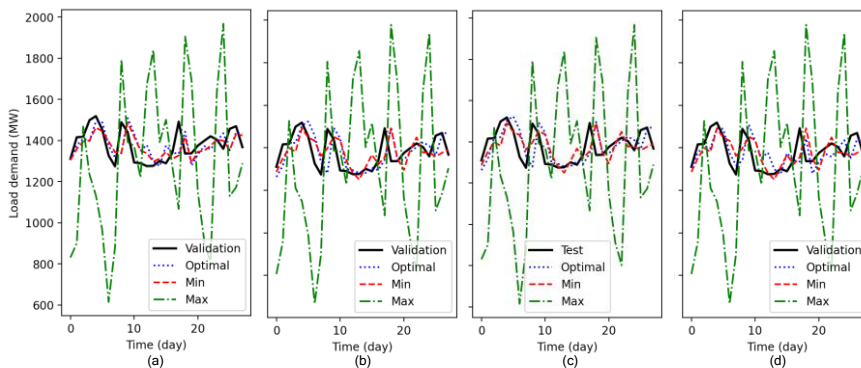

Figure 8

The prediction and validation series for testing process in case of Tasmania state load demand data:
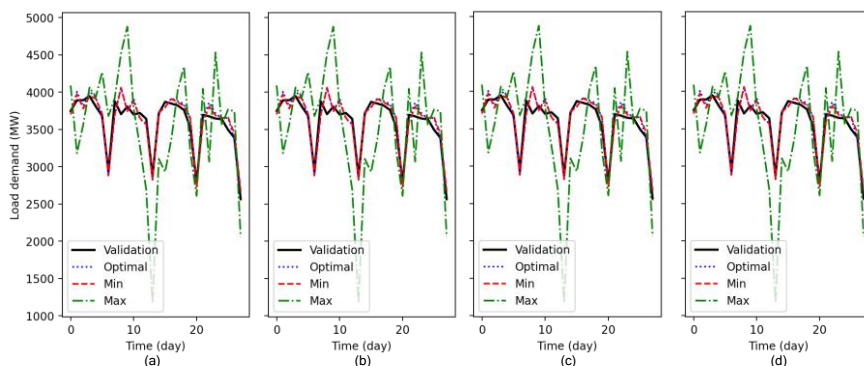(a) RMSE, (b) MAPE, (c) SMAPE, (d) MAE

Figure 9

The prediction and validation series for testing process in case of Ho Chi Minh city load demand data:
(a) RMSE, (b) MAPE, (c) SMAPE, (d) MAE

3.3   Evaluation and Discussion

As described above, the optimal models in the training process were determined by minimizing evaluation indexes such as RMSE, MAPE, SMAPE, and MAE. The analysis of the results listed in Table 4 and Table 5 shows the existence of the optimal model that satisfied the minimum criteria for evaluation indexes of RMSE, MAPE, SMAPE, and MAE. In the case of Tasmania state load demand data, there existed a model that satisfied all RMSE, MAE, and the other ones for MAPE and SMAPE. In the case of Ho Chi Minh City load demand data, there was a unique model that satisfied all RMSE, MAPE, SMAPE, and MAE.

The analysis of the data differencing order **d** is also shown in Tables 4, 5. The optimal value of **d** was 0 for RMSE and MAE, and the value of d was 1 for MAPE and SMAPE in the case of Tasmania. In the case of HCM, the optimal value of **d** was 7 for all evaluation indexes. Hence, transforming input data by representing the **d** hyperparameters can help to get better results than using original data (d=0).

An analysis of the number of input **N** hyperparameter can be extracted from Tables 4, 5. In the case of Tasmania, the optimal value of **N** was 7 for RMSE and MAE indexes and 28 for MAPE and SMAPE indexes. In the case of HCM, the optimal value of **N** was 28 for all evaluation indexes of RMSE, MAPE, SMAPE, and MAE. The results show that with extending the hyperparameters by using the input **N** hyperparameter, we achieved a better optimal model than without using the N hyperparameter (in case that the number of input data may be fixed at 7, 28, …)

It is worth noting, that the optimal model obtained in the training process does not guarantee the best results in the testing process. Let us analyze Table 4 in the case of Tasmania state load demand data. When using the optimal model in the testing process, evaluation indexes of RMSE, MAPE, SMAPE, and MAE were 74.66

MW, 4.33%, 4.33%, and 60.87 MW, respectively. Meanwhile, there were other models that gave better results in the testing process with the minimum values of 69.75 MW, 4.03%, 4.02%, and 55.72 MW for RMSE, MAPE, SMAPE, and MAE, respectively. However, compared to the average values of all models with the values of 102.74 MW, 6.05%, 6.14%, and 81.99 MW and to the maximum values of all models with the values of 386.62 MW, 24.32%, 26.02%, and 334.50 MW respectively, evaluation indexes of the optimal model was too small. In addition, analyzing the boxplot of the evaluation indexes shown in Figure 6 makes it clear that the evaluation indexes of the optimal models were close to the minimum value of all other models. Moreover, the prediction values of the optimal model shown in Figure 8 were very consistent with the validation values. Similar results were also obtained in the case of Ho Chi Minh City load demand data. These results clearly indicated that the optimal model received during training by applying the SVR Grid Search methodology can give good values in the testing process. Moreover, the SVR Grid Search methodology is thus suitable for most of the basic evaluation indexes such as RMSE, MAPE, SMAPE, and MAE.

Conclusions

An efficient SVR Grid Search methodology, based on the SVR model and their hyperparameters, was proposed, with reasonable analysis of Tasmania state and Ho Chi Minh City load demand data. In the training process, minimum evaluation indexes of RMSE, MAPE, SMAPE, and MAE were applied to specify the optimal model. In the testing process, evaluation indexes were used to compare the optimal model with all other ones. Both Tasmania State and Ho Chi Minh City load demands were used to verify the performance of the SVR Grid Search methodology. The results indicated the existence of an optimal model that satisfied the minimum requirement for evaluation indexes. Analyzing the obtained results in the testing process, showed that the evaluation indexes of these optimal models gave good results, close to the minimum and much smaller than the average of all models. The positive results obtained in this study suggest an improved and effective way to apply SVR in load forecasting.

References

[1]   Raza MQ, Khosravi A. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. Renew Sustain Energy Rev. 2015;50:1352-1372

[2]   Din GMU, Mauthe AU, Marnerides AK. Appliance-level Short-Term Load Forecasting using Deep Neural Networks. 2018 Int Conf Comput Netw Commun ICNC 2018. 2018:53-57

[3]   Tian L, Noore A. A novel approach for short-term load forecasting using support vector machines. Int J Neural Syst. 2004;14(5):329-335

[4]   Bozic M, Stojanovic M. Application of SVM methods for mid-term load forecasting. Serbian J Electr Eng. 2011;8(1):73-83

[5]   Hong WC. Electric load forecasting by support vector model. Appl Math Model. 2009;33(5):2444-2454

[6]   De Cosmis S, De Leone R, Kropat E, Meyer-Nieberg S, Pickl S. Electric load forecasting using support vector machines for robust regression. Simul Ser. 2013;45(5):72-79

[7]   Türkay BE, Demren D. Electrical load forecasting using support vector machines: A case study. Int Rev Electr Eng. 2011;6(5):2411-2418

[8]   Sadli M, Fajriana, Fuadi W, Ermatita, Pahendra I. Electrical peak load forecasting using long short term memory and support vector machine. IOP Conf Ser Mater Sci Eng. 2020;725(1)

[9]   Li X, Gong D, Li L, Sun C. Next day load forecasting using SVM. Lect Notes Comput Sci. 2005;3498(III):634-639

[10] Ostojin S, Kulic F, Svenda G, Bibic R. Short-Term Electrical Load Forecasting Using Support Vector Machines. Comput Simul Mod Sci. 2008;I:138-142

[11] Yang M, Li W, Zhang H, Wang H. Parameters optimization improvement of SVM on load forecasting. Proc - 2016 8th Int Conf Intell Human-Machine Syst Cybern IHMSC 2016

[12] Qiang S, Pu Y. Short-term power load forecasting based on support vector machine and particle swarm optimization. J Algorithms Comput Technol. 2018;13:1-8

[13] Matijaš M, Vukićević M, Krajcar S. Supplier short term load forecasting using support vector regression and exogenous input. J Electr Eng. 2011;62(5):280-285

[14] Lin S, Zhang S, Qiao J, Liu H, Yu G. A parameter choosing method of SVR for time series prediction. Proc 9th Int Conf Young Comput Sci ICYCS 2008. 2008:130-135

[15] Sano N, Higashinaka K, Suzuki T. Efficient parameter selection for support vector regression using orthogonal array. Conf Proc - IEEE Int Conf Syst Man Cybern. 2014;2014-Janua(January):2256-2261

[16] Hsia JY, Lin CJ. Parameter Selection for Linear Support Vector Regression. IEEE Trans Neural Networks Learn Syst. 2020:1-6

[17] Gonz A, Skarmeta AF, Miˇ S. Machine Learning based Electric Load Forecasting for Short and Long-term Period. 2016-2018 IEEE 4th World Forum Internet Things. 2016:511-516

[18] Syarif I, Prugel-Bennett A, Wills G. SVM parameter optimization using grid search and genetic algorithm to improve classification performance.

Telkomnika (Telecommunication Comput Electron Control. 2016;14(4):1502-1509

[19]   Zahid M, Ahmed F, Javaid N, et al. Electricity price and load forecasting using enhanced convolutional neural network and enhanced support vector regression in smart grids. Electron. 2019;8(2):1-32

[20]   Smets K, Verdonk B, Jordaan EM. Evaluation of performance measures for SVR hyperparameter selection. IEEE Int Conf Neural Networks - Conf Proc. 2007:637-642

[21]   Huang Q, Mao J, Liu Y. An improved grid search algorithm of SVR parameters optimization. Int Conf Commun Technol Proceedings, ICCT. 2012;(2):1022-1026

[22]   Ma X, Zhang Y, Wang Y. Performance evaluation of kernel functions based on grid search for support vector regression. Proc 2015 7th IEEE Int Conf Cybern Intell Syst CIS 2015 Robot Autom Mechatronics, RAM 2015. 2015:283-288

[23]   Lahiri SK, Ghanta KC. The support vector regression with the parameter tuning assisted by a differential evolution technique: Study of the critical velocity of a slurry flow in a pipeline. Chem Ind Chem Eng Q. 2008;14(3):191-203

[24]   Zhang H, Chen L, Qu Y, Zhao G, Guo Z. Support vector regression based on grid-search method for short-term wind power forecasting. J Appl Math. 2014;2014

[25]   Jason Brownlee. Deep Learning With Python. Machine Learning Mastery Pty. Ltd.: Vermont Victoria, Australia. 2019

[26]   Andreas C. Müller and Sarah Guido. Introduction to Machine Learning with Python. In O'Reilly Media, Inc. 2016

[27]   Grmanová G, Laurinec P, Rozinajová V, et al. Incremental ensemble learning for electricity load forecasting. Acta Polytech Hungarica. 2016;13(2):97-117

[28]   Ngoc TT, Dai L Van. Grid search of exponential smoothing method: A case study of Ho Chi Minh City load demand. Indones J Electr Eng Comput Sci. 2020;19(3):1121-1130

[29]   Hyndman RJ, Koehler AB, Ord JK, Snyder RD. Forecasting with Exponential Smoothing: The State Space Approach, 2008

[30]   R. Precup, T. Teban, T. E. A. d. Oliveira and E. M. Petriu. Evolving fuzzy models for myoelectric-based control of a prosthetic hand. 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Vancouver, BC. 2016; pp. 72-77

[31] Hedrea, E. L., Precup, R. E., & Bojan-Dragos, C. A. Results on tensor product-based model transformation of magnetic levitation systems. Acta Polytechnica Hungarica. 2019; 16(9): 93-111

[32] Juang, C. F., Lin, Y. Y., & Huang, R. B. Dynamic system modeling using a recurrent interval-valued fuzzy neural network and its hardware implementation. Fuzzy Sets and Systems. 2011; 179(1): 83-99

[33] Cheng, W. Y., & Juang, C. F. A fuzzy model with online incremental SVM and margin-selective gradient descent learning for classification problems. IEEE Transactions on Fuzzy Systems. 2014; 22(2): 324-337