Data Science in Finance and Economics

DSFE, 1(1): 1–20.
DOI: 10.3934/DSFE.2021001
Received: 18 April 2021
Accepted: 26 May 2021
Published: 01 June 2021

http://www.aimspress.com/journal/dsfe

Research article

Cross-validation research based on RBF-SVR model for stock index prediction

Feite Zhou*

Department of Applied Mathematics, The Hong Kong Polytechnic University, 11 Yuk Choi Road, Hung Hom, Hong Kong

* **Correspondence:** Email: jupiterzhou@foxmail.com.

Abstract: The ups and downs of stock indexes are one of the most concerned issues for investors in the stock market. To improve the accuracy of stock index prediction, this paper compares traditional K-Fold Cross-Validation (KCV) and three cross-validation methods in the Radial Basis Function Support Vector Regression (RBF-SVR) model. They are named as Abandon Tail Cross-Validation (ATCV), Sequential Division Cross-Validation (SCV) and Gap Sequential Division Cross-Validation (GSCV). It is found that KCV has very limited validation ability for stock indexes time series data with no certain relevance. However, SCV and GSCV with small gap perform better, with high accuracy about 88% and small error about 2%. This research shows that the establishment of time series forecasting models for stock indexes needs to pay more attention to cross-validation methods, which cannot randomly dividing training set and test set. It is strongly recommended to use SCV and GSCV instead of KCV. In addition, the choice of the penalty parameter C and the radial basis kernel function parameter γ largely determines the accuracy and reliability of RBF-SVR stock index prediction model.

Keywords: Support Vector Regression; parameter optimization; cross-validation; stock index prediction; time series

JEL Codes: C52

Abbreviations: RBF-SVR model: Radial Basis Function Support Vector Regression model; KCV: K-Fold Cross-Validation; ATCV: Abandon Tail Cross-Validation; SCV: Sequential Division Cross-Validation; GSCV: Gap Sequential Division Cross-Validation; GSCV-x%: Gap Sequential Division Cross-Validation

with the gap sized x%; SSE Composite Index: Shanghai Stock Exchange Composite Index; r^2: Determination Coefficient; RMSE: Root Mean Square Error; MAE: Mean Absolute Error

1. Introduction

The ups and downs of stock indexes are one of the most concerned issues for investors in the stock market, which is changing rapidly, and the risks and returns are also at a relatively high level. For investors, predicting stock indexes can help them choose the best investment opportunity to obtain the highest benefits. However, in most cases, stock indexes as a kind of time series data are often accompanied by small samples, non-linear and unstable characteristics, making the data have different manifestations at each stage. (Nivetha and Dhaya, 2017) This is the reason why stock index prediction is a difficult and challenging task in the financial market.

Using the methods of Machine Learning and Deep Learning to predict stock indexes or stocks has become the norm. Generally, researchers make predictions through the following steps: (1) data acquisition and preprocessing; (2) model selection and establishment; (3) parameter optimization and model evaluation; (4) result output. It is worth noting that some research of stock prediction has not given the results of using real set. Instead, they give the results between the model and the test set, which may let the result look like a good result for prediction but unknown in real set.

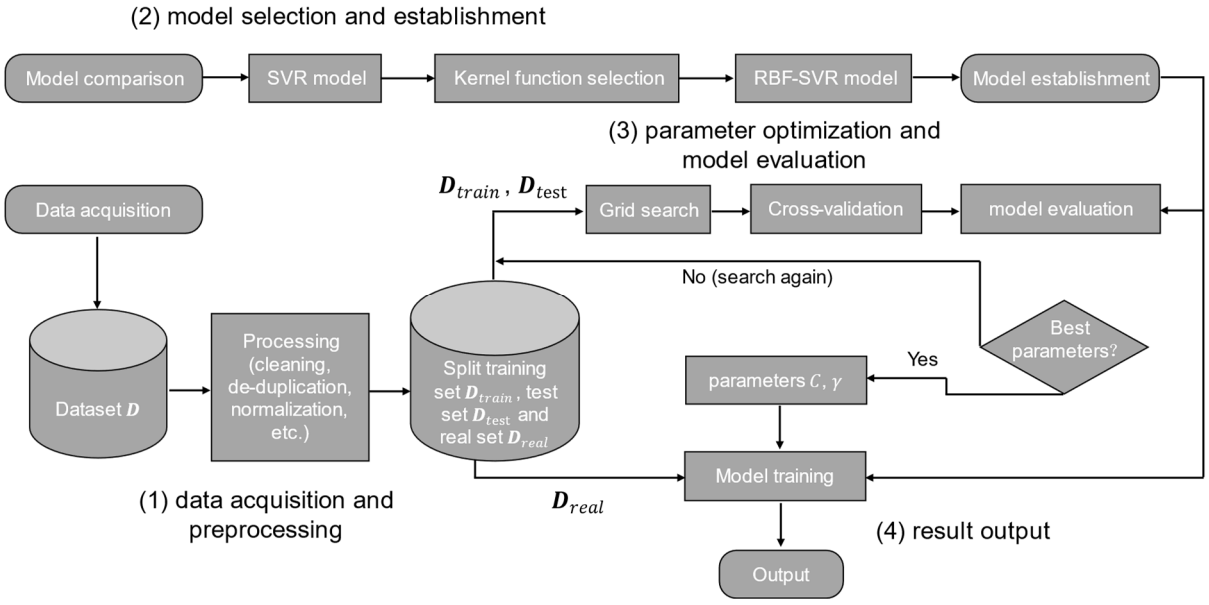

Figure 1. The flow diagram of RBF-SVR prediction model.

1.1. Model selection of support vector machine

Nowadays, in the field of stock and stock index forecasting, the three commonly used models are traditional time series models, support vector machine (SVM), and neural network models. This section will give a brief introduction to these three types of models and the reason of choosing SVM.

In the forecasting of stock indexes, researchers often cannot use traditional time series models to obtain better results (Makridakis and Hibon, 2000). Researchers usually make forecasts by annual,

monthly, or daily transaction data, which makes the time series data of stock indexes have these characteristics, including instability, small dataset, lots of potential information (Nivetha and Dhaya, 2017). It is difficult to build a suitable model. The analysis steps of the traditional time series model are relatively simple. Only if the training sample size is large enough and the data change trend is stable can more accurate prediction results be obtained. A typical model used for stock prediction is the Autoregressive Moving Average Model (ARMA) (Karanasos, 2001; Rojas et al., 2008), which is composed of the Autoregressive Model (AR) (Narayan, 2006; Vasyl et al., 2012) and Moving Average Model (MA) (Wei et al., 2014). In order to deal with the instability of time series data, based on the above models, the Autoregressive Integrated Moving Average Model (ARIMA) was proposed and used in stock forecasting (Karanasos, 2003; Pai and Lin, 2005; Jarrett and Kyper, 2011; Adebiyi et al., 2014). The research of Pai and Lin (2005) shows that using ARIMA alone is difficult to model, difficult to converge, and easy to fall into local minimums. Additionally, Jarrrtt and Kyper (2011) has proved that ARIMA model has a great improvement in the accuracy of predicting unstable time series, but its effect is still not good for non-linear stock index prediction.

Compared with traditional time series models above, SVM has better single-model forecasting ability for time series data (Thissen et al., 2003; Sapankevych and Sankar, 2009). SVM is a machine learning method based on the Vapnik-Chervonenkis Dimension and Structural Risk Minimization (Cortes and Vapnik, 1995). It can effectively solve small sample, nonlinear, and high-dimensional problems, such as the problems of stock index prediction (Kara, 2011; Lin, 2013; Wen, 2014; Devi, 2015; Heo, 2016). SVM was officially released by Cortes and Vapnik (1995). With its outstanding performance in text classification tasks, it quickly became the mainstream technology of machine learning. SVM can be divided into Support Vector Classification (SVC) and Support Vector Regression (SVR). The SVR model can be used for time series forecasting, nonlinear modeling and forecasting, optimization control, etc.

Neural network models take longer time and larger computing resources to predict stocks and get relatively better model accuracy (Naeini et al., 2010). Common models are Multi-Layer Perceptron (MLP) (Turchenko et al., 2011), Convolutional Neural Network (CNN) (Selvin et al., 2017), Recurrent Neural Network (RNN) (Selvin et al., 2017; Achkar et al., 2018), Long Short-Term Memory (LSTM) (Selvin et al., 2017; Karmiani et al., 2019). Based on the above papers, it is found that LSTM perform best in stocks prediction. LSTM, proposed by Hochreiter and Schmidhuber (1997), is a kind of RNN that is suitable for processing and predicting important events with relatively long intervals and delays in time series. The difference between LSTM and RNN is that LSTM adds a processor, called cell, to the algorithm to determine whether the information is useful or not. Although it has a certain improvement in the predictive ability of stocks and stock indexes, the time it takes is nearly several times that of the SVM. Furthermore, as the number of cycles increases, LSTM will take more and more time than SVM, but the model accuracy is very close to SVM (Karmiani et al., 2019).

In conclusion, considering calculation time, model accuracy and a single non-combined model, this paper will use SVR model to predict stock indexes instead of AMIRA and LSTM model.

1.2. Parameter optimization in Support Vector Regression

Parameter optimization has become one of the necessary model establishment processes for stock and stock index forecasting. However, in SVR model, there is still much room for improvement in the parameter optimization method, especially in the cross-validation method.

To establish the best model, the parameters of models need to be optimized. In the Radial Basis Function Support Vector Regression model (RBF-SVR model), the parameters optimized are the penalty parameter C and the radial basis kernel function parameter γ. The smaller parameter C and γ can make the model smoother and have stronger generalization ability (Huang et al., 2004; Liu and Du, 2015). The way to select the best parameters is mainly to evaluate the accuracy and reliability of the model through different model evaluation methods (Bergmeir and Benítez, 2012). For accuracy, Pearson correlation coefficient and the determination coefficient (r^2) can be used to evaluation models. For reliability, researchers usually use errors to evaluate models, including Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), etc. Different parameters have a great impact on the prediction ability of the model (Han et al., 2012; Li et al., 2021), and they are difficult to calculate or estimate directly from the dataset. Therefore, many parameter optimization methods are proposed to find the best combination of parameters.

Parameter optimization methods can be roughly divided into parameter search methods and validation methods. Parameter search methods include Grid Search Algorithm (Syarif et al., 2016), Genetic Algorithm (GA) (Syarif, 2016), Particle Swarm Optimization (PSO) (Huang, 2008; Salam, 2015) and so on. Among them, the Grid Search method has some advantages for the accuracy and reliability of models, but it takes too much time for computation (Syarif et al., 2016). In this regard, Liu and Du (2015) proposed a strategy to gradually optimize parameters based on cross-validation accuracy contour maps, which can both ensure the accuracy and establishment speed of the SVR model. As for validation methods, researchers usually use cross-validation, Hold-Out, etc. (Bergmeir and Benítez, 2012), such as K-Fold Cross-Validation (KCV) and some combination methods (Zhu et al., 2021). The basic idea of the cross-validation method is to randomly divide dataset into a training set and a test set, then exchange them to validate, and finally obtain the best model based on the average value of the model accuracy or errors.

Due to the continuity of time series data, the cross-validation method of random partitioning cannot get an effective SVR model (Bergmeir at al., 2018). At present, most researchers who use SVR model to predict stock indexes do not pay enough attention to cross-validation methods. Cross-validation allows all available data to be used for training and testing. It realizes the diversity and adequacy of model evaluation, improving the accuracy of the model. But the time series may be generated by a process that evolves over time, which undermines the basic assumption of cross-validation that the data are independent and identically distributed (i.i.d.) (Bergmeir and Benítez, 2012).

In terms of stock index forecasting, traditional cross-validation methods cannot effectively improve the accuracy of the model, owing to data relevance and instability and the fact that investors often use models to predict out-of-sample data. For example, the SVR model based on KCV has only 61%–75% model accuracy in the study of the team of Yu (2014). Researchers often assume that the predicted value is related to the trading conditions of the previous trading day (Huang et al., 2004). If the cross-validation method of random partitioning is used, it is easy to use future data to predict the past data. Such cross-validation results lack practical significance in stock index prediction and run counter to the goal of predicting the future.

In conclusion, to help investors and researchers get better stock and stock index forecasting results, this paper proposes three cross-validation methods for stock index prediction based on the research of Bergmeir (2012). The three methods are Abandon Tail Cross-Validation (ATCV), Sequential Division Cross-Validation (SCV) and Gap Sequential Division Cross-Validation (GSCV), introduced in section 3.1.

1.3. Summary and guidelines

In summary, this research will take the RBF-SVR model of predicting the Shanghai Stock Exchange (SSE) Composite Index as an example to compare KCV, ATCV, SCV and GSCV. Grid Search Algorithm will be used for searching the best penalty parameter C and radial basis kernel function parameter γ, which are evaluated by r^2, RMSE and MAE. The speed of searching and the accuracy and errors of models are used to find a relatively excellent cross-validation method for stock index prediction.

In Section 2, it gives an overview of the mathematical theory of the RBF-SVR model building process. In Section 3, it gives a theoretical overview of parameter optimization methods, in which cross-validation methods are given in Section 3.1 including KCV, ATCV, SCV and GSCV, the model evaluation indexes such as r^2, RMSE and MAE are given in Section 3.2, and the principle and process of Grid Search Algorithm are given in Section 3.3. In Section 4, it gives the actual modeling process and result analysis, using RBF-SVR model to predict Shanghai Stock Exchange Composite Index (SSE Composite Index). Finally, a conclusion of the paper is given in Section 5.

2. Mathematical theory of the RBF-SVR model establishment

To solve the problem of stock index prediction with the small sample, non-linear and unstable nature of data, this research adopts the RBF-SVR model. SVR is a regression model algorithm in SVM, which can fully use limited sample information, and search the best balance between the complexity of the model and the learning ability to obtain the best generalization ability. So far, there have been many studies that have proved the considerable application prospects of SVR in stock indexes and stocks forecasting. Hui and Wu (2012) used SVR to improve the simple moving average trading system and obtained good returns. Choudhury (2014) formulated short-term stock trading strategies by SVR and obtained good prediction results.

For a given data training set $\boldsymbol{D}_{train} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_m, y_m)\}$ where $\boldsymbol{x}_i \in \mathbb{R}^n, y_i \in \mathbb{R}$, the ϵ-insensitive loss function $\ell_\epsilon(x)$, the penalty parameter C, linear regression model $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$, in which $\boldsymbol{w}$ and b are undetermined, it can set up ϵ-SVR model, which is relatively robust regression (Yu et al., 2014). To make y and $f(x)$ as close as possible, the SVR problem can be formalized as:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m} \ell_\epsilon(f(x_i) - y_i) \tag{1}$$

where ϵ-insensitive loss function is:

$$\ell_\epsilon(x) = \begin{cases} 0 & if\ |x| \le \epsilon \\ |x| - \epsilon & otherwise \end{cases} \tag{2}$$

This means that in SVR, an interval of 2ϵ is constructed for the regression line $f(x)$ through the insensitive loss function $\ell_\epsilon(x)$, where the tolerable error is ϵ. The data points in the interval band are considered to have no loss, and only the data points outside the interval band will calculate the loss. The interval band is shown below.

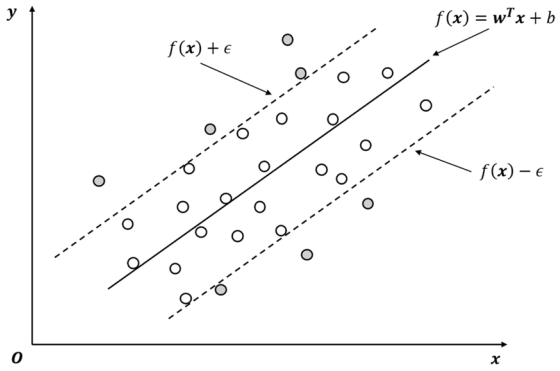

Figure 2. The interval band of SVR model.

Then, introduce slack variables ξ_i and $\hat{\xi}_i$ for each data point. If the data point is above the interval band, the loss is recorded as ξ_i and below, the loss is recorded as $\hat{\xi}_i$. Only one of these two slack variables can be established, which means there is only one non-zero value because data point is either above or below the regression model $f(\boldsymbol{x})$. If it happens to be on the boundary, ξ_i and $\hat{\xi}_i$ are both 0. After introducing the slack variables ξ_i and $\hat{\xi}_i$ into the basic Equation (1), the general form of the SVR mathematical model is:

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\left(\xi_i + \hat{\xi}_i\right) \\ &s.t.\ \ f(\boldsymbol{x}_i) - y_i \le \epsilon + \xi_i \\ &y_i - f(\boldsymbol{x}_i) \le \epsilon + \hat{\xi}_i \\ &\hat{\xi}_i, \xi_i \ge 0, i = 1,2,..,m. \end{aligned} \tag{3}$$

By Lagrangian Multiplier Method, Equation (3) can be converted into Lagrangian function with Lagrangian multiplier $\hat{\alpha}_i, \alpha_i, \hat{\mu}_i, \mu_i \ge 0$ as following.

$$\begin{aligned} L\left(\boldsymbol{w}, b, \hat{\boldsymbol{\alpha}}, \boldsymbol{\alpha}, \hat{\boldsymbol{\mu}}, \boldsymbol{\mu}, \hat{\boldsymbol{\xi}}, \boldsymbol{\xi}\right) = &\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\left(\xi_i + \hat{\xi}_i\right) - \sum_{i=1}^{m}\mu_i\xi_i - \sum_{i=1}^{m}\hat{\mu}_i\hat{\xi}_i \\ &+ \sum_{i=1}^{m}\alpha_i(f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i) + \sum_{i=1}^{m}\hat{\alpha}_i(y_i - f(\boldsymbol{x}_i) - \epsilon - \hat{\xi}_i) \end{aligned} \tag{4}$$

Take the partial derivative of the variables $\boldsymbol{w}$ and b, and then set them equal to 0. It can get the relationship among $\hat{\alpha}_i, \alpha_i, \hat{\mu}_i, \mu_i$ and $C, \boldsymbol{w}$. Next, turn it into a dual problem of SVR:

$$\max_{\alpha,\hat{\alpha}} \sum_{i=1}^{m} y_i(\hat{\alpha}_i - \alpha_i) - \epsilon(\hat{\alpha}_i - \alpha_i) - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\hat{\alpha}_i - \alpha_i)\left(\hat{\alpha}_j - \alpha_j\right)\boldsymbol{x}_i^T\boldsymbol{x}_j \tag{5}$$

$$s.t.\sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i) = 0 \; and \; 0 \le \hat{\alpha}_i, \alpha_i \le C$$

The optimization problem of Equation (5) needs to satisfy the KKT condition, and the solution can be used to obtain $\hat{\boldsymbol{\alpha}}, \boldsymbol{\alpha}$. If the sample $(\boldsymbol{x}_i, y_i)$ does not fall into the interval band, the corresponding $\hat{\alpha}_i, \alpha_i$ can obtain a non-zero value, but at least one of them is zero. The points falling outside the interval band with non- zero $\hat{\alpha}_i, \alpha_i$ are the support vectors of SVR model.

Using the partial derivative of Equation (4) and results, the SVR model is finally given as:

$$f(\boldsymbol{x}) = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\boldsymbol{x}_i^T\boldsymbol{x} + b \tag{6}$$

Since the stock index prediction is a nonlinear and inseparable problem, it is necessary to introduce a kernel function to transform the data at low-dimensional space into high-dimensional space, which can effectively overcome the dimensionality disaster and improve the generalization ability of the model (Li et al., 2010).

The Radial Basis Kernel Function (RBF) is a universal kernel function. By choosing appropriate parameters, it can map arbitrary distribution samples (Han et al., 2012). In SVR model, RBF is used to replace the inner product operation of the original space to achieve dimensional mapping, which solves the nonlinear regression problem.

Compared with RBF, the Polynomial Kernel Function not only greatly increases the amount of calculation in the case of high-dimensional data, but also prone to overfitting (Lee, 2009; Marković et al., 2017). Moreover, RBF has only one parameter γ to be optimized, but the polynomial kernel function needs to optimize multiple parameters such as dimensional parameters and constant parameters at the same time. Therefore, this paper chooses the radial basis kernel function for stock index forecasting.

The RBF is given as below:

$$\kappa(\boldsymbol{x_i}, \boldsymbol{x_j}) = \exp\left(-\gamma\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2\right) \tag{7}$$

Using RBF of Equation (7) into Equation (6) instead of inner product can get the final RBF-SVR model:

$$f(\boldsymbol{x}) = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\kappa(\boldsymbol{x_i}, \boldsymbol{x_j}) + b \tag{8}$$

Finally, the RBF-SVR model is acquired as Equation (8).

The final parameters that need to be optimized are the penalty parameter C and the RBF parameter γ due to Equations (3) and (7). The penalty parameter C is used to penalize the slack variables ξ_i and $\hat{\xi}_i$. The larger the penalty parameter C is, the smaller the slack variables are, which means the model tends to correctly classify all training samples, but the generalization ability of the model becomes weaker. A smaller parameter C will make the model smoother and have a stronger generalization ability. However, too large parameter C will make the model easily overfit and too small parameter C will make the model easily underfit.

The parameter γ of RBF defines how much influence a single training sample can have. It is related to the inverse of the variance. A smaller parameter γ means that a single sample has a larger influence, while a larger parameter γ means that a single sample has a smaller influence. Therefore, the parameter γ can be regarded as the width parameter of the kernel function, which controls the radial range of the kernel function. In general, the smaller the parameter γ, the smoother the model.

3. Parameter optimization methods of grid search and cross-validation

SVM is a supervised learning method hence the parameters of the model need to be manually selected and optimized. In actual cases, there is no theoretical guidance for feature selection, kernel function selection, kernel function parameters, penalty parameter, and insensitive loss function parameters. It leads to the effect less than expected in comparison research (Karmiani, 2019).

To find the optimal parameters of the RBF-SVR model for stock index prediction, this paper starts with the validation methods for parameter optimization based on Grid Search Algorithm. Validation methods include cross-validation, Hold-Out, etc. In practice, grid search and cross-validation methods are generally used to determine parameters. Although this method is simple, relies too much on the designer's experience, lacks a theoretical basis, and the results obtained are unstable, it can give the best parameters compared with Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) (Salam, 2015).

3.1. K-Fold Cross-Validation and optimization methods

After Stone (1974) proposed the cross-validation method, scholars from various countries have proposed a variety of cross-validation methods such as KCV, Hold-Out (Yadav and Shukla, 2016), Leave-One-Out (Cawley, 2016) and Bootstrapping (Zhou, 2016).

KCV is one of the most commonly used cross-validation methods. The basic idea is to randomly divide the original data set into K groups, where the K–1 group is the training set, and the remaining 1 group is the test set, also known as the validation set. Then, it uses the training set to train the model, and use the test set to evaluate the trained model. After that, it repeatedly evaluates the model training performance of each group. Finally, the average performance of all group is given as an indicator. K always equals to 5 or 10, which means divide the original data set into 5 or 10 groups.

However, in terms of forecasting stocks and stock indexes, KCV often cannot get a well-performing model. Because the data of stocks and stock indexes are always sequential correlated and full of potential information. Furthermore, investors often use models to predict out-of-sample data. These reasons reduce the verification ability of KCV. For instance, it has only 61%–75% model accuracy in the research of the team of Yu (2014). In the research of Karmiani (2019), the SVR model based on KCV only has 67% model accuracy.

If the cross-validation method of random partitioning is used, like KCV, it is easy to use future data to predict the past data. Such cross-validation results lack practical significance in stock index prediction and run counter to the goal of predicting the future. Therefore, this paper will use the following cross-validation methods based on the research of Bergmeir (2012) and apply them to stock index forecasting, comparing with KCV to find the best parameter cross-validation method. They are ATCV, SCV and GSCV.

3.1.1. Abandon Tail Cross-Validation

This method abandons part of the tail data, aiming at improving the ability of model to predict stock indexes in a short period of time. The research of Bergmeir (2012) shows that the effectiveness of this method for time series data cannot be proved, and there is also the disadvantage of losing potentially important information.

The basic idea is to divide original dataset into K groups in chronological order, where the first K–n groups are the training set, and the K–n+1-th group is the test set. After the evaluation is completed, the last one, No. K–n+1, is abandoned. Then, the first K–n-1 groups are selected as the training set, and the No. K–n group is the test set. Similarly, the average performance of all group is given as the indicator. Moreover, the training set should be greater than 30% of the total data set, to ensure the effectiveness of the model. The schematic diagram is as following.

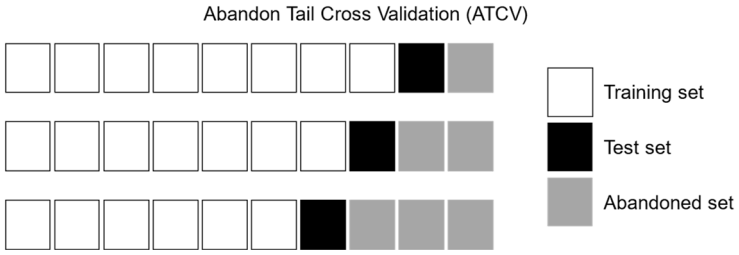

Figure 3. Abandon Tail Cross-Validation.

3.1.2. Sequential Division Cross-Validation

SCV is derived from the Forward Validation proposed by Hjort (1982). This method is more like a combination of KCV and Hold-Out method. On the premise of not abandoning data, it divides the original dataset into training set and test set for validation.

The basic idea is to divide original dataset into K groups in chronological order, where the first K–n groups are the training set, and the remaining groups are the test set. After the evaluation is completed, the first K–n-1 groups are selected as the training set. According to this loop, the average performance of all group is given as the indicator. Similarly, the training set should be greater than 30% of the total data set. The schematic diagram is as following.

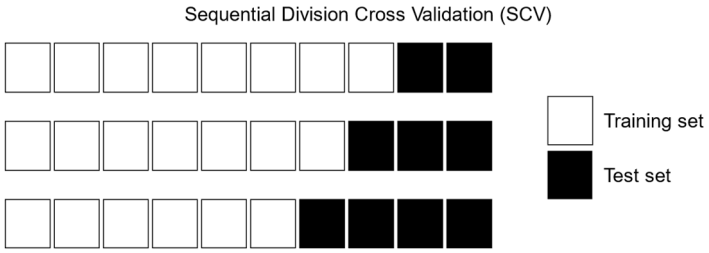

Figure 4. Sequential Division Cross-Validation.

3.1.3. Gap Sequential Division Cross-Validation

GSCV is based SCV, abandoning some of the original data between the test set and the training set before validation. When the gap of GSCV is 0, GSCV is the same as SCV. Its purpose is to improve the ability of model to predict the future and shorten the calculation time. Because in stock index forecasting, investors usually want to quickly predict the ups and downs of stock indexes of a relatively long period in the future.

The basic idea is to divide original dataset into K groups in chronological order, where the first K–n-m groups are the training set, the last n groups are the test set, and the m groups between them are abandoned, called gap. Gap needs to define its length m in advance. After the evaluation is completed, the first K–n-m-1 groups are selected as the training set, the last n+1 group are the test set, and the gap is still m. Similarly, the average performance of all group is given as the indicator and the training set should be greater than 30% of the total data set. The schematic diagram is as following.

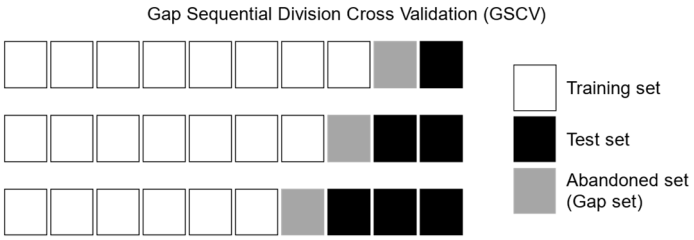

Figure 5. Gap Sequential Division Cross-Validation.

3.2. Model evaluation indexes

To evaluate the performance of the stock index forecasting model, researchers often use the accuracy or errors of prediction models. For accuracy, this paper will use the r^2. For errors, the RMSE and the MAE are selected for evaluation. The Equations are as following:

$$r^2 = 1 - \frac{\sum_{i=1}^{n}\left(\hat{x}_{i,open} - x_{i,open}\right)^2}{\sum_{i=1}^{n}\left(\hat{x}_{i,open} - \bar{x}_{i,open}\right)^2} \tag{9}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{x}_{i,open} - x_{i,open}\right)^2} \tag{10}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\left(\hat{x}_{i,open} - x_{i,open}\right)\right| \tag{11}$$

Among them, $x_{i,open}$ is the real opening price, $\hat{x}_{i,open}$ is the predicted opening price, $\bar{x}_{i,open}$ is the average of the opening prices, and n is the total numbers of data in the test set. In generally, as for r^2, the closer it is to 1, the higher the accuracy of the model is, and the closer it is to 0, the lower

the accuracy of the model. If it is a negative number, it means that there is little correlation between the model and the data set. As for RMSE and MAE, the values of them stand for the error percentage between actual value and forecasting value. RMSE and MAE are closer to 0, indicating that the error of the model is smaller.

3.3. Grid Search Algorithm

Grid Search Algorithm is a parametric search method combined with model evaluation indexes and cross-validation methods, which belongs to exhaustive search. The grid search cyclically uses each set of parameters to train the model and selects the optimal one with the smallest error or the highest accuracy of model. Generally, grid search will pick the appropriate value approximately on the logarithmic scale (Zhou, 2016) such as $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. The research of Devi (2015) shows that the accuracy of model is very high when the parameters C and γ are in a certain interval, but the accuracy in most ranges is very low. Therefore, researchers often use grid search repeatedly to find the local optimal parameter interval, and then find the global optimal parameter. In addition, for the RBF-SVR model, grid search tends to choose a smaller penalty parameter C when the corresponding model accuracy or error is equal, to improve the generalization ability of model.

The steps of a grid search method used in this article is given below:

1. Define search times n and the initial search set $V_0 = \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$.
2. Select model evaluation indexes among r^2, $RMSE$ or MAE.
3. Select value from V_i for parameters C and γ to train RBF-SVR model.
4. Use cross-validation methods to get the mean of accuracy or error.
5. Compare the mean of accuracy or error to output the local optimal parameters $x = \{C_i, \gamma_i\}$
6. Define the next search set as $V_i = [x_{i-1} * 0.5, x_{i-1} * 1.5)\ with\ step\ x * 0.1$.
7. Repeat steps 2–6 until $i = n$ or $x = x_{i-1}$

The algorithm verifies only 11 candidate parameters sets in each loop, equals 121 candidate parameters for C and γ. The accuracy of model and the computing time increase when search times n increases at the same time. For example, after obtaining the local optimal value x_0 through V_0, the algorithm will search in the nearest range of x. If $x_0 = 100$, then $V_1 = [50,150]$ and step size is 10. The algorithm will repeat the search until the number of searches reaches n.

Although this method reduces a lot of search time, it is worth noting that this type of simplified grid search often obtains local optimal parameters, but not global optimal parameters.

4. SSE Composite Index forecasting and results analysis

4.1. Data acquisition and preprocessing

The data for model training comes from the TUSHARE website, whose original data source is Sina Finance, including 243 trading days data of the Shanghai Stock Exchange (SSE) Composite Index from January 1, 2020 to December 31, 2020, used for training set $\boldsymbol{D}_{train}$ and test set $\boldsymbol{D}_{test}$. The analysis data includes six features, including opening price, closing price, highest price, lowest price, trading volume, and trading turnover. It is assuming that the daily opening price of the SSE Composite Index is related to the trading conditions of the previous trading days. Although the purpose of this research is to study the effect of the cross-validation methods rather than the predictive ability of the

RBF-SVR model, the data of 30 trading days after December 31, 2020 are selected as the real set $\boldsymbol{D_{real}}$ for testing the actual predictive ability of RBF-SVR model.

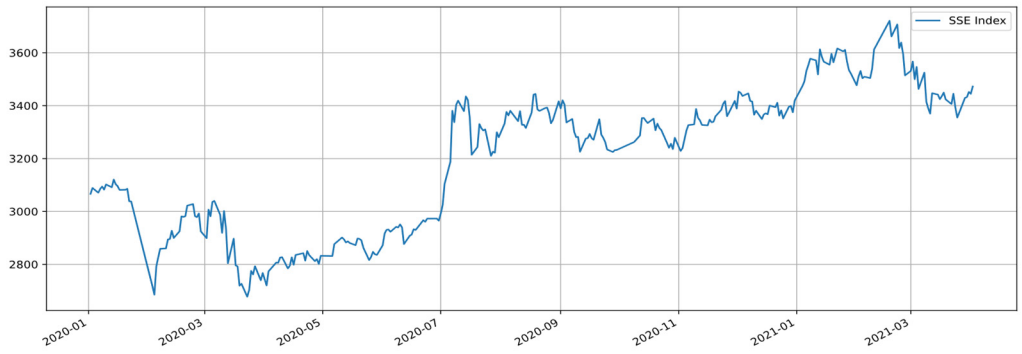

Figure 6. SSE Composite Index from January 1, 2020 to April 2, 2020.

Affected by COVID-19, the SSE Composite Index has been in a sluggish performance from February to July 2020. It had a sudden upturn in July, which made index forecasting more difficult. Although the index has certain fluctuations, compared with the financial crisis, these fluctuations are still within an acceptable range. The overall trend of data is in an upward state, which is more in line with linear regression models, including the RBF-SVR model.

Because the data on December 31, 2020 lacks the data of the next day, it is not convenient to test the model, so it is deleted. There are totally 242 pieces of records. Let the total set of data be:

$$\boldsymbol{D} = \{\boldsymbol{Y}, \boldsymbol{D}_o, \boldsymbol{D}_c, \boldsymbol{D}_h, \boldsymbol{D}_l, \boldsymbol{D}_v, \boldsymbol{D}_t\}, \boldsymbol{D} \epsilon \mathbb{R}^{m \times n} \tag{12}$$

where $\boldsymbol{Y}$ is the opening price, $\boldsymbol{D}_o, \boldsymbol{D}_c, \boldsymbol{D}_h, \boldsymbol{D}_l, \boldsymbol{D}_v, \boldsymbol{D}_t$ are respectively the opening price, closing price, highest price, lowest price, trading volume, and trading turnover of the previous trading day. The row vectors of $\boldsymbol{D}$ is the data of each trading days and is arranged in chronological order.

To prevent data overflow and the large value affecting the small value, the dataset $\boldsymbol{D}$ is normalized according to the following Equation:

$$\boldsymbol{x}_i = \frac{\boldsymbol{x}_i - \boldsymbol{x}_{i,min}}{x_{max} - x_{min}}, i = 0,1, \ldots, 6, \boldsymbol{x}_i, \overline{\boldsymbol{x}}_i \epsilon \mathbb{R}^n \tag{13}$$

where $\boldsymbol{x}_i$ is the i-th column vector of the data set D, also the i-th feature, $\boldsymbol{x}_{i,min}$ is a column vector whose elements are all x_{min}, whose length the same as $\boldsymbol{x}_i$, and x_{max} and x_{min} are respectively the maximum and minimum values in $\boldsymbol{x}_i$.

4.2. Parameter optimization and model evaluation

There are totally 242 pieces of trading data. Select the data of 30 days, a month, as the test set $\boldsymbol{D}_{test}$, and the remaining as the training set $\boldsymbol{D}_{train}$ for parameter optimization. 212 pieces of data are used to train model and 30 pieces of data are used for testing model. Moreover, this research will set $\boldsymbol{D}_{test}$ as 60, 30, 14, 7 days to compare different cross-validation methods.

According to the research of Roberts (2017), the set for parameter optimization is divided into K = 40, 20, 10, 5 nearly equal parts, of which the first parts contain more pieces of data,

making the data set $\boldsymbol{D}$ approximately equally divided. The research divides the data set into 40 equal parts, among which the first 2 parts contain 7 pieces of data, and the remaining 38 pieces contain only 6 pieces of data. Generally, the larger the K value, the more training samples used, and the more accurate the model. Therefore, it set the default K value as 40.

Use the scikit-learn package of Python 3.8.5 to build the RBF-SVR model, and the default memory size is 200M. Relying on different evaluation indexes, using Grid Search Algorithm, selecting KCV, ATCV, SCV and GSCV with different gaps, get the best parameters to establish model. Finally, output the results of r^2, RMSE, MAE, and the average computing time of each loop of Grid Search to compare.

4.3. Results output and analysis

The r^2, RMSE, MAE in this section only represents the fitting effect of the RBF-SVR model and the test set $\boldsymbol{D}_{test}$ rather than the final predictive ability of the model. The model evaluation results obtained after 10 experiments are as follows, where the time T taken is the average of the 10 experiments. The time t is the average computing time of each loop of Grid Search.

$$t = \frac{T}{n} \tag{14}$$

where n is the searching times of Grid Search Algorithm in section 3.3 and n is defined the default value 100. T is the total time for searching the best parameters.

In section 4.3.1, it gives the comparison of different evaluation indexes, including r^2, RMSE, MAE. In section 4.3.2, it gives the influence of different K value on different cross-validation methods based on MAE. In section 4.3.3, it gives the comparison of different gaps in GSCV. In section 4.3.4, it gives the performance of different cross-validation methods at different fitting period of $\boldsymbol{D}_{test}$. In section 4.3.5, it gives the comparison of computing time of different cross-validation methods. In section 4.3.6, it gives the performance of different cross-validation methods at different training period of $\boldsymbol{D}_{train}$.

4.3.1. Comparison of evaluation indexes

To find the best evaluation index to describe the average performance of all group, r^2, RMSE, MAE are chosen and used in KCV, ATCV, SCV and GSCV. These evaluation indexes are correlated but they have different effects in cross-validation methods. Generally, the larger r^2 is, the smaller RMSE and MAE are. Therefore, r^2 is chosen to describe $\boldsymbol{D}_{test}$. r^2, RMSE, MAE are used in cross-validation process.

Set the K value as 40, the gap of GSCV as 4 (10%) and fitting period of $\boldsymbol{D}_{test}$ as 60 days.

In summary, for evaluation indexes, using MAE can get better model than using model accuracy r^2. For cross-validation methods, SCV and GSCV-10% has better average performance, but the performance of KCV is far worse than the other three.

Table 1. Comparison of evaluation indexes.

	Based on r^2	Based on RMSE	Based on MAE
KCV	−0.005982	−0.030946	0.051152
ATCV	0.468106	0.898694	0.901852
SCV	0.898694	0.689813	0.898694
GSCV-10%	0.895556	0.689813	0.895556

4.3.2. Comparison of K value

To find the best K value for different cross-validation methods, MAE are chosen and used in KCV, ATCV, SCV and GSCV. Generally, K value equals 5, 10, 20. As the dataset have totally 242 pieces, the value 40 is added for comparison. Similarly, r^2 is chosen to describe $\boldsymbol{D}_{test}$.

Set the evaluation index as MAE, the gap of GSCV as 20%, that is gap = 1 when K = 5, gap = 2 when K = 10, and fitting period of $\boldsymbol{D}_{test}$ as 60 days.

Table 2. Comparison of K value.

	K = 5	K = 10	K = 20	K = 40
KCV	−0.001939	0.197782	−0.000506	0.051152
ATCV	0.720957	0.865332	0.901852	0.901852
SCV	0.723636	0.898112	0.720957	0.898694
GSCV-20%	0.723636	0.723245	0.723245	0.723245

In summary, for K value, relatively large K value can help ATCV, SCV get better results. Too little K value may make cross-validation methods have great randomness and the result unstable. As for GSCV, K value has little impact on the results if the percentage of gap is constant. For cross-validation methods, ATCV has the best average regardless of the K value but KCV has the worse one.

4.3.3. Comparison of gaps in GSCV

To find the influence of different gaps on GSCV, gaps are set as 2, 4, 8, 16 (5%, 10%, 20%, 40%). Similarly, r^2 is chosen to describe $\boldsymbol{D}_{test}$. Time t is used to evaluate the speed of GSCV at different gaps. Set the evaluation index as MAE, the K value as 40, and fitting period of $\boldsymbol{D}_{test}$ as 60 days.

Table 3. Comparison of gaps in GSCV.

	Model r^2	Time t (s/loop)
GSCV-5%	0.898694	2.284
GSCV-10%	0.895555	2.054
GSCV-20%	0.723245	1.599
GSCV-40%	0.717929	0 .812

In summary, the gaps can influence the computing speed and model accuracy. The larger the gap is, the faster the GSCV is, but too large gap will lead to lower accuracy. The gap near 10% is suggested to use for speeding up the cross-validation process with relatively high model accuracy.

4.3.4. Comparison of fitting period of $\boldsymbol{D}_{test}$

To find the performance of different cross-validation methods at different fitting period of $\boldsymbol{D}_{test}$, MAE are chosen and used in KCV, ATCV, SCV and GSCV. Generally, investors want to get the stock index trend of one week, two weeks, one month, and two months, hence the fitting periods of $\boldsymbol{D}_{test}$ are set as 7, 14, 30, and 60 days to compare. Similarly, r^2 is chosen to describe $\boldsymbol{D}_{test}$.

Set the evaluation index as MAE, the K value as 40, and the gap of GSCV as 4 (10%).

Table 4. Comparison of fitting period of $\boldsymbol{D}_{test}$.

	7	14	30	60
KCV	−0.606978	−0.951610	−0.193934	0.051152
ATCV	0.661665	0.760214	0.840177	0.901852
SCV	0.687866	0.780847	0.766305	0.898694
GSCV-10%	0.608619	0.714565	0.767680	0.895556

In summary, ATCV, SCV and GSCV-10% have better performance for long term fitting period than short term. Among them, ATCV performs better. KCV, as a cross-validation method of random segmentation, it has a certain degree of randomness in the accuracy of the model and cannot give a good time series model like stock index model.

4.3.5. Comparison of computing time

To find the speed of different cross-validation methods, r^2 is chosen to describe $\boldsymbol{D}_{test}$. Time t is used to evaluate the speed. Set the evaluation index as MAE, the K value as 40, and fitting period of $\boldsymbol{D}_{test}$ as 60 days, and the gap of GSCV as 4 (10%).

Table 5. Comparison of computing time.

	Model r^2	Time t (s/loop)	r^2/t
KCV	0.051152	3.335	0.015
ATCV	0.901852	2.484	0.363
SCV	0.898694	2.503	0.359
GSCV-10%	0.895556	1.937	0.462

In summary, for speed, GSCV-10% has the least computing time 1.937 s for each loop. As for efficiency, defined as r^2/t here, GSCV-10% also the highest efficiency.

4.3.6. Comparison of training period of $\boldsymbol{D}_{train}$

To find the performance of different cross-validation methods at different training period of $\boldsymbol{D}_{train}$. Different start data of training set are selected from 2017 to 2020. From March 1, 2020 to December 31, 2020, there are 206 trading days. From June 1, 2019 to December 31, 2020, there are 387 trading days.

MAE are chosen and used in KCV, ATCV, SCV and GSCV. The fitting periods of $\boldsymbol{D}_{test}$ are set as 60 days. Similarly, r^2 is chosen to describe $\boldsymbol{D}_{test}$.

Set the evaluation index as MAE, the K value as 40, and the gap of GSCV as 4 (10%).

Table 6. Comparison of training period of $\boldsymbol{D}_{train}$.

	June 1, 2019	September 1, 2019	March 1, 2020	January 1, 2020
KCV	−0.302232	−0.313440	−5.471353	0.051152
ATCV	0.281671	0.537941	0.326708	0.901852
SCV	0.315354	0.484126	0.327354	0.898694
GSCV-10%	0.276271	0.474042	0.325896	0.723245

In summary, ATCV, SCV and GSCV-10% have better performance than KCV. KCV has a certain degree of randomness in the accuracy of the model and cannot give a good time series model like stock index model.

4.4. Experiment conclusion and practical testing

Owing to the results of section 4.3, it can be found that:

•ATCV, SCV and GSCV with small gaps have good performance in model accuracy and error for time series data, such as stock indexes, while traditional KCV and GSCV with too large gaps are not suitable for stock index forecasting.

•For cross-validation methods of time series data, it is better to use errors as the model evaluation indicator such as MAE.

•Experiments have proved the continuity and correlation of time series data. The cross-validation methods of randomly dividing training set and test set, like KCV, is not suitable for time series data.

•The choice of the penalty parameter C and the radial basis kernel function parameter γ largely determines the accuracy and reliability of RBF-SVR stock index prediction model.

•For the RBF-SVR model, the selection of features is important. The features used in this research may not reflect most of the information of the stock index data, which makes the prediction results have large random fluctuations.

To verify the actual prediction ability of the cross-validation methods proposed in the research, the transaction data of 60 days in total from January 1, 2021 to April 2, 2021 is selected as the real set $\boldsymbol{D}_{real}$, which is the dataset not used in model establishment. To avoid a certain degree of randomness, select 7, 14, 30 days to compare. The 14-day data is from January 1, 2021 to January 21, 2021.The 30-day data is from January 1, 2021 to February 21, 2021.

Then, use KCV, ATCV, SCV, and GSCV with gap as 4 (10%) to establish model with K = 40 and evaluation index MAE.

Table 7. Comparison of practical testing.

	60	30	14	7
KCV	−0.146489	−2.259984	−2.451867	−1.958825
ATCV	−1.065538	−5.239521	−0.066538	0.296596
SCV	0.877276	0.844623	0.307411	−0.566581
GSCV-10%	0.877276	0.844623	−0.685385	−0.566581

In summary, SCV and GSCV-10% have a great model accuracy for long term stock index forecasting. ATCV are more suitable for short term forecasting but the model accuracy is unsatisfying. For KCV, it is not suitable for stock index forecasting. Their model diagram is shown below.

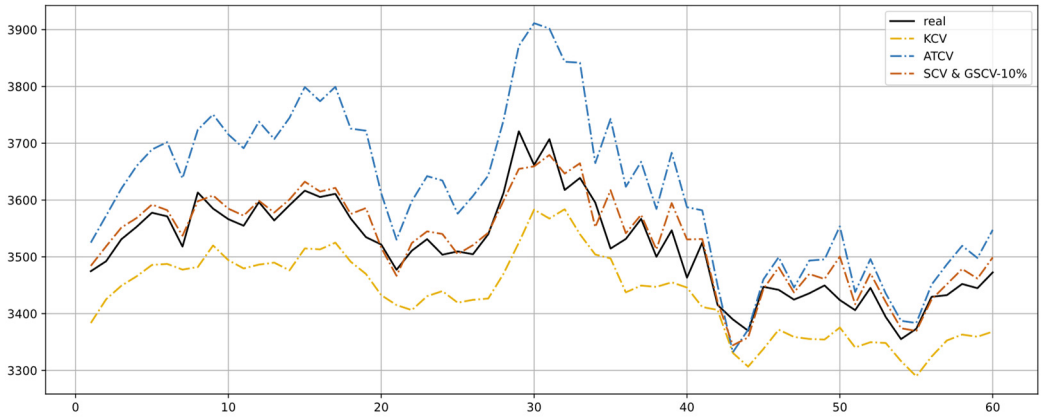

Figure 7. Model diagram of practical testing.

5. Discussion and conclusions

This paper selects 243 trading days data of the Shanghai Stock Exchange (SSE) Composite Index from January 1, 2020 to December 31, 2020 for research from TUSHARE website. Aiming at the small sample, non-linear and unstable nature of stock index data, through Radial Basis Kernel Function RBF-SVR model and the Grid Search Algorithm, three cross-validation methods proposed by the research are compared with traditional KCV in the field of predicting the SSE Composite Index. They are named as ATCV, SCV and GSCV, which are shown in section 3.1. Finally, it is found that SCV and GSCV are more suitable for long term stock and indexes forecasting with relatively high model accuracy.

With the help of various mathematical theories, the RBF-SVR model and its penalty parameter C and RBF parameter γ that need to be optimized are shown in section 2. The penalty parameter C is used to penalize the slack variables. The larger the C is, the more correctly the model classify all samples. The generalization ability of the model becomes weaker when C is too large or too small. A relatively smaller parameter C will make the model smoother and have a stronger generalization ability. As for the RBF parameter γ, it defines the influence of a single training sample, which can also be regarded as the width parameter of the kernel function. The smaller the parameter γ, the smoother the model.

From the perspective of the cross-validation methods, 4 different methods are used to compare the model accuracy and error (section 4.3.1), folds' value K (section 4.3.2), computing time of each loop (section 4.3.4), model fitting ability (section 4.3.5) and actual forecasting ability (section 4.4). It is found that using MAE, relatively large K value can be helpful for cross-validation to get a better model. Among KCV, ATCV, SCV, and GSCV, it can be concluded that: (1) KCV is not suitable for stock and indexes forecasting, whose most results of model accuracy r^2 are less than 0 or at a low level. (2) SCV and GSCV can be used for long term forecasting with a high model accuracy $r^2 \approx 88\%$ for 60-day prediction and $r^2 \approx 84\%$ for 30-day prediction. (3) Increasing the gap of GSCV can speed up calculation but reduce model accuracy, shown in section 4.3.3. It is recommended to set the gap as 10%. (4) For long term prediction, ATCV may cause the dataset to lose potentially important

information and result in poor prediction outcome. For short term prediction, ATCV performs better than SCV and GSCV, but the accuracy of the model obtained still needs to be improved.

Finally, it is recommended for researchers that when using the RBF-SVR model to predict financial time series data such as stock indexes or stocks, researchers pay more attention to cross-validation methods of parameter optimization, which cannot randomly dividing training set and test set. The choice of the penalty parameter C and the radial basis kernel function parameter γ largely determines the accuracy and reliability of RBF-SVR model. Based on MAE, Using SCV and GSCV with small gap that proposed in this research to replace the traditional cross-validation methods, KCV, can help researchers get better forecasting model.

In addition, although the real set ultimately used for prediction did not participate in model training, the features used still belong to historical data. It is the reason why model cannot truly predict the value of future stock indexes and it is also a developing direction for future work, but the model in this research can still provide some help for investors.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. I am very grateful to Professor Zhenghui Li from the Institute of Finance, Guangzhou University for his guidance and Bergmeir for their research on cross-validation methods for time series data.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

Achkar R, Elias-Sleiman F, Ezzidine H, et al. (2018) Comparison of BPA-MLP and LSTM-RNN for stocks prediction. 2018 6th International Symposium on Computational and Business Intelligence (ISCBI). 2018: 48–51.

Adebiyi AA, Adewumi AO, Ayo CK (2014) Stock price prediction using the ARIMA model. 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation. 2014: 106–112.

Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. *Inform Sci* 191: 192–213.

Bergmeir C, Hyndman RJ, Koo B (2018) A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Comput Stats Data Anal* 120: 70–83.

Cawley GC (2006) Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs. The 2006 IEEE International Joint Conference on Neural Network Proceedings. 2006: 1661–1668.

Choudhury S, Ghosh S, Bhattacharya A, et al. (2014) A real time clustering and SVM based price-volatility prediction for optimal trading strategy. *Neurocomputing* 131: 419–426.

Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20: 273–297.

Devi KN, Bhaskaran VM, Kumar GP (2015) Cuckoo optimized SVM for stock market prediction. 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). 2015: 1–5.

Han SJ, Cao QB, Han M (2012) Parameter selection in SVM with RBF kernel function. World Automation Congress 2012. 2012: 1–4.

Heo JY, Yang JY (2016) Stock price prediction based on financial statements using SVM. *Int J Hybri Inform Technol* 9: 57–66.

Hjort UH (1982) Model selection and forward validation. *Scand J Stat* 9: 95–105.

Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9: 1735–1780.

Huang CL, Dun JF (2008) A distributed PSO-SVM hybrid system with feature selection and parameter optimization. *Appl Soft Comput* 8: 1381–1391.

Huang W, Nakamori Y, Wang SY (2004) Forecasting stock market movement direction with support vector machine. *Comput Oper Res* 32: 2513–2522.

Hui X, Wu Y (2012) Research on simple moving average trading system based on SVM. 2012 International Conference on Management Science & Engineering 19th Annual Conference Proceedings. 2012: 1393–1397.

Goodfellow I (2016) Support vector machines, In: Goodfellow I, Bengio Y, Courville A, *Deep learning*, 2 Eds., The MIT Press, 88–90.

Jarrett JE, Kyper E (2011) ARIMA modeling with intervention to forecast and analyze Chinese stock prices. *Int J Eng Bus Manage* 3: 53–58.

Kara Y, Boyacioglu MA, Ömer KB (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul stock exchange. *Expert Sys Appl* 38: 5311–5319.

Karanasos M, Kim J (2003) Moments of the ARMA-EGARCH model. *Economet J* 6: 146–166.

Karmiani D, Kazi R, Nambisan A, et al. (2019) Comparison of predictive algorithms: backpropagation, SVM, LSTM and Kalman Filter for stock market. 2019 Amity International Conference on Artificial Intelligence (AICAI). 2019: 228–234.

Lee CM (2009) Using support vector machine with a hybrid feature selection method to the stock trend prediction. *Expert Syst Appl* 36: 10896–10904.

Li CH, Lin CT, Kuo BC, et al. (2010) An automatic method for selecting the parameter of the RBF kernel function to support vector machines. 2010 IEEE International Geoscience and Remote Sensing Symposium. 2010: 836–839.

Li ZR, Li CJ, Zhu H (2021) Parameter optimization of HVAC load forecasting model based on Support Vector Regression. *Build Energy Effic* 49: 43–48.

Lin Y, Guo H, Hu J (2013) An SVM-based approach for stock market trend prediction. 2013 International Joint Conference on Neural Networks (IJCNN). 2013: 1–7.

Liu Y, Du J (2015) Parameter optimization of the SVM for big data. 2015 8th International Symposium on Computational Intelligence and Design (ISCID). 2015: 341–344

Makridakis S, Hibon M (2000) The M3-Competition: results, conclusions, and implications. *Int J Forecast* 16: 451–476.

Marković I, Stojanović M, Stanković J, et al. (2017) Stock market trend prediction using AHP and weighted kernel LS-SVM. *Soft Comput* 21: 5387–5398.

Naeini MP, Taremian H, Hashemi HB (2010) Stock market value prediction using neural networks. 2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM). 2010: 132–136.

Narayan PK (2006). The behaviour of US stock prices: Evidence from a threshold autoregressive model. *Math Comput Simulat* 71: 103–108.

Nivetha RY, Dhaya C (2017) Developing a prediction model for stock analysis. 2017 International Conference on Technical Advancements in Computers and Communications (ICTACC). 2017: 1–3.

Pai PF, Lin CS (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega* 33: 497–505.

Roberts DR, Bahn V, Ciuti S, et al. (2017). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography* 40: 913–929.

Rojas I, Valenzuela O, Rojas F, et al. (2008) Soft-computing techniques and ARMA model for time series prediction. *Neurocomputing* 71: 519–537.

Salam MA (2015) Comparative study between FPA, BA, MCS, ABC, and PSO algorithms in training and optimizing of LS-SVM for stock market prediction. *Int J Adv Comput Res* 5: 35–45.

Sapankevych NI, Sankar R (2009) Time series prediction using support vector machines. *IEEE Comput Intell M* 4: 24–38.

Selvin S, Vinayakumar R, Gopalakrishnan EA, et al. (2017) Stock price prediction using LSTM, RNN and CNN-sliding window model. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2017: 1643–1647.

Stone M (1974) Cross-Validatory Choice and Assessment of Statistical Prediction. *J Roy Stat Soc* 36: 111–147.

Syarif I, Prugel-Bennett A, Wills G (2016) SVM parameter optimization using grid search and genetic algorithm to improve classification performance. *Telkomnika* 14: 1502–1509.

Thissen U, Van Brakel R, Weijer AP, et al. (2003) Using support vector machines for time series prediction. *Chemometr Intell Lab* 69: 35–49.

Turchenko V, Beraldi P, Simone FD, et al. (2011) Short-term stock price prediction using MLP in moving simulation mode. The 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems. 2011: 666–671.

Vasyl G, Bastian G, Roman L (2012) The conditional autoregressive Wishart model for multivariate stock market volatility. *J Econometrics* 167: 211–223.

Wei LY, Cheng CH, Wu HH (2014) A hybrid ANFIS based on n-period moving average model to forecast TAIEX stock. *Appl Soft Comput J* 19: 86–92.

Wen F, Xiao J, Zhifang HE, et al. (2014) Stock price prediction based on SSA and SVM. *Procedia Comput Sci* 31: 625–631.

Yadav S, Shukla S (2016) Analysis of K-Fold Cross-Validation over hold-out validation on colossal datasets for quality classification. 2016 IEEE 6th International Conference on Advanced Computing (IACC). 2016:7 8–83.

Yu HH, Chen RD, Zhang GP (2014) An SVM stock selection model within PCA. *Procedia Comput Sci* 31: 406–412.

Zhou ZH (2016) Support vector machines, In: Zhou ZH, *Machine learning*, Chinese version, 2 Eds., Bei Jing: The Tsinghua University Press, 121–145.

Zhu WG, Li YX, Yang WQ, et al. (2021). Short-term load forecast based on K-Fold Cross-Validation and stacking integration. *J Electr Pow Sci Technol* 36: 87–95.

AIMS Press