https://doi.org/10.3311/PPee.20887

Periodica Polytechnica Electrical Engineering and Computer Science, 67(1), pp. 51–60, 2023

A New Grid Search Algorithm Based on Median Values for SVR Model in Case of Load Forecasting

Thanh Ngoc Tran[1*]

[1] Department of Electrical Power Supply, Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, 12 Nguyen Van Bao, Ward 4, Go Vap District, 70000 Ho Chi Minh City, Vietnam
[*] Corresponding author, e-mail: tranthanhngoc@iuh.edu.vn

Received: 26 July 2022, Accepted: 20 September 2022, Published online: 20 October 2022

Abstract

In this paper, a Median Grid Search algorithm based on the median values is proposed to develop a more accurate algorithm used to investigate the optimal hyperparameter of the SVR model for load forecasting. In addition, the methodology to benchmark the proposed Grid Search and the conventional Grid Search is built and sufficiently utilized. The data gathered from the South Australia state, Australia and Ho Chi Minh City load demands are used in experiments. Experimental results demonstrate that the proposed algorithm outperforms the conventional algorithm.

Keywords

SVR, grid search, load forecasting, boxplot

1 Introduction

Load forecasting plays a major role in the generation, transmission, distribution and the retail sale of electric power systems [1–4]. In recent years, Support Vector Regression (SVR) has become one of the most attractive research tools for load forecasting [5–7]. However, the accuracy of the SVR model critically depends on its hyperparameters, such as penalty coefficient C, kernel function K, and kernel parameter γ [8–11]. Therefore, it is essential to find optimal values of these hyperparameters to improve the reliability of the SVR model. Some optimization algorithms have been used to solve this problem, including Random Search, Grid Search, Gradient-based Optimization, Particle Swarm Optimization, Genetic Algorithms, etc. [12–16]. Among these algorithms, Grid Search is a powerful technique which has been developed by different researchers in recent years. The Grid Search algorithm generates evenly spaced values for each hyperparameter being tested, and then uses cross - validation to choose the lowest error rate of each combination [17–22]. However, due to the considerably giant fluctuation around this minimum, the Grid Search algorithm may not give the best values in testing step regardless of the best ones obtained in training.

In this regard, the present study proposes a new Grid Search algorithm based on the Median values instead of the minima to improve the performance of the Grid Search algorithm, so that the error rate of the SVR model will be reduced effectively. The boxplot of error rate is distributed by each hyperparameter to identify the optimal values. The methodology to benchmark the proposed and conventional models is presented. In the training process, the optimal hyperparameters of both Median and conventional Grid Search algorithms are determined. In the testing process, these optimal hyperparameters are evaluated by their error rates.

The paper is organized as follows. Section 2 presents a critical discussion about the Support Vector Regression, the Grid Search algorithm, the Median Grid Search algorithm, as well as the methodology used to evaluate these algorithms; Section 3 reports the experimental results and their analysis. Section 4 (conclusions) summarizes the main points of the research.

2 Research method

2.1 Support vector regression

Given training data $(\boldsymbol{x}_i, y_i) \in R^n \times R$, where $\boldsymbol{x}_i \in R^n$ is the feature input vector; $y_i \in R$ is the target value; $i = 1,\dots, m$ denote the number of training instances; n denote the number of features in each instance. In SVR, x is mapped to $\Phi(x)$ in a higher dimensional feature space presented by the following expression [9, 17, 18].

Cite this article as: Tran, T. N. "A New Grid Search Algorithm Based on Median Values for SVR Model in Case of Load Forecasting", Periodica Polytechnica Electrical Engineering and Computer Science, 67(1), pp. 51–60, 2023. https://doi.org/10.3311/PPee.20887

$$f(x) = <\boldsymbol{\omega},\ \Phi(x)> + \boldsymbol{b} \tag{1}$$

Here < . , . > denotes the dot product, $\boldsymbol{\omega}$ and $\boldsymbol{b}$ are the weight vector and bias. The factors $\boldsymbol{\omega}$ and $\boldsymbol{b}$ are estimated by minimizing the following regularized risk as follow:

$$R = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{m} L(y_i, f(\boldsymbol{x}_i)), \tag{2}$$

where $L(y_i, f(\boldsymbol{x}_i))$ is the linear ε insensitive loss defined by:

$$L(y, f(x)) = |y - f(x)|_\varepsilon = \max\{0, |y - f(x)| - \varepsilon\}. \tag{3}$$

In Eqs. (2) and (3), C is the regularization parameter, ε is the error sensitivity parameter. By substituting the Eq. (3) for Eq. (2), and introducing the positive slack variables ξ_i, ξ_i^* to denote deviations from an ε-zone, the Eq. (2) can be expressed by objective function Eq. (4) and subject to constraints expressed in Eq. (5):

$$R = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*), \tag{4}$$

$$\begin{aligned} & y_i - (\boldsymbol{\omega}^T\varphi(\boldsymbol{x}_i) + \boldsymbol{b}) \le \varepsilon + \xi_i \\ & (\boldsymbol{\omega}^T\varphi(\boldsymbol{x}_i) + \boldsymbol{b}) - y_i \le \varepsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0;\ i = 1, 2, ..., N. \end{aligned} \tag{5}$$

Applying the Lagrange multiplier and the optimally constraints, the Eq. (1) has the following explicit form as Eq. (6), where α_i^*, α_i are Lagrange multipliers and $K(\boldsymbol{x}_i, x)$ are the Kernel function, defined as the dot product between $\varphi(\boldsymbol{x}_i)^T$ and $\varphi(x)$:

$$f(x) = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)K(\boldsymbol{x}_i, x) + \boldsymbol{b}. \tag{6}$$

One of the regular kernel functions is radial basis function (RBF) which most commonly with a Gaussian RBF kernel and have used in this paper:

$$K(x, y) = e^{-\gamma\|x-y\|^2}. \tag{7}$$

A deep learning model in general and SVR in particular can be consisted of two types of parameters. The first one includes parameters of the model learned during the model training, and the second one contains hyperparameters which can be randomly set before starting training instead. Based on the detail analysis of SVR model given above, there exist three hyperparameters that control performance of the SVR model, including the ε parameter characterizing the constraints of $f(x)$; the C parameter standing for the relationship between the regulation term and the empirical error; and the Kernel parameter γ. These hyperparameters strongly affect the accuracy of the SVR model for that the inappropriateness of one hyperparameter may lead to overfitting and underfitting phenomena. Thus, answering the question of how to optimize these hyperparameters within a specified range is the key point in using of SVR model. The obtained optimal hyperparameters are a set of values that yields an optimal model which minimizes a predefined performance error rate. For this purpose, several algorithms have been reported, such as Random Search, Grid Search, Gradient-based Optimization, Particle Swarm Optimization, Genetic Algorithms, etc. Among these algorithms, Grid Search is a powerful technique which has been developed by different researchers and have been considered in the present work.

2.2 The conventional grid search algorithm

The conventional Grid Search algorithm is a searching process through a grid of subsets that were pre-specified by the combinations of different values of the hyperparameters [17–20]. Optimal hyperparameters are those corresponding to the model that produces the smallest error. Let us consider the Grid Search algorithm with three hyperparameters ε, C, γ for the SVR model. Hyperparameters are configured with M values for ε $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_M\}$; N values for hyperparameter C, $\{C_1, C_2, \dots, C_N\}$, and P values for hyperparameter γ, $\{\gamma_1, \gamma_2, \dots, \gamma_P\}$. A combination of these three hyperparameters consisting of number of elements is $M \times N \times P$. As a result, the Grid Search algorithm conducts the searching of the best model based on this combination. Fig. 1 shows the Grid Search algorithm for the SVR model based on the boxplot of the error rate. Firstly, the ε_{opt} is obtained with the lowest values of the error rate in the ε boxplot; then C_{opt}, and γ_{opt} are sequentially chosen to use the C and γ boxplot. Note that any order of ε, C, and γ of the Grid Search algorithm in Fig. 1 all have the unique optimal hyperparameters $\{\varepsilon_{opt}, C_{opt}, \gamma_{opt}\}$ because it is considered based on the lowest values of the error rate.

Cross-validation (CV) is a standard technique of the Grid Search algorithm for the SVR model [15, 17]. For the K-fold CV, the dataset S is divided into K subsets $S_1, \dots, S_K$. For each $i = 1, \dots, K$, an individual model is built by applying the GS algorithm to the training data $S_{\setminus i}$ (the remaining fold of S except those in S_i). The average of the K model evaluations is called as cross-validation error and used to validate the performance of the *GS* algorithm when applied to S.

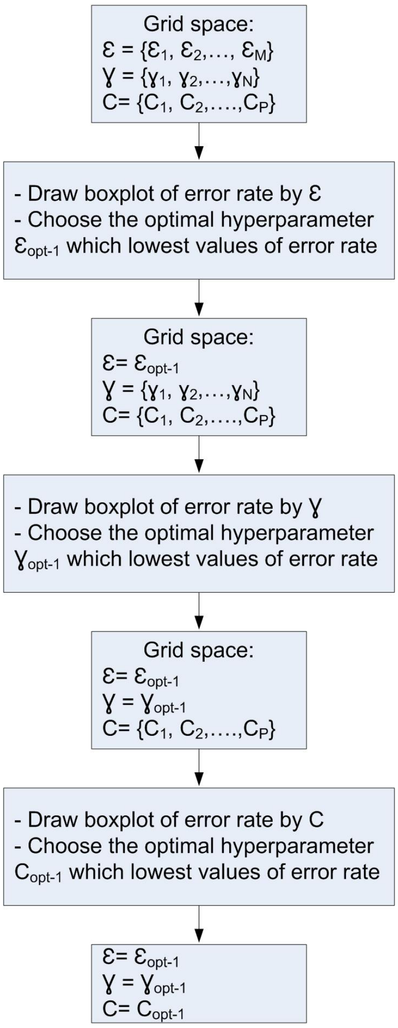


Fig. 1 The Grid Search algorithm

As described above, the Grid Search algorithm uses all possible combinations of the specified hyperparameters, evaluates the performance for each combination and selects the optimal value of the hyperparameters with the minimum error rate. However, the error rate can fluctuate around this minimum value. As a result, the Grid Search algorithm may not give the best values in testing step regardless of the best ones obtained in training. In this regard, the present study proposes a new Grid Search algorithm based on the conventional Grid Search algorithm and the median values instead of the minima in this paper.

2.3 Median grid search algorithm

In this study, we propose the new Grid Search based on the boxplot components. Boxplot is a standardized way for displaying the distribution of data by dividing the dataset into equal fourths, as shown in Fig. 2 [23–26]. The 1st (Q1), the 2nd (Q2) and the 3rd (Q3) quartiles correspond to the 25th, 50th, and 75th percentile of the dataset. The second quartile (Q2) lies in the middle and divides the data into halves, so Q2 is also known as the median.

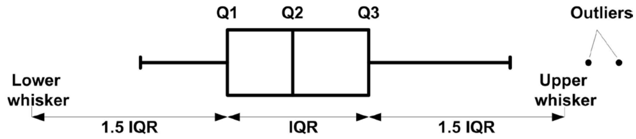


Fig. 2 The boxplot components

The important component in the boxplot is the Interquartile Range (IQR) showing the range between the Q1 and Q3. The upper and lower whiskers are also shown in the Fig. 2 to represent scores outside the IQR. Outliers are defined as data points which are located outside the whiskers of the box plot.

Based on the conventional Grid Search algorithm of the SVR model, this paper proposes the new Grid Search algorithm using the boxplot model. The different between the new and old ones is related to the optimal hyperparameters, which are chosen by the lowest of the median values instead of minima of the error rates. And that is the reason why the new algorithm is called Median Grid Search in this study. The algorithm of the Median Grid Search is shown in Fig. 3. The typical hyperparameters of the SVR model are ε, γ, C, so their combination gives 6 cases as $\{\varepsilon, C, \gamma\}$, $\{\varepsilon, \gamma, C\}$, $\{C, \varepsilon, \gamma\}$, $\{C, \gamma, \varepsilon\}$, $\{\gamma, C, \varepsilon\}$, $\{\gamma, \varepsilon, C\}$ distributed accordingly from Model 1 to Model 6. In the Model 1, the optimal hyperparameters sequentially obtained are $\varepsilon_{opt\text{-}1}$, then the $C_{opt\text{-}1}$, and the last of $\gamma_{opt\text{-}1}$, with the criterion of minimizing the median values in the boxplot. In the same process, the similar optimal hyperparameters of $\varepsilon_{opt\text{-}2}$, $\gamma_{opt\text{-}2}$, $C_{opt\text{-}2}$ and $\gamma_{opt\text{-}6}$, $\varepsilon_{opt\text{-}6}$, $C_{opt\text{-}6}$ are also sequentially obtained for Models 2 and 6, respectively. As a result of choosing the lowest of the median values, so perhaps we have 6 optimal hyperparameters corresponding to 6 models (1 to 6). In the last process, the optimal model of the methodology is obtained by the minimum error rate of 6 models. Note that along with the conventional Grid Search algorithm, the Cross-Validation technique is also applied in the Median Grid Search algorithm.

2.4 Methodology of evaluating the median grid search algorithm

To evaluate the effectiveness of the Median Grid Search model, it is necessary to benchmark the proposed model and the conventional one. In this regard, the methodology consisting of three processes (data processing, training processing, and testing processing) is used as shown in Fig. 4 for the conventional Grid Search (Fig. 4 (a)) and the Median Grid Search algorithm (Fig. 4 (b)).

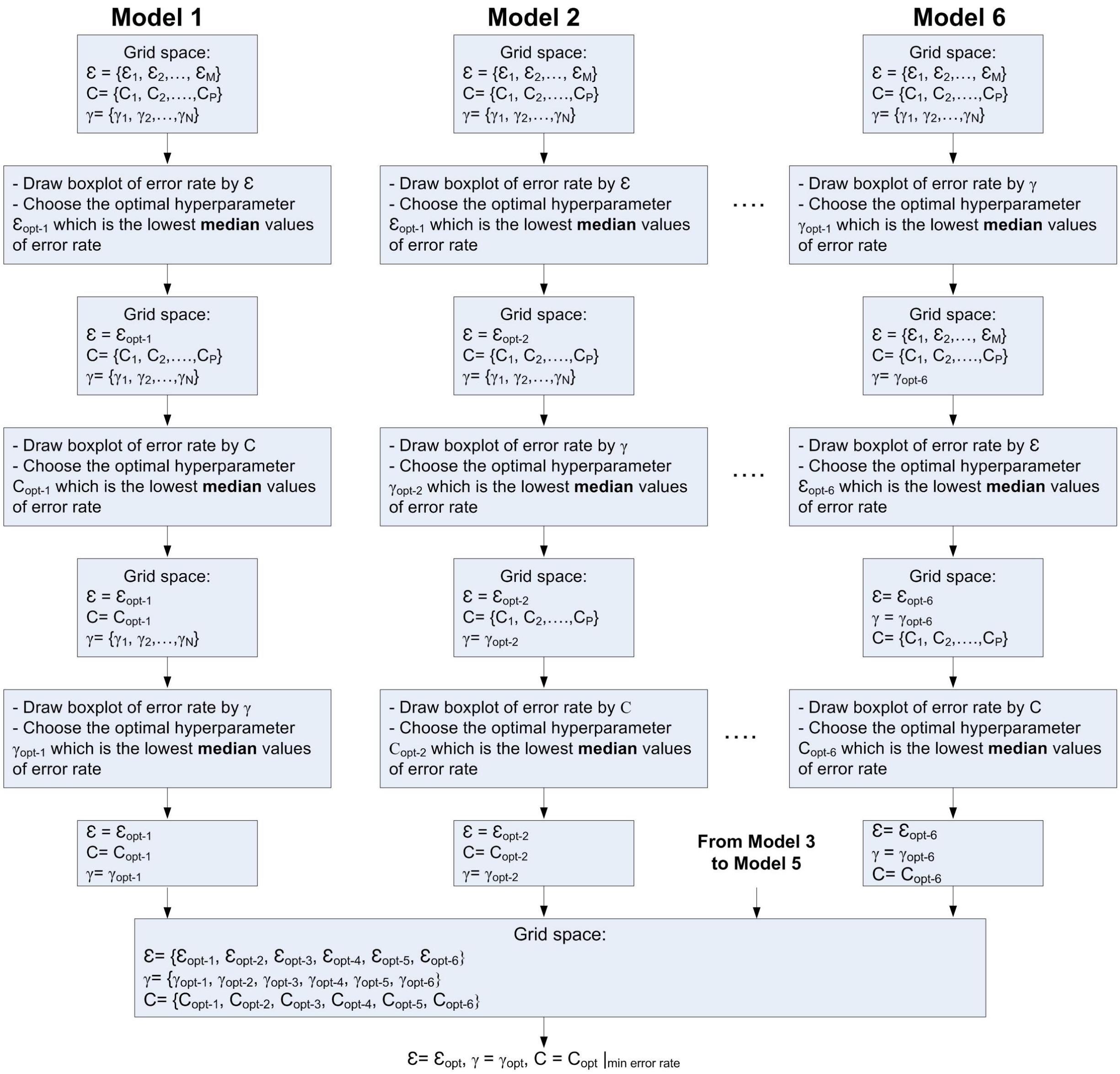


Fig. 3 The Median Grid Search algorithm

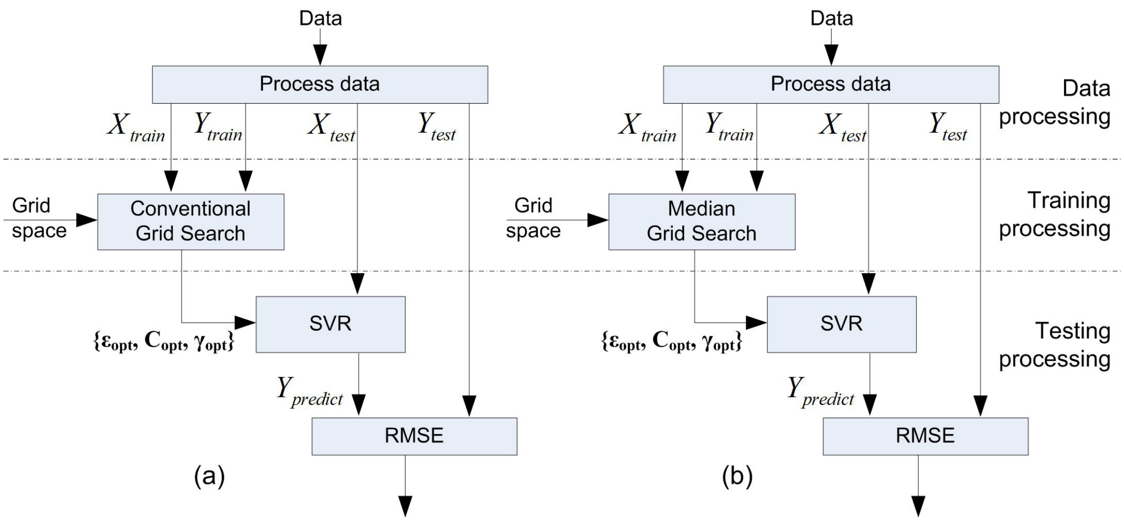


Fig. 4 Methodology for evaluating algorithms: The conventional Grid Search (a); The Median Grid Search (b)

- In the data processing, the data was handled to make two couples of input-target, denoted as (X_{train}, Y_{train}) and (X_{test}, Y_{test}) for training and testing process, respectively.
- In the training processing, the conventional Grid Search and the Median Grid Search algorithms were applied to determine the optimal hyperparameters of the SVR model. Grid space is a total combination of different sets of SVR hyperparameters. In this study, Grid space = $\{\varepsilon_m, C_n, \gamma_p\}$, $m = 1{:}\ M$, $n = 1{:}\ N$, $p = 1{:}\ P$. The output of the training process is the optimal hyperparameter. The cross-validation technique was also implemented in this process to enhance the performance of both Grid Search algorithms.
- In the testing processing, these optimal hyperparameters are used to produce the predicted values Y_{predict}. And the error rate (RMSE) of the SVR model was calculated based on the difference between Y_{predict} and Y_{test}. After that, these RMSE values between the conventional Grid Search and the Median Grid Search will be compared to evaluate the effect of the proposed Median Grid Search algorithm.

The error rate in this study is the RMSE calculated by the following equation [27, 28]:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|^2}\ , \tag{8}$$

where: $[y_1, y_2, \ldots, y_n]$ is the test values, and $[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n]$ is the prediction values.

3 Experimental results

3.1 Experimental settings

To evaluate the proposed Median Grid Search algorithm with the benchmark Grid Search algorithm, this paper considered real half-hour load demand data of South Australia state (SA), Australia, and hourly load demand data of Ho Chi Minh City (HCM). In each dataset, two cases are considered. Firstly, the range of training processing is of two weeks, and the testing range - one week. This case corresponds to K-fold = 2 in the cross-validation technique. Secondly, the range of training processing is taken as three weeks, and the test range - one day, corresponding to K-fold = 3 in the cross-validation technique. Hence, there are total four cases that will be performed experiments with the characteristics listed in Table 1. The experiments were implemented using the Scikit-learn library in the Python environment with Google Colab.

The tuning values of the SVR hyperparameters for the grid space are shown in Table 2 for all cases for #1 to #4. The ranges of hyperparameters of ε, C and γ are determined as [1e-04, 1], [1e+01, 1e+05] and [1e-06, 1e-02], respectively. The step size is obtained by log scale, and the number values of each hyperparameter are 25. Hence, the grid space = $\{\varepsilon_m, C_n, \gamma_p\}$, m = 1:25, n = 1:25, p = 1:25 have 25^3 possible SVR models.

3.2 Experimental results

Fig. 5, Fig. 6, and Fig. 7 illustrate the process step by step in the training process of the Median Grid Search algorithm and the conventional Grid Search. These processes are described as detailed in Fig. 1 and Fig. 3. The typical Model 2 of the Median Grid Search algorithm is used in this analysis. The experimental setting is case #1 (described detailed in Table 1).

Fig. 5 shows the boxplot of the error rate distributed by hyperparameter ε. Each value of $\{\varepsilon_i\}$, i = 1:25 corresponds to the combination of 25 × 25 = 625 values of pair $\{\gamma, C\}$. The Fig. 5 (a) for the Median Grid Search algorithm indicates that the optimal value of the hyperparameter ε is

Table 1 The data characteristic of 4 cases in experiments

Case	Dataset	Train process		Test process		K-fold
		X_{train}	Y_{train}	X_{test}	Y_{test}	
Case #1	HCM	(336, 24)	336	(168, 24)	168	2
Case #2	SA	(672, 48)	672	(336, 48)	336	2
Case #3	HCM	(504, 24)	504	(24, 24)	24	3
Case #4	SA	(1008, 48)	1008	(48, 48)	48	3

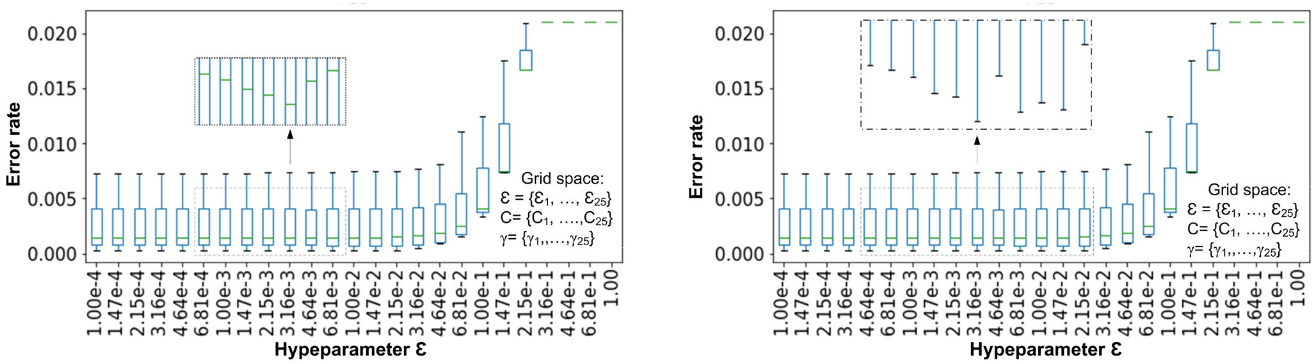
Table 2 The space grid of hyperparameters

Order	1	2	3	4	5	6	7	…	24	25
ε	1.E-4	1.47E-4	2.15E-4	3.16E-4	4.64E-4	6.81E-4	1.00E-3	…	6.81E-1	1.E+0
C	1.E+1	1.47E+1	2.15E+1	3.16E+1	4.64E+1	6.81E+1	1.00E+2	…	6.81E+4	1.E+5
γ	1.E-6	1.47E-6	2.15E-6	3.16E-6	4.64E-6	6.81E-6	1.00E-5	…	6.81E-3	1.E-2

equal to 3.16e-3 corresponding to the minimum value of medians. In the case of the conventional Grid Search algorithm (Fig. 5 (b)), the optimal value of the hyperparameter ε is of 3.16e-3 (the lowest value).

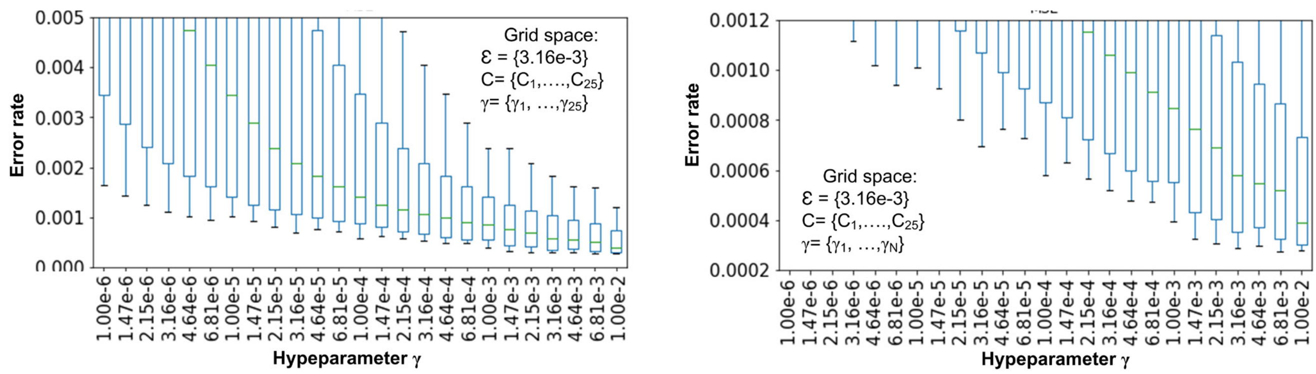
Fig. 6 shows the boxplot of the error rate distributed by hyperparameter γ. Each value of $\{\gamma_i\}$, $i = 1{:}25$ is counted to the combination of $1 \times 25 = 25$ values of the pair $\{\varepsilon = 3.16\text{e-}3, C\}$. Similar to the above hyperparameter ε, the values of $\gamma = 1.00\text{e-}2$ (Fig. 6 (a)) and 6.81e-3 (Fig. 6 (b)) are referred to the median minimum and the lowest value, respectively.

The boxplot of the error rate distributed by hyperparameter C is presented in Fig. 7. Each value of $\{C_i\}$, $i = 1{:}25$ is taken accordingly to definite values of the pair $\{\varepsilon = 3.16\text{e-}3, \gamma = 1.00\text{e-}2\}$ for the Median Grid Search algorithm, and $\{\varepsilon = 3.16\text{e-}3, \gamma = 6.81\text{e-}3\}$ for the conventional Grid Search. The Fig. 7 (a) show that the optimal value of the hyperparameter C is equal to 2.15e+3 corresponding to the minimum value of medians. Meanwhile, the optimal value of the hyperparameter ε of 4.64e+3 corresponds to the lowest value (Fig. 7 (b)).



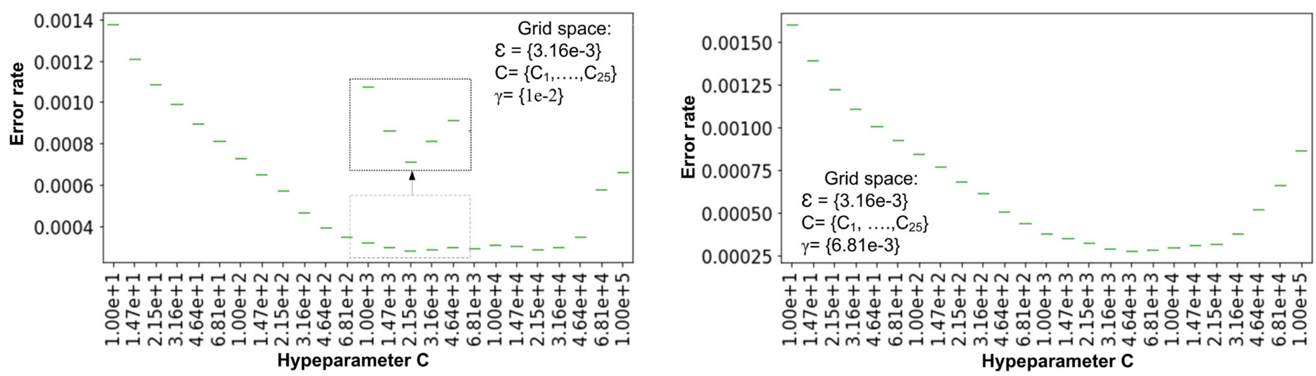
(a) (b)

Fig. 5 The boxplot of error rate by ε: Median Grid Search (a); Conventional Grid Search (b)



(a) (b)

Fig. 6 The boxplot of error rate by γ: Median Grid Search (a); Conventional Grid Search (b)



(a) (b)

Fig. 7 The boxplot of error rate by C: Median Grid Search (a); Conventional Grid Search (b)

After three steps described in the Fig. 5 (a), Fig. 6 (a) and Fig. 7 (a), the optimal hyperparameters of the Model 2 are obtained as {ε = 3.16E-3, γ = 1.00E-2, C = 2.15E+3}. Besides, as seen in the Fig. 5 (b), Fig. 6 (b), Fig. 7 (b), the optimal hyperparameters of the conventional Grid Search denoted as Grid Search is {ε = 3.16E-3, γ = 6.81E-3, C = 4.64E+3}. In the same way, the results for Model 1, 3, 4, 5, and 6 were also obtained. All results are listed in Table 3, noted that the last row contains the values of the error rates. According to the last process of algorithm (Fig. 3), values for the optimal model of the Median Grid Search algorithm are {ε = 3.16E-3, γ = 1.00E-2, C = 2.15E3}, corresponding to the minimum values of the error rates of Model 1 to Model 6 (equal to 2.79E-4).

As the same as the case #1, Table 4 shows the experimental results for the case # 2 with the optimal model of the Median Grid Search algorithm of {ε = 1.00E-2, γ = 6.81E-3, C = 1.00E+4}, and the conventional Grid Search algorithm of {ε = 1.47E-2, γ = 3.16E-3, C = 3.16E+4}.

Similarly, Table 5 shows the experiments results corresponding to the case # 3 with {ε = 6.81E-3, γ = 1.00E-2, C = 1.47E+4} and {ε = 1.00E-2, γ = 1.00E-2, C = 2.15E+4} for the optimal model of the Median and conventional Grid Search algorithms, respectively.

Table 6 shows the experiments results corresponding to the case # 4 with {ε = 1.00E-2, γ = 1.00E-2, C = 6.81E+3} and {ε = 1.00E-2, γ = 1.00E-2, C = 1.47E+4} for the optimal model of the Median and conventional Grid Search algorithms, respectively.

Table 3 Case #1 (HCM, K-fold = 2)

Hyperparameter	Grid Search	Median Grid Search					
		Model 1	**Model 2**	Model 3	Model 4	Model 5	Model 6
ς	3.16E-3	3.16E-3	**3.16E-3**	2.15E-2	2.15E-2	2.15E-3	1.00E-2
C	4.64E+3	4.64E+4	**2.15E+3**	1.00E+5	1.00E+5	4.64E+3	3.16E+3
γ	6.81E-3	2.15E-3	**1.00E-2**	2.15E-3	2.15E-3	1.00E-2	1.00E-2
Error rate	2.75E-4	3.19E-4	**2.79E-4**	3.03E-4	3.03E-4	2.97E-4	2.82E-4

Table 4 Case #2 (SA, K-fold = 2)

Hyperparameter	Grid Search	Median Grid Search					
		Model 1	Model 2	Model 3	Model 4	**Model 5**	Model 6
ς	1.47E-2	1.00E-2	1.00E-2	2.15E-2	1.00E-2	**1.00E-2**	1.47E-2
C	3.16E+4	4.64E+3	4.64E+3	2.15E+3	2.15E+3	**1.00E+4**	6.81E+3
γ	3.16E-3	1.00E-2	1.00E-2	3.16E-3	4.64E-3	**6.81E-3**	6.81E-3
Error rate	4.96E-4	5.14E-4	5.14E-4	6.02E-4	5.16E-4	**4.98E-4**	5.12E-4

Table 5 Case #3 (HCM, K-fold = 3)

Hyperparameter	Grid Search	Median Grid Search					
		Model 1	Model 2	Model 3	Model 4	**Model 5**	Model 6
ς	1.00E-2	1.00E-4	1.00E-4	3.16E-2	2.15E-2	**6.81E-3**	3.16E-3
C	2.15E+4	4.64E+4	1.00E+4	1.00E+5	1.00E+5	**1.47E+4**	1.47E+4
γ	1.00E-2	1.47E-3	1.00E-2	1.00E-2	1.47E-3	**1.00E-2**	1.00E-2
Error	1.69E-4	2.80E-4	1.93E-4	2.33E-4	3.16E-4	**1.73E-4**	1.81E-4

Table 6 Case #4 (SA, K-fold = 3)

Hyperparameter	Grid Search	Median Grid Search					
		Model 1	Model 2	Model 3	Model 4	**Model 5**	Model 6
ς	1.00E-2	2.15E-2	2.15E-2	3.16E-2	1.47E-2	**1.00E-2**	2.15E-2
C	1.47E+4	6.81E+3	1.00E+5	2.15E+4	2.15E+4	**6.81E+3**	1.00E+5
γ	1.00E-2	1.00E-2	1.00E-2	6.81E-3	6.81E-3	**1.00E-2**	1.00E-2
Error rate	2.50E-4	3.83E-4	3.75E-4	5.15E-4	2.74E-4	**2.66E-4**	3.75E-4

Based on the results of the training process as reported from Tables 3 to 6, the optimal hyperparameters of the conventional Grid Search and the Median Grid Search algorithm can be obtained. Then, the testing process (as described in Fig. 4) will be performed. The prediction values of both algorithms along with their error rates RMSE are determined. These results are shown in Table 7.

The results presented in Table 7 clearly indicate that the optimal hyperparameters for the conventional Grid Search and the Median Grid Search algorithms are different. Besides, the error rates of the Median algorithm are smaller than the conventional algorithm in the testing process. Let consider case #1. When using the conventional algorithm, the error rate RMSE was of 2,732.84 MW. Meanwhile, the Median algorithm gave a better result with RMSE value of 2,670.74 MW. The same results were also obtained in the case of #2, #3, and #4.

Fig. 8 shows the prediction and test series for the last day of the testing process corresponding to the cases #1, 2, 3, 4. The black curve is referred the real value of the load, while the blue and red ones - the prediction values for the conventional and Median Grid Search algorithm, respectively. The first point noticed in Fig. 8 is that both the prediction curves of the conventional algorithm and the Median algorithm well fit the actual value of the load. A closer observation shows the presence of a few locations on the prediction curve of the Median algorithm (brown circles in Fig. 8), proving the better response than those for the traditional algorithm.

The analysis of results in Table 7 and Fig. 8 reveals the smaller error rate of the Median Grid Search algorithm, i.e., the more accuracy in comparison with the conventional algorithm, verifying the effectiveness of the proposed Median Grid Search algorithm in our research.

Table 7 The accuracy error RMSE (MW) in the testing process

Cases	Conventional Grid Search		Median Grid Search	
	Optimal hyperparameter	RMSE	Optimal hyperparameter	RMSE
Case #1	$\varepsilon = 3.16E{-}3$, $\gamma = 6.81E{-}3$, $C = 4.64E{+}3$	**2,732.84**	$\varepsilon = 3.16E{-}3$, $\gamma = 1.00e{-}2$, $C = 2.15E{+}3$	**2,670.74**
Case #2	$\varepsilon = 1.47E{-}2$, $\gamma = 3.16E{-}3$, $C = 3.16E{+}4$	**1,482.86**	$\varepsilon = 1.00E{-}2$, $\gamma = 6.81E{-}3$, $C = 1.00E{+}4$	**1,285.76**
Case #3	$\varepsilon = 1.00E{-}2$, $\gamma = 1.00E{-}2$, $C = 2.15E{+}4$	**2,787.99**	$\varepsilon = 6.81E{-}3$, $\gamma = 1.00E{-}2$, $C = 1.47E{+}4$	**2,778.54**
Case #4	$\varepsilon = 1.00E{-}2$, $\gamma = 1.00E{-}2$, $C = 1.47E{+}4$	**447.58**	$\varepsilon = 1.00E{-}2$, $\gamma = 1.00E{-}2$, $C = 6.81E{+}3$	**369.52**

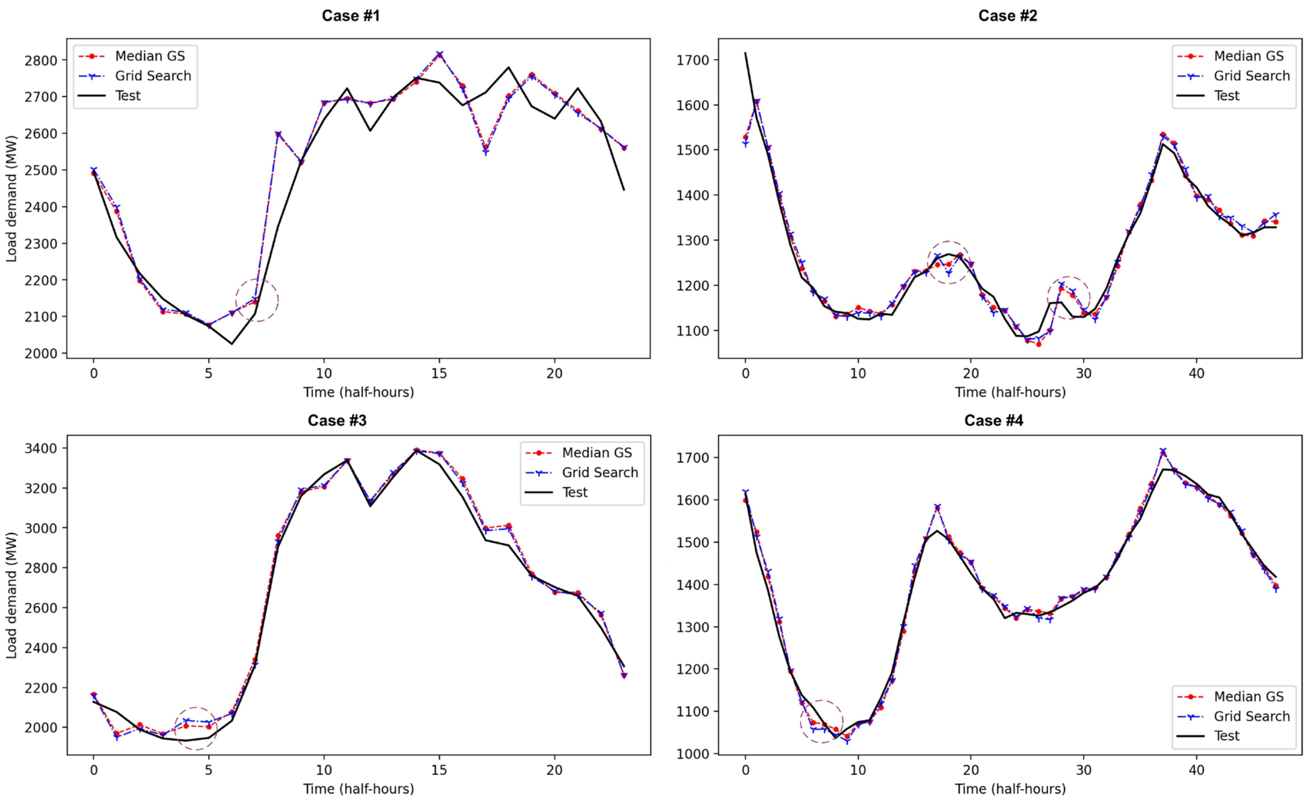


Fig. 8 The prediction and test series in the testing process

4 Conclusions

An efficient Median Grid Search methodology based on the SVR model and their hyperparameters was proposed in this paper. Besides, a methodology for evaluating the Median Grid Search algorithm was presented to benchmark with the conventional Grid Search. In the training process, the optimal hyperparameters were specified. In the testing process, error rates were used to evaluate the Median Grid Search algorithm and the conventional Grid Search algorithm. Both South Australia state, Australia and Ho Chi Minh City load demands were used to verify the performance of the methodology. The results for the Median Grid Search algorithm indicated the existence of an optimal hyperparameter that is different from the conventional Grid Search algorithm. Analyzing the obtained results in the testing process showed that the error rates of the Median Grid Search algorithm gave good results with values smaller than the conventional Grid Search algorithm. The positive results obtained in this study suggest an effective way to apply SVR for load forecasting in particular and time series in general.

References

[1] Mohi Ud Din, G., Mauthe, A. U., Marnerides, A. K. "Appliance-Level Short-Term Load Forecasting Using Deep Neural Networks", In: 2018 International Conference on Computing, Networking and Communications (ICNC), Maui, HI, USA, 2018, pp. 53–57. ISBN 978-1-5386-3651-0
https://doi.org/10.1109/ICCNC.2018.8390366

[2] Aydarous, A. A., Elshahed, M. A., Moustafa Hassan, M. A. "Short-Term Load Forecasting Approach Based on Different Input Methods of One Variable: Conceptual and Validation Study", In: 2018 Twentieth International Middle East Power Systems Conference (MEPCON), Cairo, Egypt, 2019, pp. 179–184. ISBN 978-1-5386-6652-4
https://doi.org/10.1109/MEPCON.2018.8635115

[3] Walther, J., Spanier, D., Panten, N., Abele, E. "Very Short-Term Load Forecasting on Factory Level - A Machine Learning Approach", Procedia CIRP, 80, pp. 705–710, 2019.
https://doi.org/10.1016/j.procir.2019.01.060

[4] Uri, N. D., Maybee, J. S. "Forecasting the Load Duration Curve Using Box-Jenkins Time Series Analysis", Engineering Optimization, 3(4), pp. 193–199, 1978.
https://doi.org/10.1080/03052157808902391

[5] Türkay, B. E., Demren, D. "Electrical Load Forecasting Using Support Vector Machines: A Case Study", International Review of Electrical Engineering, 6(5), pp. 49–53, 2011.

[6] Kavousi-Fard, A., Kavousi-Fard, F. "A New Hybrid Correction Method for Short-Term Load Forecasting Based on ARIMA, SVR and CSA", Journal of Experimental and Theoretical Artificial Intelligence, 25(4), pp. 559–574, 2013.
https://doi.org/10.1080/0952813X.2013.782351

[7] Sadli, M., Fajriana, Fuadi, W., Ermatita, Pahendra, I. "Electrical Peak Load Forecasting Using Long Short-Term Memory and Support Vector Machine", IOP Conference Series: Materials Science and Engineering, 725, 012060, 2020.
https://doi.org/10.1088/1757-899X/725/1/012060

[8] Sano, N., Higashinaka, K., Suzuki, T. "Efficient Parameter Selection for Support Vector Regression Using Orthogonal Array", In: 2014 IEEE International Conference on Systems, Man and Cybernetics, San Diego, CA, USA, 2014, pp. 2256–2261. ISBN 978-1-4799-3840-7
https://doi.org/10.1109/smc.2014.6974261

[9] Lin, S., Zhang, S., Qiao, J., Liu, H., Yu, G. "A Parameter Choosing Method of SVR for Time Series Prediction", In: The 9th International Conference for Young Computer Scientists, Hunan, China, pp. 130–135, 2008. ISBN 978-0-7695-3398-8
https://doi.org/10.1109/ICYCS.2008.393

[10] Yang, L., Shami, A. "On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice", Neurocomputing, 415, pp. 295–316, 2020.
https://doi.org/10.1016/j.neucom.2020.07.061

[11] Lin, S.-W., Ying, K.-C., Chen, S.-C., Lee, Z.-J. "Particle Swarm Optimization for Parameter Determination and Feature Selection of Support Vector Machines", Expert Systems with Applications, 35(4), pp. 1817–1824, 2008.
https://doi.org/10.1016/j.eswa.2007.08.088

[12] Čabala, J., Jadlovský, J. "Choosing the Optimal Production Strategy by Multi-Objective Optimization Methods", Acta Polytechnica Hungarica, 17(5), pp. 7–26, 2020.
https://doi.org/10.12700/APH.17.5.2020.5.1

[13] Kaur, G., Gill, S. S., Rattan, M. "Whale Optimization Algorithm for Performance Improvement of Silicon-on-Insulator Finfet", International Journal of Artificial Intelligence, 18(1), pp. 63–81, 2020.

[14] Moattar, M., Moradi, M. H. "Conflict Monitoring Optimization Heuristic Inspired by Brain Fear and Conflict Systems", International Journal of Artificial Intelligence, 18(1), pp. 45–62, 2020.

[15] Precup, R.-E., Hedrea, E.-L., Roman, R.-C., Petriu, E. M., Szedlak-Stinean, A.-I., Bojan-Dragos, C.-A. "Experiment-Based Approach to Teach Optimization Techniques", IEEE Transactions on Education, 64(2), pp. 88–94, 2021.
https://doi.org/10.1109/TE.2020.3008878

[16] Phan, A. V., Nguyen, M. L., Bui, L. T. "Feature weighting and SVM parameters optimization based on genetic algorithms for classification problems", Applied Intelligence, 46, pp. 455–469, 2017.
https://doi.org/10.1007/s10489-016-0843-6

[17] Božić, M., Stojanović,M. "Application of SVM methods for mid-term load forecasting", Serbian Journal of Electrical Engineering, 8(1), pp. 73–83, 2011.
https://doi.org/10.2298/sjee1101073B

[18] Huang, Q., Mao, J., Liu, Y. "An improved grid search algorithm of SVR parameters optimization", In: 14th International Conference on Communication Technology Proceedings, Chengdu, China, pp. 2012, pp. 1022–1026. ISBN 978-1-4673-2100-6
https://doi.org/10.1109/ICCT.2012.6511415

[19] Zakrani, A., Najm, A., Marzak, A. "Support Vector Regression Based on Grid-Search Method for Agile Software Effort Prediction", In: 5th International Congress on Information Science and Technology (CiSt), Marrakech, Morocco, 2018, pp. 492–497. ISBN 978-1-5386-4386-0
https://doi.org/10.1109/CIST.2018.8596370

[20] Thanh Ngoc, T., Dai, L. V., Thuyen, C. M. "Support Vector Regression based on Grid Search method of Hyperparameters for Load Forecasting", Acta Polytechnica Hungarica, 18(2), pp. 143–158, 2021.
https://doi.org/10.12700/aph.18.2.2021.2.8

[21] Ceperic, E., Ceperic, V., Baric, A. "A Strategy for Short-Term Load Forecasting by Support Vector Regression Machines", IEEE Transactions on Power Systems, 28(4), pp. 4356–4364, 2013.
https://doi.org/10.1109/TPWRS.2013.2269803

[22] Liu, Z., Xu, H. "Kernel Parameter Selection for Support Vector Machine Classification", Journal of Algorithms & Computational Technology, 8(2), pp. 163–177, 2014.
https://doi.org/10.1260/1748-3018.8.2.163

[23] Chen, T., Huang, W., Wu, R., Ouyang, H. "Short Term Load Forecasting Based on SBiGRU and CEEMDAN-SBiGRU Combined Model", IEEE Access, 9, pp. 89311-89324, 2020.
https://doi.org/10.1109/ACCESS.2020.3043043

[24] Castellini, A., Bianchi, F., Farinelli, A. "Predictive Model Generation for Load Forecasting in District Heating Networks", IEEE Intelligent Systems, 36(4), pp. 86–95, 2020.
https://doi.org/10.1109/MIS.2020.3005903

[25] Masayuki K., Nakano, R. "Optimizing Sparse Kernel Ridge Regression hyperparameters based on leave-one-out cross-validation", In: Proceedings of International Joint Conference on Neural Networks, Orlando, FL, USA, 2007, pp. 3463–3468. ISBN 978-1-4244-1379-9

[26] Brownlee, J. "Statistical Methods for Machine Learning: Discover how to Transform Data into Knowledge with Python", Machine Learning Mastery, 2019. ISBN 979-8540446273 [online] Available at: https://dokumen.pub/statistical-methods-for-machine-learning.html [Accessed: 25 July 2022]

[27] Khan, S. R., Noor, S. "Short Term Load Forecasting Using SVM Based PUK Kernel", In: 20203rd International Conference on Computing, Mathematics and Engineering Technologies: Idea to Innovation for Building the Knowledge Economy - ICoMET, Sukkur, Pakistan, 2020, pp. 1–9. ISBN 978-1-7281-4971-4
https://doi.org/10.1109/iCoMET48670.2020.9074068

[28] Nichiforov, C., Stamatescu, G., Stamatescu, I., Fagarasan, I., Iliescu, S. S. "Intelligent Load Forecasting for Building Energy Management Systems", In: 2018 IEEE International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 2018, pp. 896–901. ISBN 978-1-5386-6090-4
https://doi.org/10.1109/ICCA.2018.8444311