Combining PSO-SVR and Random Forest Based Feature Selection for Day-ahead Peak Load Forecasting

Huachao Zhai, Jinxing Che

Abstract—**In recent years, with computer and Internet technology developing at a breakneck speed, ensuring the fair dispatching and stable operation of the smart grid has increasingly become the focus of power companies. The emergence of the smart grid makes it difficult for a single model to accurately predict the complex power data, and the combined prediction model has become the main target of research by experts and scholars. To this end, a new combination prediction model is proposed in this research. The missing values in the data set are filled by the forecasts of the Random Forest (RF) approach, then the significance of input attributes is computed. After that, the feature selection is performed, and the peak load forecasting is modeled by using support vector regression based on particle swarm optimization (PSO-SVR). The experimental results using real data from a county in Jiangxi Province indicate that the model's prediction ability with feature selection is better than that of the model without feature selection.**

Index Terms—**peak load forecasting, support vector regression (SVR), feature selection, random forest (RF), particle swarm optimization (PSO)**

I. Introduction

Electric power load forecasting is a critical component of the entire power system. Electric power load forecasting is increasingly becoming crucial to maintaining power systems' safe dispatch and dependable operation. As infrastructure continues to upgrade, technology advances exponentially, and the smart grid emerges. As we all know, coal and other raw materials of thermal power are finite natural resources. Once these resources deplete, they cannot replenish quickly. As a result, all countries in the world are following the sustainable development plan. The stable operation of the electric power system can ensure the normal life of people and safety in the face of a worldwide outbreak of Coronavirus disease 2019 (COVID-19). Therefore, knowing how to transport electric energy efficiently and fairly is crucial. Load forecasting is an equation-based measurement of the change in the degree of power consumption in a certain area over time. For instance, we can study the electricity usage of households, communities, or regions. If accurate power forecasting is employed, the problem of electricity distribution could also be eased and rationalized.

Electrical load data is a time series. The three classifications depending on the duration of the investigated period are short-term load forecasting, medium-term load forecasting, and long-term load forecasting. Short-term load forecasting lasts a few hours or days, while medium-term load forecasting lasts a few days or months. Long-term load forecasting has a broad research scope. The year is a common study unit, and it examines power usage conditions over one or more years. It provides effective support for the power dispatching problem regardless of the form of time range prediction. Furthermore, 1% of accuracy increase in load forecasting can reduce the cost of service [1]. Therefore, even a small improvement in accuracy can result in cost savings. Accurate load forecasting ensures the electrical grid's stability and allows for sensible scheduling. As a result, the load forecasting model's accuracy must be increased.

For load forecasting, there are several kinds of testing approaches and prediction models. It can divide into standard methods and methods of artificial intelligence [2-4], based on various prediction methods. The author in [5] forecasts using auto-regressive moving average (ARMA) models. The author in [6] uses autoregressive integrated moving average (ARIMA) autoregressive models to predict the power load. The typical forecasting methodology is operated by a mathematical equation using a simple linear regression or multiple regression mathematical model to depict the input-output connection. The benefits of these approaches are straightforward to apply, but the pitfalls are often evident. They do not handle nonlinear data. Scholars have started to investigate artificial intelligence approaches to solve the issue that conventional prediction methods do not accommodate nonlinear data. Deep recurrent neural networks [7], random forest (RF) [13, 14], support vector machine (SVM) [9-12], and artificial neural network (ANN) [8] apply in these artificial approaches. However, different configurations of functions impact the predictive outcomes and the prognostic model's sophistication. To assist anticipate the electricity load, this paper provides a hybrid short-term load forecasting model based on random forest (RF) and support vector regression (SVR). This model uses the random forest to pick the informative features, and we also change SVR parameters by PSO. It is more accurate than using a single form of forecasting.

The study was supported by the National Statistical Science Research Project (Grant No. 2020LZ03) and the “Thousand talents plan” for high-level talents in Jiangxi Province (Grant No. jxsq2019201064). The research was supported by the National Natural Science Foundation of China (Grant No. 71971105).

Huachao Zhai is a postgraduate in School of Information Engineering, Nanchang Institute of Technology, Nanchang 330099, Jiangxi, China (corresponding author, e-mail: huachaozhai@163.com).

Jinxing Che is an associate professor in the College of Science, Nanchang Institute of Technology, Nanchang 330099, Jiangxi, China (corresponding author, e-mail: jinxingche1@163.com).

II. METHODS

A. Decision Tree

Decision tree (DT) is a classical machine learning method, and it is an unsupervised learning process. The structure of the decision tree is an inverted tree. In the decision tree, the node containing all the training data is called the vertex of DT, and it can also be called the root node. To segment the non-leaf nodes, we choose the best feature from the initial feature space until the stop condition is reached. All of the branches and nodes after segmentation form a tree number, which is the decision tree. If all the training samples in a node belong to the same data type, then the node is called a leaf node, and there is no need for further splitting. A split rule is a path that connects the root node to the corresponding leaf node. The split rule is an "if-then" rule, which can use to create decision trees.

The decision tree is simple and effective. However, as the amount of training data increases, so does the amount of computation. The DT is highly dependent on the training data. Therefore, the generalization ability of the decision tree is not enough. If the development of the tree is not limited, the model is prone to over-fitting or falling into local optimum.

B. Random Forest

Random Forest (RF) is a technique of ensemble learning. It is common in classification and regression tasks. Based on decision trees and ensemble learning, random forest combines them organically. It inherits many advantages of decision trees and increases the model's generalization ability. Different from decision trees, the computational complexity of random forest will not increase significantly with improving prediction accuracy. In 2001, Bryman put forward the random forest algorithm. Bag and CART methods are used to create a forest, which is a collection of deciduous trees.

S is assumed to be a primary data set of n samples. When creating the training set for each CART using bagging, the probability that each sample can be selected is 1/n. Due to bagging, some samples from the initial sample space may never be chosen as training data, while others may be chosen several times. The Out-of-Bag (OOB) data set for the tree comprises all samples that will never be selected. As a result, each CART's training data set is unique, decreasing forest tree-to-tree associations. The diversity of CART ensures the anti-noise performance of RF, and it reduces the model's sensitivity to outliers.

Random forest is different from the decision tree. Specifically, for each non-leaf node, random forest produces a candidate feature set m at random. The parameter m_{try} controls m, which is a subset of M's initial feature space. Then, the best feature for dividing non-leaf nodes is selected from m.

In a random forest, every tree is unique. Therefore, the generalization degree of random forest is very high, and only a few parameters need to be adjusted. The number of trees and the dimension of candidate feature sets are two unique parameters that affect the performance of random forest prediction. When there is a scarcity of trees, the prediction effect of random forest is not ideal and does not match the standards of regression prediction. With the growth in tree numbers, the generalization error of a random forest tends to have a stable upper bound, according to the rule of strong numbers and tree structure. As a result, random forests are unlikely to overfit.

m_{try} has a more substantial influence on the random forest than n_{tree}. After many experimental studies, scholars have found the empirical value of m_{try} when using the random forest to make regression predictions.

$$m_{try} = \frac{t}{3} \tag{1}$$

In equation (1), t represents the size of the initial feature set.

After the training process accomplishes, the random forest can return the OOB_{error} and the value of each feature's importance. Equation (2) gives the mathematical calculation method of OOB_{error}. The OOB_{error} can be used to calculate the generalization error of the random forest.

$$OOB_{error} = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_{prei}) \tag{2}$$

Where n equals n_{tree}, y_i is the actual data of the i-th tree, and y_{prei} is the output data.

The final predictive result of regression can obtain by calculating the average value of all trees' output. Equation (3) shows the specific mathematical calculation equation of the result.

$$y_{pre} = \frac{1}{n}\sum_{i=1}^{n} y_{prei} \tag{3}$$

In equation (3), y_{pre} is the forecasting result of forecastin g, y_{prei} is the output value of the *i-th* tree.

C. Random Forest based Feature selection

Because of the large amount of data in short-term power load forecasting, there are many characteristics in the data, such as daily meteorological data and historical load data. Since there may be redundant feature attributes in the entire feature space, if we select all the features toward load forecasting, it will impair the accuracy and efficiency of the forecasting model. Therefore, it is necessary to delete and filter redundant features. The early method is to eliminate unimportant features that are considered based on the experience of experts. Nowadays, researchers usually use algorithms to filter features. The random forest can return the significant value of each element, and the model only needs to modify a few parameters. Its calculation speed is breakneck so that it can use as a feature selection tool.

The importance value of each feature can calculate by using the random forest, and its working principle is as follows:

Firstly, the OOB_{error} of a current tree can be obtained with equation (2). Assuming this tree is the *i-th* tree, the corresponding OOB_{error} is recorded as $OOBerror_i$. Then, the corresponding feature values are randomly rearranged in the OOB data set, while other features are not affected. This process will generate a new OOB data set and it is named $OOB_i^{'}$. And the OOB_{error} for $OOB_i^{'}$ is calculated and

saved as $OOBerror_i^{'}$. Therefore, we can use I_i to express the important value of the feature in the *i-th* tree and the calculation equation as shown in equation (4).

$$I_i = OOBerror_i^{'} - OBBerror_i \tag{4}$$

Repeat this process for each tree. The average value of the calculated results of each tree determines the importance value of the final feature. The calculation equation as shown in equation (5).

$$I = \frac{1}{n}\sum_{i=1}^{n} I_i \tag{5}$$

In equation (5) I is the importance value of one feature.

If a characteristic is really essential, the values of its various samples will differ. The higher the level of importance, the more obvious the characteristics, and the closer the relationship with the predicted results.

D. Support vector regression

Support vector machine (SVM) is a supervised learning system usually used to solve two kinds of classification problems. Support vector regression (SVR) machine is a subset of support vector machine, often used in various forecasting fields. When dealing with classification, the result of *SVM* is to find a hyperplane and maximize the distance between the hyperplane and the nearest sample point. It is also vital to find a hyperplane while dealing with regression situations. However, the distance between this hyperplane and the farthest sample point must be the shortest. And all of the other sample positions must deviate from the hyperplane as little as possible.

The regression theory of *SVR* is as follows: Given a data set $\{(x_i, y_i), i = 1,2,...,n\}$, n represents the number of items in the data set, X_i represents the n-dimensional vector corresponding to the *i-th* element in the data set. $X_i = \{x_1, x_2, ..., x_n\} \in R^n, y_i \in R$, X_i, y_i correspond to each another. As training data, take N samples from the data set.

In a support vector regression machine model, we must transform the training data into the high-dimension feature space from the low-dimension feature space. In this high-dimensional feature space, there is an optimal hyperplane. It shows the functional connection between the output data (dependent variables) and input data (independent variables). Equation (6) indicates this functional relationship.

$$f(x) = \mathbf{W}^T \varphi(x) + b \tag{6}$$

In equation (6), $f(x)$ represents the predicted value of the model, W represents the weight coefficient and b represents the adjustable coefficient. $\varphi(x)$ is a mapping function of high-dimensional feature space.

The insensitive loss function marks ε, and the residual error between actual data and the predicted value marks $R(x, y)$. Equation (7) and equation (8) can show the calculational function about ε and $R(x, y)$, respectively.

$$| y - f(x) |_{\varepsilon} = \max(0, | y - f(x) | - \varepsilon) \tag{7}$$

$$R(x, y) = y - f(x) \tag{8}$$

Where y is the actual data, *f(x)* is the prediction value.

Therefore, the ideal regression model should include all residuals, as shown in equation (9).

$$-\varepsilon \leqslant R(x, y) \leqslant \varepsilon \tag{9}$$

All training data should satisfy equation (9). Therefore, if the residual is satisfied $R(x, y) = \pm\varepsilon$, then the hyperplane is closest to the farthest data sample. The hyperplane $R(x, y) = 0$ and the distance between $data(x, y)$ is defined as $| R(x, y) | / \| W^* \|$, and W^* is defined via equation (10).

$$W^* = \left(1, -\mathbf{W}^T\right)^T \tag{10}$$

The maximum distance between $data(x, y)$ and the hyperplane $R(x, y) = 0$ is assumed δ. As a result, all training data should satisfy equation (11). Moreover, the higher the δ is, the better the generalization ability of the support vector regression model is.

$$| R(x, y) | \leqslant \delta \| W^* \| \tag{11}$$

If the maximum distance $| R(x, y) | = \varepsilon$. Equation (11) now can be rewritten as equation (12).

$$\varepsilon = \delta \| W^* \| \tag{12}$$

However, not all training error always falls within the $(-\varepsilon, \varepsilon)$ range, and training error occasionally exceeds this range. If the training error is less than $-\varepsilon$, it is recorded as ξ_i. If the training error is beyond ε, it is expressed by ξ_i^*. Equation (13) and equation (14) respectively describe these two parameters ξ_i, ξ_i^*.

$$\xi_i = \begin{cases} 0, & R(x_i, y_i) - \varepsilon \leqslant 0 \\ R(x_i, y_i) - \varepsilon, & others \end{cases} \tag{13}$$

$$\xi_i^* = \begin{cases} 0, & \varepsilon - R(x_i, y_i) \leqslant 0 \\ \varepsilon - R(x_i, y_i), & others \end{cases} \tag{14}$$

The ultimate goal of a support vector regression machine is to find an ideal hyperplane between the insensitive function and the training data to minimize training error. Equation (15) describes the objective function of SVR.

$$\min F\left(W, b, \xi_i, \xi_i^*\right) = \frac{1}{2} \| W \|^2 + C\sum_{i=1}^{N}\left(\xi_i + \xi_i^*\right) \tag{15}$$

Following are related restrictions:

$$y_i - \mathbf{W}^T \varphi(x_i) - b \leqslant \varepsilon + \xi_i \quad i = 1, 2, \ldots, \bar{N}$$

$$\mathbf{W}^T \varphi(x_i) + b - y_i \leqslant \varepsilon + \xi_i^* \quad i = 1, 2, \ldots, \bar{N}$$

$$\xi_i \geqslant 0, \xi_i^* \geqslant 0 \quad i = 1, 2, \ldots, \bar{N}$$

Where C is the penalty factor for balancing the training error and the distance between the hyperplane and the training data.

The weight coefficient **W** in equation (6) can be expressed as equation (16) following the completion of the quadratic programming solution with unequal restrictions.

$$\mathbf{W} = \sum_{i=1}^{N} (\beta_i^* - \beta_i) \varphi(x_i) \tag{16}$$

The Lagrange multipliers β_i^* and β are produced by completing the quadratic programming issue. Finally, the SVR functional relationship can be written as equation (17).

$$f(x) = \sum_{i=1}^{N} (\beta_i^* - \beta_i) K(x_i - x) + b \tag{17}$$

In which, $K(x_i - x)$ is a kernel function, which can transfer

training data to the high dimensional space in a nonlinear way. Therefore, nonlinear issues can be solved using support vector regression, such as load forecasting.

E. Particle Swarm Optimization (PSO)

Some parameters in machine learning models must be selected manually, which are called super parameters. In the support vector machine model, different kernel functions have different parameters. Different parameter combinations have different effects on the prediction results of the model. Choosing the best parameters can improve the prediction accuracy of the prediction model and reduce the model's prediction error. Particle Swarm Optimization (PSO) is a population-based intelligent optimization technology inspired by the behavior of foraging birds. By updating the speed and position of particles, we can change the particle's position to obtain the best position of particles. The position of the optimal particle expresses the ideal solution of the correlation function. Equation (18) describes the particle velocity update equation.

$$v_i^d = \omega v_i^d + c_1 r_1 (p_i^d - x_i^d) + c_2 r_2 (p_j^d - x_i^d) \quad (18)$$

$$x_i^d = x_i^d + \alpha v_i^d \quad (20)$$

Where ω is a non-negative inertial factor, c_1 and c_2 are non-negative constants, r_1 and r_2 are random integers in the range [0,1]. Equation (19) depicts an equation for updating the particle position. In equation (19) α is a limiting factor of speed control, x_i , v_i are the position and velocity of the current particle.

F. Indicator of model assessment

In this section, we calculate the model scores of each parameter combination and select the best parameter. Many approaches, such as mean absolute error (MAE), root mean square error (RMSE), coefficient of determination(R^2), and mean absolute percentage error (MAPE), can be used to evaluate the predictive ability of the model. Equation (21- 24) shows the mathematical expressions of these evaluation functions.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \quad (21)$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \quad (22)$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2} \quad (23)$$

$$R^2 = 1 - \frac{\sum_i (Y_i - \hat{Y}_i)^2}{\sum_i (Y_i - \frac{1}{n}\sum_{i=1}^{n} Y_i)^2} \quad (24)$$

Among these, $\hat{Y}_i$ is the predict value of the model output, Y_i is the true value.

III. Experiments

In this experiment, a combined model based on random forest and support vector machine is proposed. According to the actual power load data, the local power load peak value is predicted. Figure 1 shows the forecast flow chart of the whole combined model. First, the missing values of data sets are detected. If the data set has been completed, please proceed directly to the next step. On the contrary, if there is a missing part in the data set, it is necessary to fill in the missing parts in the data set before continuing to the next step. This operation can also be referred to as data preprocessing. Second, there are many features in the original load data. Among these features, it is not difficult to infer that some are important while others are not. In order to calculate the importance of each characteristic to the peak load, a random forest regression model is introduced. Using the random forest regression model, we can get the important index of each characteristic. The third step is to remove the six features with the smallest index from the dataset and get a new dataset. In this new data set, all characteristic attributes are highly correlated with peak load. Finally, based on the new data set, the support vector regression model of particle swarm optimization is used to predict the peak value.

x

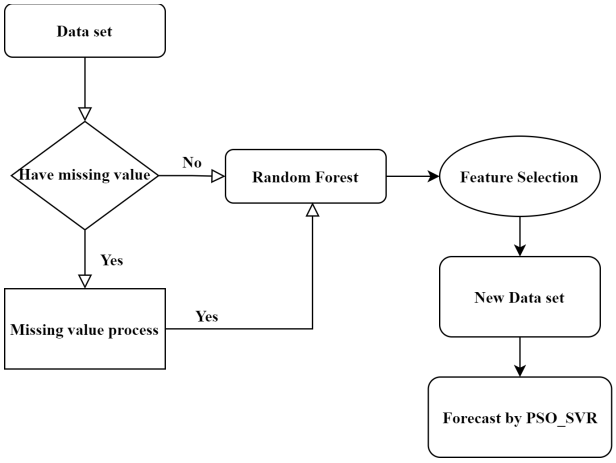

Fig. 1 The forecast flow chart of the whole combined model

The experiment data comes from the annual peak power load data of a county in Jiangxi Province in 2013. In the sampling period of one day, the data captured the highest and lowest temperatures and peak power load of the day. Due to the influence of the sampling and data transmission program, the data set missed two load peaks and only accessed relevant temperature data. Load data is a kind of time-series data, and the data in a day should be related to the previous data to some extent. Therefore, it is very important to collect comprehensive and continuous data. In this experiment, the data without missing values and 7-day historical load data are taken as the training data sets. The projection value of the random forest regression model is used as the new data to replace the missing load data, which is based on stochastic forest regression. In the final data collection, none of the missing values. Figure 2 depicts the entire data set and the original data set.

We perform the correlation analysis on the whole data set to assess the degree of correlation more correctly between the current and previous data. Figure 3 displays the ACF plot of load data. And we find that the current data is associated with the top 20 data. As a result, the historical data of the first 20 days are selected and combined with the temperature data of that day to create a new data set, and then we use the data set to model the prediction model. The random forest regression model and the PSO-SVR model make predictions based on the new data set.

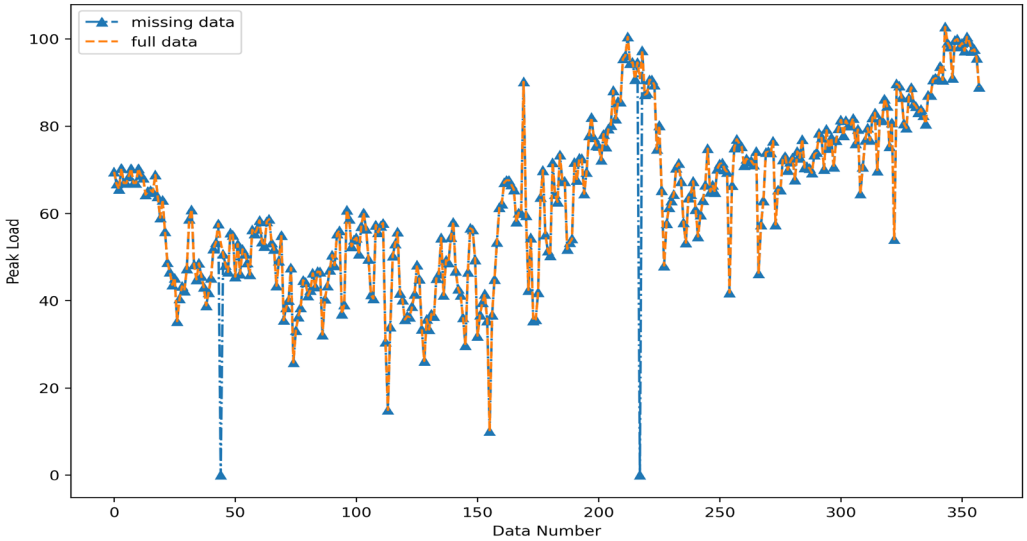

Fig. 2 Complete data without missing values

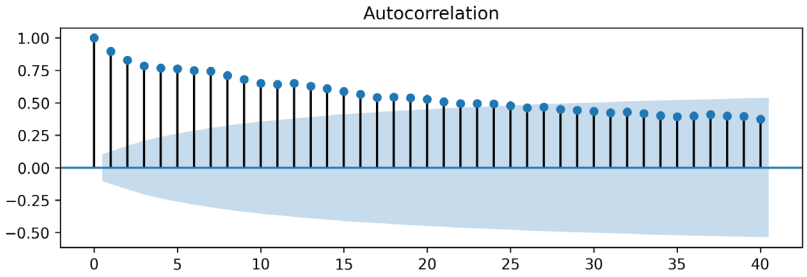

Fig.3 ACF plot of load data

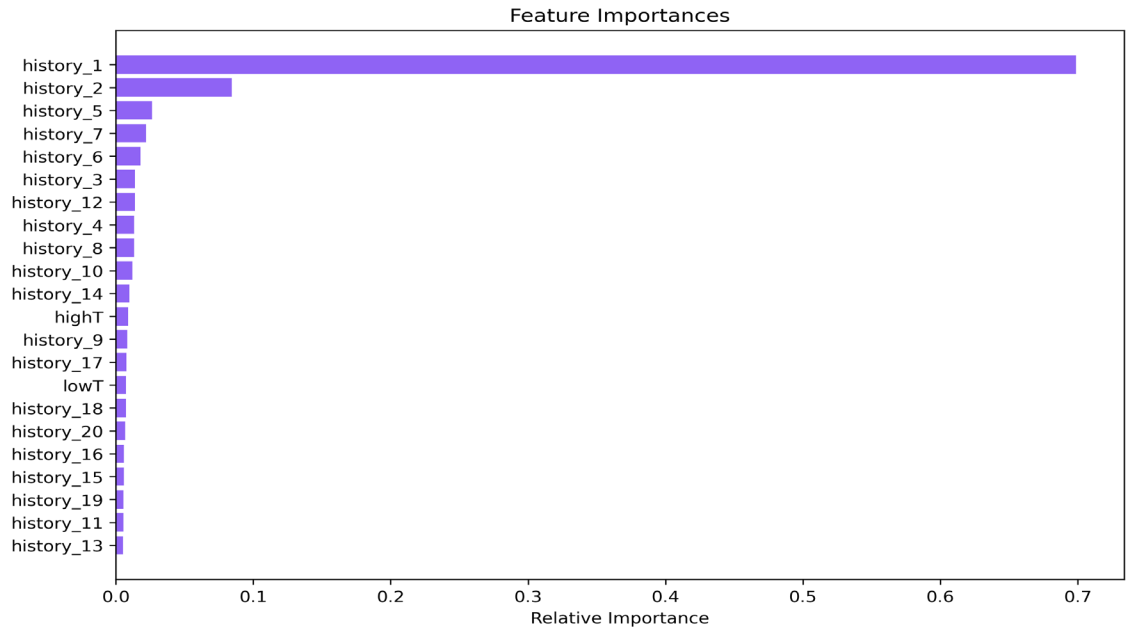

Fig.4 Feature importance value

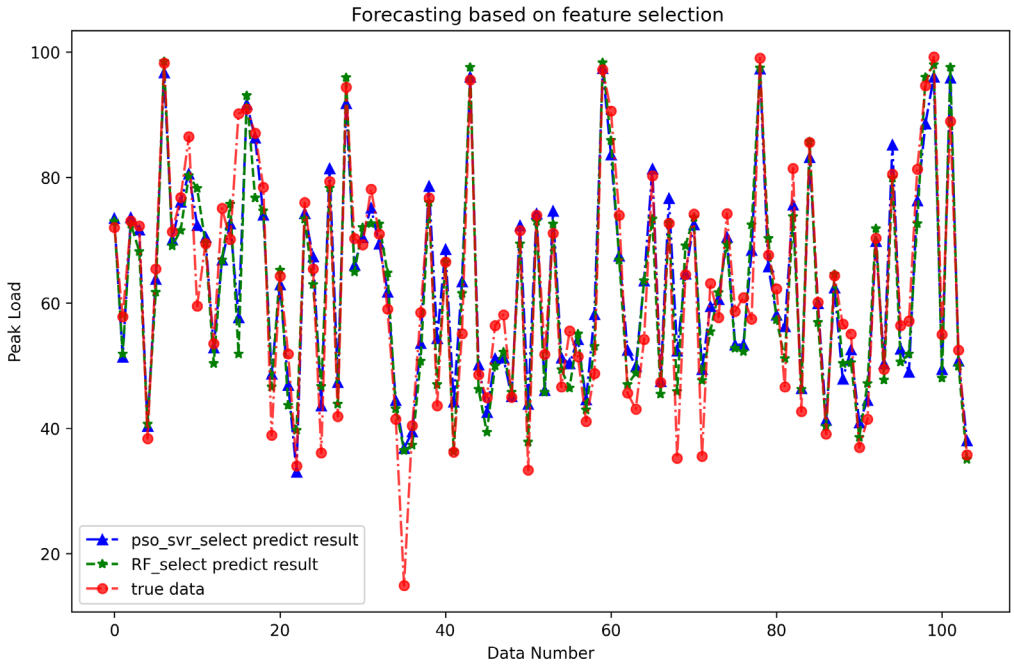

Fig.5 Experimental results of model on peak load forecasting

Once the random forest model training is completed, it can return the importance value of each feature. Prediction results are affected by characteristics. The numerical value of the relevance of each feature is organized according to its size, and the result is shown in figure 4.

The variable $history\_i, (i = 1, 2, ..., 20)$ represents the historical data of the first 20 days, so $history\_1$ represents the historical load data of yesterday. The lowest and highest temperatures in a day are expressed as lowT and highT, respectively.

As mentioned in the previous article, the greater the influence of features on prediction, the higher the relevance of features. Finally, six features that have the least influence on the predicted value are deleted and constructed a new data set to eliminate redundant features from the data set. A random forest regression prediction model and a PSO_SVR prediction model are established based on the new data set. Figure 5 shows the final forecast result. The prediction model with feature selection and the prediction model without feature selection are compared. We use four error indicators, MAE, MAPE, RMSE, and R^2, to compare the four prediction models used in the experiment. Table I shows the specific evaluation results.

TABLE I
Indicators of model evaluation for four models

Model	MAE	MAPE	RMSE	R^2
RF	4.87	9.53	7.03	84.91
RF_select	4.74	**9.25**	6.89	85.50
PSO_SVR	4.78	10.13	6.79	85.96
PSO_SVR _select	**4.53**	9.34	**6.51**	**87.07**

The value with the best error index is written in bold type in table I. By comparing the results, we can see that the error indices of the two models improve after feature selection. The model's score based on R^2 is shown in figure 6. The histogram can perceive the difference more intuitively between the four prediction results.

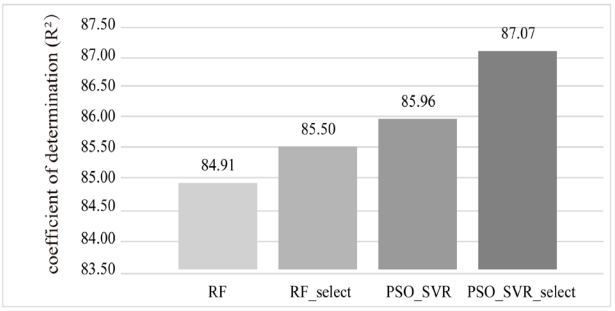

Fig. 6 Comparison of model's determination coefficients

IV. Conclusion

In this paper, a combination forecasting model based on random forest and PSO_SVR is proposed. For feature selection, Random Forest is employed, while PSO_SVR was used to predict the peak load. By checking the actual historical data of a county in Jiangxi Province, the strategy described in this study can improve the prediction accuracy of the prediction model. Because of feature selection, the prediction error of the prediction model and the complexity of the data set are reduced. In future research, we will continue to improve the model's prediction accuracy and reduce its complexity at the same time.

References

[1] Lahouar, A. and J. Ben Hadj Slama, Day-ahead load forecast using random forest and expert input selection. Energy Conversion and Management, 2015. 103: p. 1040-1051.

[2] Ding, J., et al., An integrated method based on relevance vector machine for short-term load forecasting. European Journal of Operational Research, 2020. 287(2): p. 497-510.
[3] Heydari, A., et al., Short-term electricity price and load forecasting in isolated power grids based on composite neural network and gravitational search optimization algorithm. Applied Energy, 2020. 277.
[4] Dai, Y. and P. Zhao, A hybrid load forecasting model based on support vector machine with intelligent methods for feature selection and parameter optimization. Applied Energy, 2020. 279.
[5] Pappas, S.S., et al., Electricity demand loads modeling using AutoRegressive Moving Average (ARMA) models. Energy, 2008. 33(9): p. 1353-1360.
[6] Lee, C.-M. and C.-N. Ko, Short-term load forecasting using lifting scheme and ARIMA models. Expert Systems with Applications, 2011. 38(5): p. 5902-5911.
[7] Chitalia, G., et al., Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks. Applied Energy, 2020. 278.
[8] Do, Q.H., Artificial neural network applications in business and engineering. 2020, Engineering Science Reference, an imprint of IGI Global,: Hershey, PA. p. 1 online resource.
[9] Barman, M. and N.B. Dev Choudhury, Season specific approach for short-term load forecasting based on hybrid FA-SVM and similarity concept. Energy, 2019. 174: p. 886-896.
[10] Deshwal V, Sharma M. Breast Cancer Detection using SVM Classifier with Grid Search Technique. International Journal of Computer Applications, 2019, 975: 8887.
[11] Li, Y., J. Che, and Y. Yang, Subsampled support vector regression ensemble for short term electric load forecasting. Energy, 2018. 164: p. 160-170.
[12] Selakov, A., et al., Hybrid PSO–SVM method for short-term load forecasting during periods with significant temperature variations in city of Burbank. Applied Soft Computing, 2014. 16: p. 80-88.
[13] Huang, N., G. Lu, and D. Xu, A Permutation Importance-Based Feature Selection Method for Short-Term Electricity Load Forecasting Using Random Forest. Energies, 2016. 9(10).
[14] Probst, P., M.N. Wright, and A.-L. Boulesteix, Hyperparameters and tuning strategies for random forest. WIREs: Data Mining & Knowledge Discovery, 2019. 9(3): p. N.PAG-N.PAG.
[15] Decision tree and ensemble learning based on ant colony optimization. 2018, New York, NY: Springer Berlin Heidelberg. pages cm.
[16] Criminisi, A. and J. Shotton, Decision forests for computer vision and medical image analysis. Advances in computer vision and pattern recognition,. 2013, London ; New York: Springer. xix, 368 pages.
[17] Fayed, H.A. and A.F. Atiya, Speed up grid-search for parameter selection of support vector machines. Applied Soft Computing, 2019. 80: p. 202-210.
[18] Chen, H., et al., Grid search parametric optimization for FT-NIR quantitative analysis of solid soluble content in strawberry samples. Vibrational Spectroscopy, 2018. 94: p. 7-15.

Huachao Zhai received a B. S. degree from Hebei University of Science and Technology in 2019. He is studying for a master's degree in Nanchang Institute of Technology.

Jinxing Che received his B. S. degree from Jiujiang University in 2007 and his M. S. degree in applied mathematics from Lanzhou University in 2010, as well as his Ph. D. degree in mathematical statistics from Xidian University, China in 2019. He is currently an associate professor and Master's Supervisor in College of Science, Nanchang Institute of Technology, China. His main research interest is data analysis theory and application, hydrological information processing as well as prediction theory and method.