Stock Prices Forecasting and Optimization Strategies Based on Support Vector Machines

Junyuan Qiu[1,a], Jiahao Xu[2,b], Zirui Yu[3,c,*]

[1]*International Business College, South China Normal University, Foshan, Guangdong, 528225, China*
[2]*Dundee International Academy, Central South University, Changsha, Hunan, 410000, China*
[3]*Faculty of Information Science and Engineering, Ocean University of China, Qingdao, Shandong Province, 266404, China*
a. 20233637075@m.scnu.edu.cn, b. 2545204@dundee.ac.uk, c. yuzirui@stu.ouc.edu.cn
**corresponding author*

Abstract: With the global trend of digitization gaining prominence, the usage of machine learning methods such as Support Vector Machines and Reinforcement Learning for stock price prediction is becoming a hot topic. Over the past 40 years, China's economic market has undergone significant changes since the country's reform and opening up. In this study, the closing price and return of China's CSI 300 stock index are used as the database, and various data processing methods such as wavelet domain denoising, RSI screening and various SVM model optimization methods such as grid search and cross-validation are used to predict the upward or downward trend of stocks on the day after. The results of the study are presented by the model evaluation report and the heat map of the confusion matrix, which shows that the model prediction accuracy is 61% with the default parameters, and the accuracy improves to 67% after optimization. The results indicate that support vector machines are effective in stock price prediction, but there is still room for further improvement. This paper offers a potential approach that can increase return on investment and assist investors and financial institutions in making more informed investment decisions.

Keywords: SVM, Stock Prediction, Wavelet Domain Denoising, Parameter Optimization, Machine Learning

1. Introduction

Since China's reform and opening up, with the rapid development of the economy, the living standard of the Chinese people has improved significantly, and the investment field has been broadened. Stock investment has received widespread attention. Since the stock market was piloted in 1989, China's stock market has undergone radical changes after more than 30 years of development. As of September 2022, the number of listed companies in the A-share market reached 4, 911, with a total market capitalization of up to 82.58 trillion yuan and more than 200 million shareholders [1]. This indicates the strength of China's financial markets as well as the significance of the stock market to the country's economic structure. As markets continue to grow and investor structures become more sophisticated, the need for analyzing and forecasting the stock market grows.

In the field of stock investment, investors rely on two traditional analytical methods: technical analysis and fundamental analysis. Technical analysis provides decision support based on market behavior, while fundamental analysis focuses on company fundamentals. However, these methods require investors to have high theoretical knowledge and rich practical experience, and their prediction results are often highly subjective. Therefore, how to enhance the science and precision of stock prediction has become an important topic for research.

In recent years, with the development of econometrics and statistics, the time series forecasting method has been extensively utilized in stock price forecasting., which focuses more on predicting the future trend of stock prices through historical data. However, traditional forecasting models such as ARMA and GARCH have limitations in dealing with the complex and variable nonlinear characteristics of the stock market [2]. The SVMs, as powerful machine learning tools, have been proven to have excellent performance in many fields, especially in binary classification problems. However, SVMs need to be further optimized and improved because of the complexity of stock market prediction [3].

The purpose of this study is to explore the SVM-based stock price trend prediction method and its optimization strategy. Taking the CSI 300 index as an example, an improved SVM model is proposed through literature research, text mining, data processing, and model optimization. The study uses RSI and wavelet domain denoising to preprocess the data, optimizes the SVM parameters through grid search and cross-validation methods, and finally verifies the validity of the model by comparing the prediction precision before and after the model optimization. The innovation of this study lies in the comprehensive application of multiple data processing and model optimization techniques to enhance the precision of stock market forecasting and offer more scientific and reliable decision support to investors, thereby reducing investment risks and enhancing economic benefits.

2. Research Status

Vapnik proposed support vector machines in the 1990s, which are outstanding in solving small-sample, nonlinear, and high-dimensional pattern recognition, and can be generalized to problems such as function fitting [4]. Support vector machines take various feature parameters as support vectors, map them into a high-dimensional space using a kernel function, and differentiate the data by finding a partition plane. In stock prediction research, forecasters use data from several technical indicators to make predictions. A Chinese scholar named Yuchuan, Z. used support vector machines to accomplish this model of stock judgment through technical indicators and has achieved an accuracy rate of about 65% through experiments [5]. However, it was found that support vector machines are more difficult to solve quadratic optimization problems when the training set is larger. Lifang, P. used the data of Shahe stock as a sample and experimented with neural networks and time series as a comparison, which showed that the support vector machine prediction model obtains smaller computational error and a better prediction curve. Nevertheless, the basis of parameter selection was not pointed out [6]. Zhiyuan, H. has proposed a GA-SVM algorithm based on the improvement of AUC values to test Shanghai Pudong Development Bank in different time windows using 30 independent variable features. It was determined that the method is effective in short-term investment, but the threshold is too absolute when transforming the rise and fall for positive and negative samples [7]. Yibing, C. used a regression model based on an improved SVM to forecast the Chinese stock market index. She found that the model gives better results when the stock market is on a steady rise or fall, but neither the neural network nor the support vector machine fits well under abrupt change conditions [8]. The parameter selection is especially critical in how to use support vector machines. All of the above use the radial basis function as the kernel function. Cheng, L. tried to use the support vector machine with a wavelet kernel for stock index futures price prediction and found that it performs better than the ordinary kernel function. It is considered that it can be used to predict the closing price of the main contract of stock index futures except for a special case of CSI 500 [9]. The study is mainly a comparison of kernel function selection and does not compare with other algorithms.

3. Support Vector Machine

3.1. Introduction of the SVM Concept

The SVM is a common machine learning algorithm, which is not only suitable for classification problems but also can be predicted. The algorithm converts the low-dimensional linear inseparable space into high-dimensional linear separable space, establishes an optimal decision hyperplane, divides the sample into two parts, and maximizes the distance between the two nearest samples of these two separation planes. Its advantage is that it is applicable regardless of whether the sample is linear separable, approximately linear separable, or nonlinear separable, and has a high accuracy [10].

3.2. Introduction of Common Kernel Functions of SVMs

(Xi and Xj in the following formula represent the feature vectors of two input samples).

Linear kernel functions: directly divide in the original feature space, do not make any transformation to the data, and the calculation formula is:

$$K(X_i, X_j) = X_i^T X_j \tag{1}$$

Polynomial kernel function: maps data to the upper air for classification, which is suitable for orthogonal normalized datasets, and is calculated as follows [10]:

$$K(X_i, X_j) = \left(\theta + \gamma X_i^T X_j\right)^d \ , d \geq 1 \tag{2}$$

Where θ is a constant term, γ scales the inner product, and d is the number of polynomials. Gaussian kernel function: It maps data into infinite dimensional space, classifies data points based on the distance between data points and support vector machine, and has good anti-interference ability in processing data noise [10]. The calculation formula is:

$$K(X_i, X_j) = \exp\left(-\frac{\left\|X_i - X_j\right\|^2}{2\sigma^2}\right) \ , \sigma > 0 \tag{3}$$

The σ is the bandwidth parameter, which controls the radial range of the function. Laplace kernel function: It is computationally simpler than the Gaussian function, and performs better than linear datasets in processing nonlinear datasets.

$$K(X_i, X_j) = \exp\left(-\frac{\left\|X_i - X_j\right\|}{\sigma}\right) \ , \sigma > 0 \tag{4}$$

Similar to the Gaussian function, σ is also a bandwidth parameter.

Sigmoid kernel function: The data is mapped into nonlinear feature space by hyperbolic tangent function, which is suitable for dealing with nonlinear separable cases. It is relatively rarely used in practical cases. The calculation formula is:

$$K(X_i, X_j) = \tanh\left(\beta X_i^T X_j + \theta\right) \ , \beta > 0 \, , \theta < 0 \tag{5}$$

Where parameter β controls the slope of the kernel function and parameter θ controls the horizontal offset of the kernel function.

3.3. Introduction of SVM Parameters

SVM parameters and kernel function have an important impact on the final prediction accuracy of the model, and optimization of the parameters can enhance the prediction precision of the model. In general, the parameters to be optimized are penalty parameter C, and kernel function σ [10]. Common optimization methods are Grid Search, Random Search, and Bayesian optimization. In this paper, grid search is used to optimize the parameters.

3.4. Data Noise Reduction

Data noise reduction is a common technique in signal systems to improve the predictive ability and accuracy of models, and real signal information can be extracted from signals containing noise. In the financial domain, stock price data is usually affected by market fluctuations, trading volume changes, information dissemination delays, and other noises, so it is necessary to denoise stock price data. The traditional noise reduction methods include the moving average method, Fourier transform denoising method, Wiener filter method, and Kalman filter method. The moving average method is a simple and rough denoising method, which removes some useful information at the same time while removing noise; Fourier transform denoising is suitable for processing stable signals with strong change cycles and few spikes; Wiener filtering requires knowing the information of noise and useful signals in advance before use; Kalman filtering requires knowing the movement law of the system when used. Financial data time series is a kind of non-stationary, nonlinear, fluctuating data with unknown motion law, noise, and useful information that are not easy to distinguish, so the above methods are not applicable [11]. Wavelet denoising is a method that uses wavelet transform to decompose the signal into frequency domain components of different scales, filter the noise through threshold processing, and then reconstruct the filtered signal back to the time domain. Compared with the traditional filter denoising method, wavelet denoising can effectively retain the important characteristics of the signal and filter out noise. In wavelet denoising, the commonly used threshold processing methods are soft threshold and hard threshold: soft threshold sets the signal coefficient less than the threshold to zero, and linearly decays the signal coefficient greater than the threshold. The formula is as follows:

$$soft(x,T) = \begin{cases} x+T & x \leq -T \\ 0 & |x| < T \\ x-T & x \geq T \end{cases} \tag{6}$$

The hard threshold values then directly set the signal coefficients smaller than the threshold to zero and remain unchanged for those larger than the threshold, as follows:

$$\eta_H(\omega,\lambda) = \{ \begin{matrix} \omega & ,|\omega| > \lambda \\ 0 & ,|\omega| < \lambda \end{matrix} (7)$$

In this paper, wavelet denoising will be used to denoise the data by comparing the denoising effect of soft threshold and hard threshold to select the best method.

4. Empirical Analysis

4.1. Data Selection

This paper selects the stock price index of CSI 300 for research, data processing, and data modeling process are completed with Python. Firstly, the API interface is obtained from the Alpha Vantage website, and the opening price, maximum price, minimum price, closing price, and trading volume of CSI 300 from 2003 to 2024 are obtained. The abnormal data with 0 index in 2003 and 2004 are deleted. The actual period is 4657 data from January 4, 2005, to March 8, 2024. The reason for selecting this data is described below from the perspective of relative strength index (RSI) [11]. The relative strength index shows the market according to the price rise and fall and divides a stock into two categories according to the price rise and price fall within n days, the higher RSI index is the price increase category, and the lower RSI index is the price decline category, and the specific calculation formula is [11]:

$$RSI = 100 - \frac{100}{1 + RS} \tag{8}$$

Where RS is the relative intensity and is calculated as:

$$RS = \frac{Average\ Gain}{Average\ Loss} \tag{9}$$

Taking the closing price (Close) of CSI 300 as an example, nearly 1,000 data are selected to establish a scatter plot of relative strength indicators and stock fluctuation as shown in Figure 1, in which the abscissa indicates the RSI size and the ordinate indicates the relative stock fluctuation:

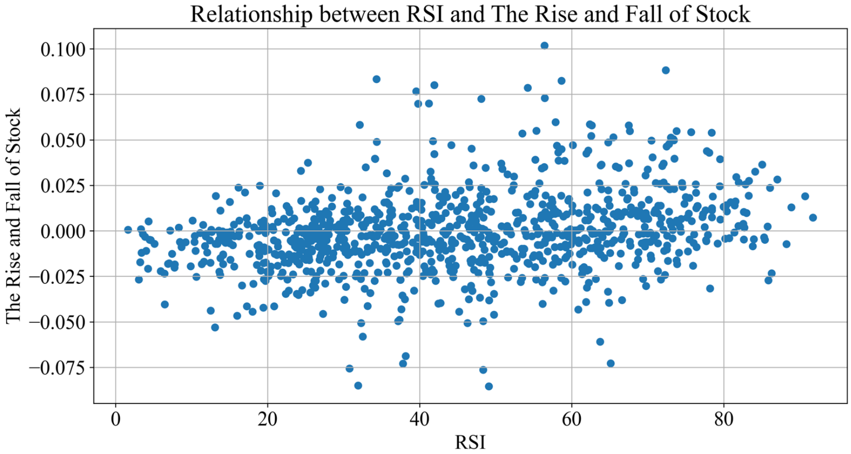

Figure 1: Relationship between RSI and The Rise and Fall of Stock (Photo/Picture credit: Original).

It can be seen from Figure 1 that at a lower level of RSI, the ordinate of the data is concentrated in the negative half-axis of the vertical axis, indicating that the stock has a downward trend; at a higher level of RSI, the ordinate of the data is concentrated in the positive half axis of the vertical axis, indicating that the stock has an upward trend, which is generally consistent with the above description of this index and can be used as an example to predict the stock price.

4.2. Data Processing

4.2.1. Sample Division

This paper divides the data into a training set and test set according to the ratio of 8:2 from far to near according to time, and draws the closing price trend diagram of Shanghai and Shenzhen 300 from January 4, 2005, to March 8, 2024, by using Python's Pandas library and matplotlib library as shown in Figure 2:

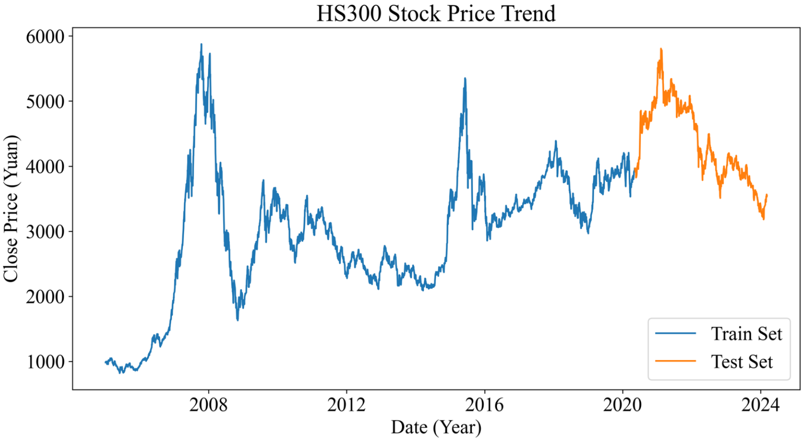

Figure 2: HS300 Stock Price Trend (Photo/Picture credit: Original).

In Figure 2, the blue part is the training set, and the orange part is the test set. It can be seen from the figure that the stock price of CSI 300 has fluctuated unevenly in the past 19 years, and the three-stock price plummeting occurred in 2008, 2015, and 2021, corresponding to the major historical events of the financial crisis in 08, the stock disaster in 15 years and the global epidemic in 21, respectively. After 21 years, the stock price showed a downward trend.

4.2.2. Feature Engineering

In empirical modeling of machine learning, feature engineering usually needs to extract the optimal data to predict to achieve the optimal effect of prediction, and feature engineering is to transform and process the original data to facilitate the machine learning model to better understand and use the process of data. In this paper, the feature engineering of the CSI 300 example includes the following aspects: in terms of feature enhancement, the data cleaning is carried out first, because the sample is large, and the direct deletion method is used for missing values and abnormal values; in terms of feature scaling, the data are normalized. In this paper, Z-score normalization is selected from Min-Max scaling, Z-score normalization method, and normalization method for normalization processing. The formula is as follows:

$$X_{norm} = \frac{X - \mu}{\sigma} \tag{10}$$

Where μ is the mean and σ is the standard deviation of the data.

4.2.3. Threshold Denoising

Financial time series data contains a lot of noise. In this paper, the wavelet denoising method is used to denoise the data of CSI 300 closing price after feature engineering with wavelet transform layers of 5, 6, 7, and 8 in turn. The results show that when the wavelet transform layers are 7, the denoising effect of the data is the best, and overfitting occurs when the layers are too high. Next, soft threshold and hard threshold methods are successively used for comparison: the comparison plot of the original data and denoised data under the soft threshold processing method is shown in Figure 3, and the data fit is too low:

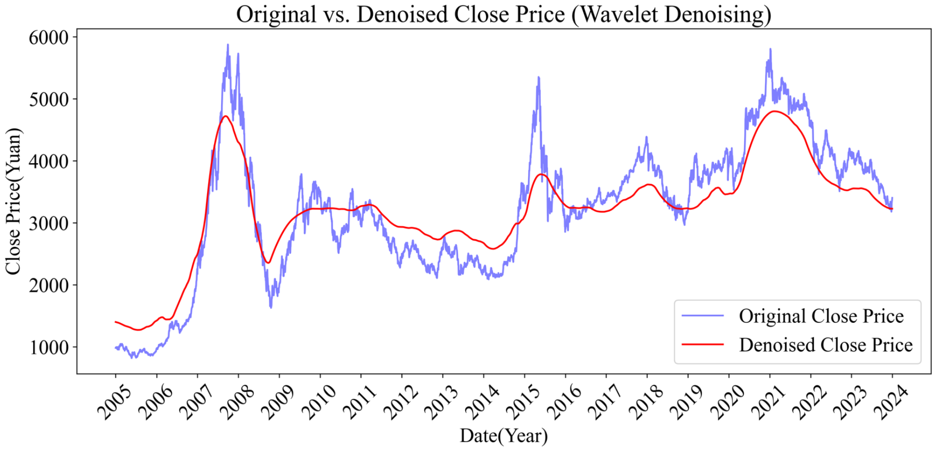

Figure 3: Original vs. Denoised Close Price (Wavelet Denoising)-Soft (Photo/Picture credit: Original).

The comparison between the original data and denoised data under the hard threshold processing method is shown in Figure 4, and it can be found that the data fitting degree is higher than that of the soft threshold processing method:

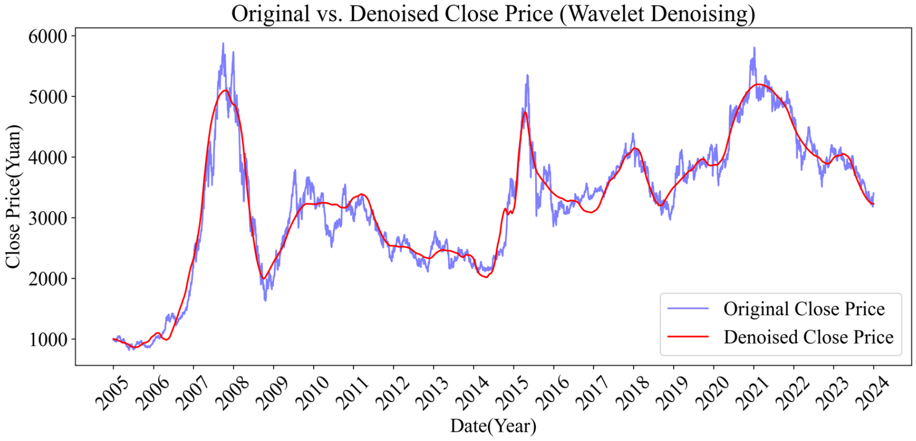

Figure 4: Original vs. Denoised Close Price (Wavelet Denoising)-Hard (Photo/Picture credit: Original).

In summary, the wavelet denoising hard threshold processing method is used to denoise the data.

4.3. Model Prediction

4.3.1. Model Parameter Setting

The penalty parameter C, referred to earlier, represents the tolerance for errors. In this paper, we use the model’s default settings for the penalty parameter C, the kernel function, and the gamma parameter. The gamma parameter determines the distribution of data after mapping to the space; a larger value results in fewer support vectors.

Next, we will use the grid search method mentioned previously to optimize the model parameters:

- Define parameters: Set the range for the value of C.
- Choose scoring metrics: Select the metrics for assessing model performance, such as accuracy, recall, and F1 score.
- Set up cross-validation: Decide on the strategy for cross-validation.

Implement the grid search: Use the grid search functionality in the scikit-learn library in Python.

After aggregating the mean test scores and the variance of scores from the cross-validation, the results show that the optimal parameters for the model, after optimization, are C=10 and gamma=0.001.

4.3.2. Indicators for Model Evaluation

The penalty parameter C, or Error Tolerance, has implications for model bias and variance. A small value of C results in high bias and low variance, suggesting the model might be overly simple and unable to capture the complexities of the data, although it may perform consistently well on data outside the training set (low variance). Conversely, a large value of C leads to a complex model that fits the training data well but may lack generalizability.

Regularization: The parameter C is effectively the inverse of the regularization term. Regularization aims to limit model complexity to prevent overfitting. In SVM, a smaller C value is indicative of stronger regularization, whereas a larger C value is indicative of weaker regularization. Mathematically, the objective function of an SVM includes a regularization term, typically the L2 norm (sum of squares) of the model weights. C acts as a multiplier before this norm, controlling the strength of regularization. With a large C, the regularization is weaker, and the model prioritizes minimizing training error. With a small C, the regularization is stronger, and the model focuses on keeping the weights small to prevent overfitting.

In practice, cross-validation (such as k-fold cross-validation) is commonly used to determine the optimal C value. The process entails splitting the dataset into k subsets, using k-1 of these subsets to train the model, then verifying the remaining subset. To get the average performance of the model, this process is repeated k times, using a different validation subset each time. The results are then averaged.

Choosing the optimal C value is crucial to balance the model’s performance on training data and its ability to generalize to unseen data, ensuring the model neither overfits nor underfits. When evaluating the impact of parameter C on model performance, the following metrics are commonly used:

- Accuracy: The proportion of correctly classified samples in the dataset.
- Precision: The proportion of correct identifications.
- Recall: The proportion of actual positives that were correctly identified.
- F1 Score: The harmonic mean of precision and recall, providing a single metric that combines both.
- AUC-ROC: The area under the ROC curve, which describes the model’s ability to distinguish between classes. The ROC curve is a plot of the true positive rate (TPR) against the false positive rate (FPR), and the AUC measures the total area underneath this curve.

4.3.3. Evaluation of Model Prediction Results

Predictions were made based on the pre-optimisation model and the printed assessment is shown in Table 1:

Table 1: Model Prediction Evaluation Report

	precision	recall	f1-score	support
-1	0.43	0.56	0.48	300
1	0.75	0.64	0.69	628
accuracy			0.61	928
macro avg	0.59	0.60	0.59	928
weighted avg	0.65	0.61	0.62	928

As can be seen in Table 1, the accuracy of the model prediction is 61%.

The model evaluation report resulting from the optimization of the parameters tuned by grid search is shown in Table 2:

Table 2: Model Prediction Evaluation Report (Optimized)

	precision	recall	f1-score	support
-1	0.74	0.55	0.63	481
1	0.62	0.80	0.70	451
accuracy			0.67	932
macro avg	0.68	0.67	0.66	932
weighted avg	0.68	0.67	0.66	932

It can be seen that after adjusting the parameters, the accuracy of the model prediction is increased to 67%, and the optimization effect is more obvious.

4.4. Model Prediction Results and Analysis

The new CSI 300 upward and downward trend prediction versus the actual price is drawn according to the optimized model parameters as in Figure 5:

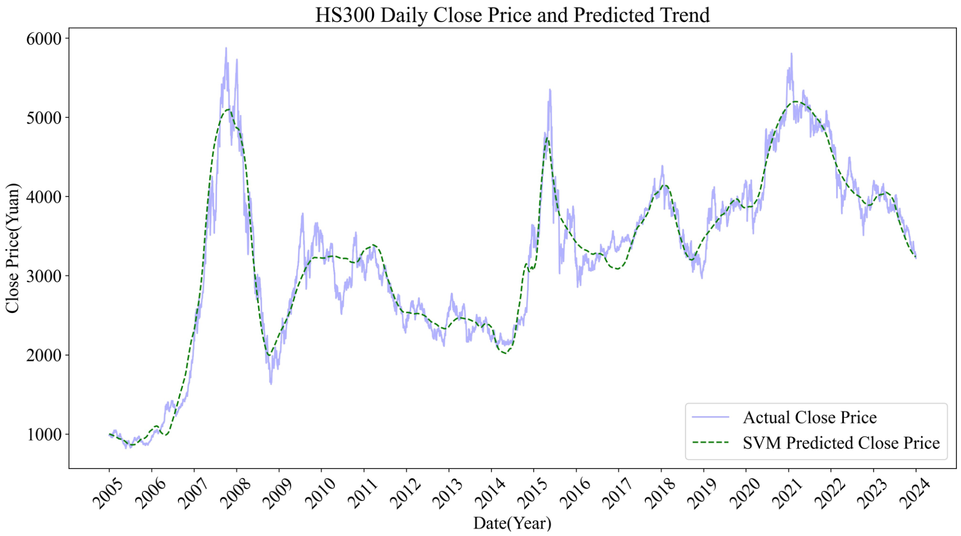

Figure 5: HS300 Daily Close Price and Predicted Trend (Photo/Picture credit: Original).

The blue solid line in Figure 5 shows the actual closing price of CSI 300 based on the time series, and the green dashed line indicates the stock's rise or fall on the following day as predicted using the SVM model.

Looking first at the solid blue line section, the CSI 300 closing price saw a big rise in late 2007 to 2008, followed by a big fall due to the economic crisis after the big rise, and then a stock market rebound in 2009 with a closing price of around 3700. This was followed by a constant fluctuation of closing prices in the range of 2000-3500 from 2009 to mid-2014. Short-term fluctuations can be driven by short-term influences such as market sentiment, news events, technical factors, etc., and these many, many short-term fluctuations shape the long-term trend. In context, the fluctuations of the CSI 300 Index, a composite index reflecting 300 larger and more liquid stocks in the two main stock exchanges of Shanghai and Shenzhen, are usually influenced by macroeconomic factors such as the rate of economic growth, the level of inflation, and monetary policy, in addition to the international financial markets, sectoral factors, technological factors, and market sentiments, etc. 2014 Between mid-year and mid-2015, it rose by 165.8 percent, from over 2,000 to over 5,000, reaching the highest closing value in the decade. After the subsequent decline, the stock index rose 33% between end-August and end-December 2015, a period of 3.9 months, bouncing from over 2,900 to over 3,900 as a result of the official bailout. After the 20s, in 2021, the CSI 300 index had a major bull market, surpassing the top of the 15-year bull market and catching up with the '07 peak at over 5,800, potentially a long-lasting "housing" and capital shifting into the stock market for profit, or it could be the contrast between the Chinese and foreign situation coming out of the new crown epidemic.

And then see the green dotted line part, the overall model predicts the general trend and the actual match, but in some extreme data such as the peak when the prediction of a large error, such as the peak in 07 the actual value is much higher than the predicted value. Secondly, although the general trend is accurate, zoomed into the details of each period there are large and small errors, but the error range is generally not more than plus or minus 100.

After using the optimized model to predict the data to get the results, the confusion matrix was calculated using the predicted results and the real labels, and the heat map of the confusion matrix was plotted as in Figure 6:

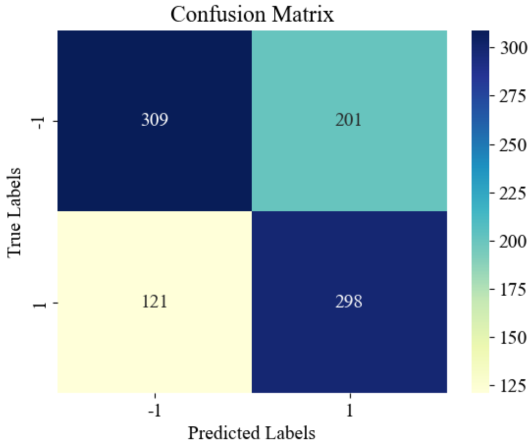

Figure 6: Confusion Matrix Heatmap (Photo/Picture credit: Original).

As can be seen from Figure 6, in 20% of the test sets of the CSI 300 time series, the model predicts the number of negative samples in negative categories as 263, the number of negative samples in positive categories as 218, the number of positive samples into negative categories as 92, and the number of positive samples into positive categories as 359. the model prediction accuracy, i.e., the ratio of the sum of the main diagonal elements to the total elements, is 67%, which is in line with the assessment report.

5. Conclusion

The stock market is characterized by high risk and high return, and how to accurately predict the upward and downward trend of stocks has been a continuous concern for investors and research scholars. For a long time, Shanghai and Shenzhen 300 have maintained a high degree of heat in China. This study employs the widely-used support vector machine model to examine the upward and downward trends of the stocks in the CSI 300, using 4,657 trading days as the time series. The stock-related market indexes, such as the opening price, closing price, highest price, lowest price, and technical indexes of the RSI, are chosen, and the relevant feature engineering principles are applied to create forecasts. The accuracy of the results obtained from the forecasting process is 61%.

A grid search was used to find the parameter combination with the highest average test score, C=10, and gamma=0.001, to adjust the model parameters, and the new evaluation report obtained after the optimization showed that the accuracy rate was increased to 67%.

The shortcoming of this study is that even though a high accuracy rate was obtained, it is still unknown whether support vector machines are more suitable for stock prediction compared to other machine learning methods. In future research, a side-by-side comparison can be made between Support Vector Machines and other machine learning models such as Bayesian models, Gradient Boosting Trees, XGBoost, LightGBM, etc., to highlight the strengths or weaknesses of SVMs, and then analyze and optimise the strengths and propose improvements for the weaknesses.

Authors Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Jiahua, F. (2023). Stock trend prediction method and application based on KNN improved SVM. Henan University.

[2] Qiji, C. Xuejun, H. (2023). A study on stock return forecasting based on SVM and ARIMA-EGARCH. Journal of Economic Research, (21): 84-86.

[3] Hongquan, L. Liang, Z. (2023). Systemic financial risk monitoring and early warning based on machine learning techniques. Operations Research and Management, 32(11): 212-219.

[4] Sheng, L. Ke, Q. Kaicong, W., etc. (2019). A review of machine learning in stock price prediction. Economist, (03): 71-73+78.

[5] Yuchuan, Z. Zuoquan, Z. (2007). Application of SVMs in stock price prediction. Journal of Beijing Jiaotong University, (06): 73-76.

[6] Lifang, P. Zhiqing, M. Hua, J., etc. (2006). Application of time series-based support vector machine in stock forecasting. Computing Technology and Automation, (03): 88-91.

[7] Zhiyuan, H. (2017). Stock prediction system based on data mining method. Nanjing University of Science and Technology.

[8] Yibing, C. Lingling, Z. Yong S. (2011). Financial time series forecasting based on improved support vector regression machine. China Management Modernization Research Association. Abstracts of the Sixth (2011) Annual Conference on Management in China, 1.

[9] Cheng, L. (2018). Stock index futures price prediction based on wavelet kernel support vector machine regression. Shanghai Normal University.

[10] Yuping, K. (2023). A time series and machine algorithm based stock forecasting analysis of Industrial and Commercial Bank of China. China Management Informatization, 26(06): 146-148.

[11] Yi, Y. (2021). Research on Chinese spirits stock prediction based on machine learning algorithm modeling. Shandong University.