PRAGUE UNIVERSITY OF ECONOMICS AND BUSINESS

Acta Informatica Pragensia
2024, Volume 13, Issue 1, pp. 24–37
https://doi.org/10.18267/j.aip.226

Article Open Access

Optimized Ensemble Support Vector Regression Models for Predicting Stock Prices with Multiple Kernels

Subba Reddy Thumu, Geethanjali Nellore

Department of Computer Science and Technology, Sri Krishnadevaraya University, Anantapur, India

Corresponding author: Subba Reddy Thumu (subbareddy.thumu@gmail.com)

Abstract

Stock forecasting is a complicated and daily challenge for investors because of the non-linearity of the market and the high volatility of financial assets such as stocks, bonds and other commodities. There is a need for a powerful and adaptive stock prediction model that handles complexities and provides accurate predictions. The support vector regression (SVR) model is one of the most prominent machine learning models for forecasting time series data. An ensemble hyperbolic tangent kernel SVR (HTK-SVR-BO) is proposed in this paper, combining Tanh and inverse Tanh kernels with Bayesian optimization. Combining the strengths of multiple kernels using the ensemble technique and then using optimization to identify the optimal values for each SVR model to enhance the ensemble model performance is possible. Our proposed model is compared with an ensemble SVR model (LPR-SVR-BO), which uses well-known SVR kernel types, including linear, polynomial and radial basis function (RBF). We apply the proposed models to Microsoft Corporation (MSFT) stock prices. The mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE), R2 score (model accuracy) and mean absolute percentage error (MAPE) are the regression metrics used to compare the effectiveness of each ensemble model. In our comparison, HTK-SVR-BO performs better in terms of regression metrics compared to LPR-SVR-BO and achieves results of 0.27424, 0.13392, 0.36595, 0.99997 and 5.2331 respectively. According to the analysis, the proposed model is more predictive and may generalize to previously unknown data more effectively, so it can be accurate when forecasting future stock prices.

Keywords

Microsoft corporation (MSFT); Stock forecast; SVR; Hyperbolic tangent kernels (HTK); Linear polynomial RBF kernels (LPR); Ensemble model; Bayesian optimization (BO); Regression metrics.

Citation: Thumu, S. R., & Nellore, G. (2024). Optimized Ensemble Support Vector Regression Models for Predicting Stock Prices with Multiple Kernels. *Acta Informatica Pragensia*, 13(1), 24–37. https://doi.org/10.18267/j.aip.226

Academic Editor: Zdenek Smutny, Prague University of Economics and Business, Czech Republic

Copyright: © 2024 by the author(s). Licensee Prague University of Economics and Business, Czech Republic.
This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution License (CC BY 4.0).

1 Introduction

Stock price predictions are challenging in the dynamic landscape of financial markets, where it has never been easier to spot trends. A reliable forecasting method is necessary due to the complex interplay between market forces, sentiments and external forces, which requires advanced methods (Siddique et al., 2019). Machine learning algorithms are used to construct predictive stock models based on historical data, people's sentiments and news events (Singh et al., 2019). The Microsoft Corporation (MSFT) share dataset provides the foundation for our predictive endeavours (Sharma et al., 2021). Stock prices, volume trading, opening and closing prices, the highest and lowest stock prices and more are included in this dataset. Various factors influence these data points, including corporate performance, market sentiment, economic indicators and geopolitical events.

Machine learning tool support vector regression (SVR) models complex relationships within financial data (Lin et al., 2013; Sharma et al., 2021; Thumu & Nellore, 2021). SVR is a machine learning technique for forecasting continuous numerical outcomes based on SVM (support vector machine). SVR (Rubio & Alba, 2022, Kumar et al., 2021; Hu et al., 2009; Huang et al., 2022; Virigineni et al., 2022) essentially seeks to identify an optimal hyperplane to maximise margins while accommodating a preset error level, effectively capturing patterns and trends (Dash et al., 2021). Kernel tricks facilitate nonlinear relationship modelling by projecting data into a higher-dimensional space. As a result, SVR can capture intricate patterns not visible in the original feature space. Each kernel function caters to specific data characteristics, such as linear, polynomial and radial basis functions. Combining these diverse kernels creates an ensemble model that is more accurate and robust than a single kernel (Divina et al., 2018). Various real-world ensembles (bagging, boosting, stacking and voting) impact the algorithm outcome (Gao et al., 2019; Rubio & Alba, 2022; Gohar et al., 2023; Kenfack et al., 2021). In our ensemble SVR framework, a Bayesian optimization approach is applied to fine-tune the models for improved performance. Hyperparameters significantly influence machine learning models and finding their optimal values can improve predictive accuracy (Dash et al., 2021; Majhi & Anish, 2015; Turner et al., 2021; Yang & Shami, 2020). With Bayesian optimization (Alam et al., 2021; Snoek et al., 2012; Wu et al., 2019), the hyperparameter space is efficiently explored and the most optimal values are selected, leading to better model performance.

The purpose of this work is to improve the performance and efficiency of the SVR algorithm by utilizing user-defined kernels with Bayesian optimization (BO). Here, the kernels are a pair of mathematical functions, Tanh() and aTanh(). A framework incorporating both hyperbolic and inverse hyperbolic tangents is presented. Inspired by sigmoidal activation functions of neural networks, these kernel functions reveal nonlinear relationships within data using a unique approach. Using an ensemble technique, the evidence of both these kernels may be combined into a single outcome. Finally, the BO is applied to the ensemble SVR in order to fine-tune the hyperparameters for reduced error rates and higher accuracy.

The rest of the paper is structured as follows. Section 2 gives a summary of the literature; Section 3 describes the methods and related concepts; Section 4 proposes a new SVR approach; Section 5 provides simulation results and discussion; and conclusions and recommendations for further study are presented in Section 6.

2 Literature Review

As part of this section, various machine learning models and optimization strategies will be presented that will be used to construct the proposed model. Sharma et al. (2021) utilized a variety of machine learning techniques to estimate and forecast MSFT stock market values, such as SVR with linear and radial basis function (RBF) kernels, artificial neural networks (ANNs) and autoregressive integrated moving averages

(ARIMA). According to the proposed work, the SVR-RBF kernel has shown a lower root mean squared error (RMSE) for estimating daily stock prices than other models.

Dash et al. (2021) proposed a new method of forecasting stocks using SVR that uses a fine-tuned version. In order to select and optimize the kernel function, grid search is applied, reducing time and memory requirements and increasing accuracy. The suggested approach will be used to examine various stock market performance metrics. It shows superior accuracy and quicker computation compared to similar approaches in terms of RMSE and mean absolute percentage error (MAPE).

Siddique et al. (2018) suggested a hybrid model that combines SVR and particle swarm optimization (PSO) to improve predictions of the price of Tata Steel stock. A lagged time series dataset was analysed using 35 attributes using SVR and PSO to optimize SVR hyperparameters. It achieved MAPE of approximately 0.7% on the testing dataset, outperforming existing models.

In order to estimate short-term electric energy consumption, Divina et al. (2018) presented an ensemble learning technique that stacks predictions from three base learning methods to provide final predictions. The authors used a dataset of Spain's energy use spanning more than nine years to evaluate their suggested methodology. The experimental findings showed that the ensemble strategy works better on the dataset than current state-of-the-art techniques, resulting in lower error rates and extremely accurate predictions.

To forecast the shear strength of fibre-reinforced polymer (FRP) reinforced concrete elements, Shah Alam et al. (2021) developed a hybrid model combining the Bayesian optimization algorithm (BOA) and SVR. The hybrid BOA-SVR model was more accurate and resilient when compared to design codes, guidelines and other artificial intelligence models using statistical performance measures.

Rubio and Alba (2022) suggested a hybrid model that combined SVR and ARIMA to predict the daily and cumulative returns of Colombian businesses listed on the New York Stock Exchange (NYSE). The ARIMA model is widely used in time series prediction but cannot capture non-linear regression behaviour due to increased volatility. In contrast, the SVR model uses machine learning to solve non-linear regression estimation problems. Combining these two models results in a hybrid ARIMA-SVR model that provides investors with more accurate forecasts.

This paper builds upon the existing literature on stock price prediction by introducing an ensemble approach that takes advantage of the strengths of numerous SVR kernel functions, including the hyperbolic tangent kernel, while incorporating Bayesian optimization for hyperparameter tuning. Microsoft stock prices can be predicted more accurately and reliably using this comprehensive approach, which will help investors and financial analysts make informed decisions.

3 Data and Methodology

3.1 Dataset

The proposed work involved gathering historical daily stock prices for MSFT (Sharma et al., 2021), comprising date, opening price, high price, low price, close price, adjusted close price and volume. The dataset (see Additional Information and Declarations section) used for the work was gathered from Yahoo finance between 1 April 1986 and 30 December 2022. A computer with 16 GB of RAM and an Intel core i7 processor running on Microsoft windows 11 x64 was used. Implementations of each algorithm were carried out using the MATLAB programming language. The sample dataset is presented in Table 1.

Table 1. *MSFT sample dataset.*

Date	Open	High	Low	Close	AdjClose	Volume
01-04-1986	0.095486	0.095486	0.094618	0.094618	0.058903	11088000
02-04-1986	0.094618	0.097222	0.094618	0.095486	0.059444	27014400
03-04-1986	0.096354	0.098958	0.096354	0.096354	0.059984	23040000
04-04-1986	0.096354	0.097222	0.096354	0.096354	0.059984	26582400
07-04-1986	0.096354	0.097222	0.092882	0.094618	0.058903	16560000
08-04-1986	0.094618	0.097222	0.094618	0.095486	0.059444	10252800
09-04-1986	0.095486	0.09809	0.095486	0.097222	0.060524	12153600
10-04-1986	0.097222	0.098958	0.095486	0.09809	0.061065	13881600

The adjusted close price attribute is considered the target attribute for the machine learning model, but most researchers have used it as their target variable. Adjusted close price is more advantageous than close price. In most corporate actions, the adjusted close price is significant, consistent over time, allows meaningful comparisons across periods and provides better results.

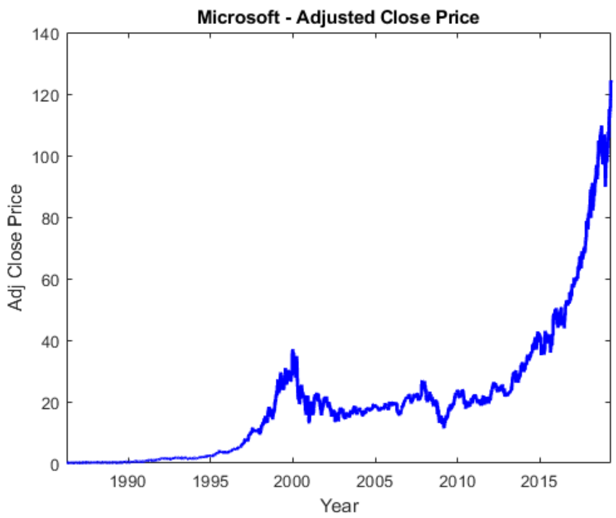


Figure 1. *MSFT dataset.*

3.2 Pre-processing

Machine learning requires pre-processing to enhance data quality, improve model performance and enable algorithms to make accurate predictions by addressing noise, missing data and inconsistent formats. The collected dataset contains 9270 records, with only a few (five) null values. It is necessary to remove those records from the dataset to clean it. As a result, the dataset now consists of 9265 records. A correlation matrix is also found; after finding the correlation, all attributes except volume have a high correlation, so it is eliminated. Our dataset is now ready to be used with machine learning models.

3.3 Support vector regression

SVR (Rubio & Alba, 2022; Sharma et al., 2021) is a new method for predicting time series for nonlinear regression issues. The structural minimization concept is used in SVR, which employs a linear function hypothesis space in a high-dimensional feature space. Kernel functions are used to map SVR inputs into a high-dimensional feature space. The SVR model decision function may be written as

$$y = w \, . \, \phi(x) + b \tag{1}$$

where w and b are constant vectors, x is the input and $\phi(x)$ is a nonlinear function.

The main goal of the SVR method is to solve the following optimization issue by determining the optimal values for w and b in Equation (1).

$$\min_{w,b,\xi_i,\xi_i^*} \left\{\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*)\right\} \quad (2)$$

where C is a penalty coefficient parameter that controls the severity of penalised loss when a training mistake occurs, and the slack variables ξ_i and ξ_i^* are included to account for training errors above and below an interval of ε tube. After minimising Equation (2), the dual optimization issue transforms into constructing a dual pattern, further expressing the nonlinear SVR.

$$\min_{\alpha_i,\alpha_i^*} \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i).(\alpha_j^* - \alpha_j).K(x_i,x_j) + \varepsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) - \sum_{i=1}^{l} y_i.(\alpha_i^* - \alpha_i) \quad (3)$$

where α_i and α_i^* are Lagrange multipliers with the kernel function K, where $K(x_i,x_j) = \Phi(x_i) \cdot \Phi(x_j)$. Among widely used SVR kernel functions are the linear, polynomial and radial basis functions. The decision function for SVR may thus be written as

$$f(x_i) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)K(x_i,x_j) + b \quad (4)$$

It is considering that the operation involved between training tuples for kernel functions is the dot product (Dash et al., 2021). For each occurrence of the dot product of data tuples, a kernel function can be used instead of performing the dot product on transformed data tuples. As a result, all calculations are carried out in the original input spaces. Here are some of the admissible kernel functions:

Linear kernel of degree n is: $$\boldsymbol{K(X,Y) = X^T Y} \quad (5)$$

Polynomial kernel of degree d is: $$\boldsymbol{K(X,Y) = (\gamma \cdot X^T Y + 1)^d, \gamma > 0} \quad (6)$$

Radial basis function kernel is: $$\boldsymbol{K(X,Y) = \exp\left(-\frac{\|X-Y\|^2}{2\sigma^2}\right)} \quad (7)$$

Here γ, σ, d are kernel parameters.

3.4 Bayesian optimization

This section discusses the development of Bayesian optimization (BO) for setting SVR kernel parameters. Generally, BO consists of two major components: as a first step to constructing a Bayesian statistical model for objective functions, learning from observations and predicting unseen objective function values from observations. The Gaussian process (GP) (Snoek et al., 2012) prior was chosen for this purpose due to its flexibility and tractability. The second step is to select an acquisition function (Wu et al., 2019). Optimal observations are made by evaluating the objective to determine where to observe next. One of the activation functions, called expected improvement (EI) (Wu et al., 2019) is chosen here. The EI function analyses the region surrounding the current optimal value to calculate the predicted level of improvement that a point can achieve.

It is a display of the EI acquisition function performance for optimization. The dotted red line shows the objective function value, the blue line shows the GP posterior mean, and the red vertical line shows the position of the current optimum value (Wu et al., 2019). T signifies the iteration steps. The global optimal value is discovered using the EI function after five rounds. The EI function will be used to simplify the machine learning algorithm hyperparameters.

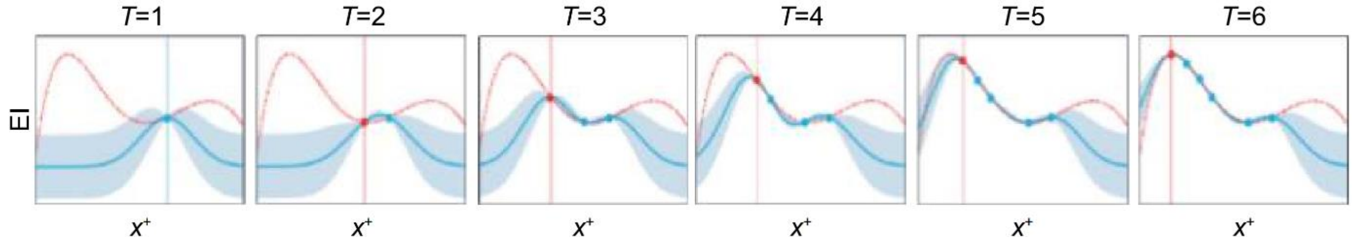


Figure 2. *Optimization process of acquisition function. Source: (Wu et al., 2019).*

3.5 Ensemble technique

In order to improve prediction performance and accuracy, ensemble learning combines multiple learning models, drawing strength from each model and combining them into a composite model that is more robust and accurate while also avoiding over-fitting of the data. Several ensemble techniques are available, including bagging, boosting and stacking. Multiple models are built during bagging and their results are combined via voting or averaging. Our machine learning model uses continuous data so that the bagging technique can aggregate the data. Ensemble models are created by gathering heterogeneous or homogeneous regression models with varying parameters, calculating the results for each regression model and combining them to yield better results. An overview of the ensemble technique can be seen in the diagram.

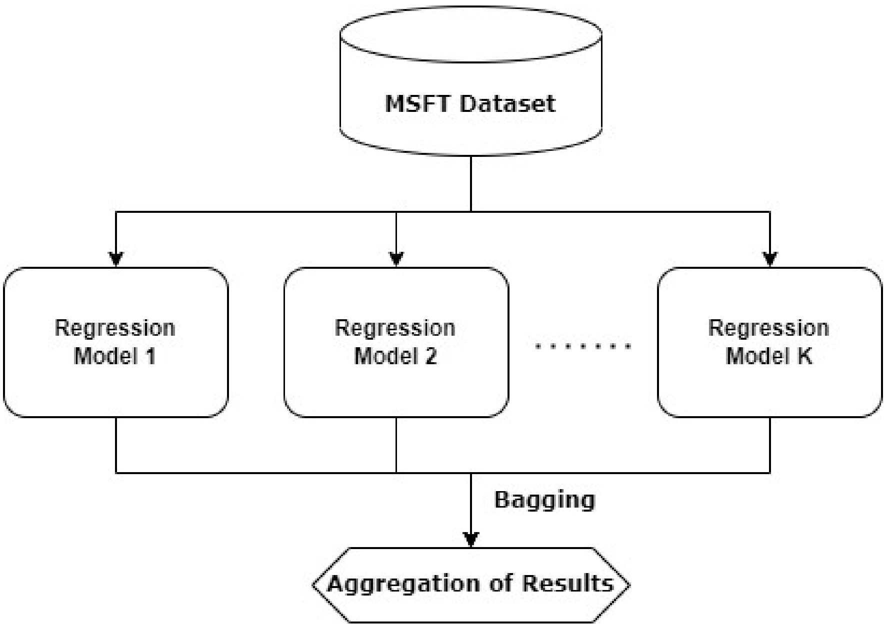


Figure 3. *Bagging ensemble technique.*

3.6 Prediction evaluation measures

In order to determine which model is the best approach for the original time series, the errors are calculated using the following measures: mean absolute percentage error (MAPE) (Dash et al., 2021), mean absolute error (MAE) (Rubio & Alba, 2022), mean squared error (MSE) (Rubio & Alba, 2022) and root mean squared error (RMSE) (Sharma et al., 2021) and by evaluating the R^2 score (Rubio & Alba, 2022), the model accuracy can be determined.

$$MAE = \frac{1}{M}\sum|z_i - \hat{z}_i| \quad (8)$$

$$MAPE = \frac{1}{M}\sum\frac{|z_i - \hat{z}_i|}{z_i} \quad (9)$$

$$MSE = \frac{1}{M}\sum(z_i - \hat{z}_i)^2 \quad (10)$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{M}\sum(z_i - \widehat{z_i})^2} \quad (11)$$

$$R2 - Score = 1 - \frac{SSR}{SSM} = 1 - \frac{\sum(z_i - \widehat{z_i})^2}{\sum(z_i - \overline{z_i})^2} \quad (12)$$

SSR = regression line – squared sum error

SSM = mean line – squared sum error

M is the total number of records, z_i is the actual value, $\widehat{z_i}$ is the predicted value, and $\overline{z_i}$ is the mean value of z_i.

4 Proposed Method

Creating custom kernels for SVM enhances model performance, captures complicated data patterns, incorporates domain knowledge and allows users to develop customised solutions for specific problem domains. Tanh() and aTanh() functions are machine learning tools for capturing complicated non-linear patterns in data, improving performance and making models easier to understand. They employ feature transformers, activation functions and SVM kernels to identify patterns.

$$\text{Hyperbolic tangent kernel} - TK(x_1, x_2) = \text{Tanh}\ (p_1 + (p_2 * (x_1 . x_2'))) \quad (13)$$

$$\text{Inverse hyperbolic tangent kernel} - ITK(x_1, x_2) = \text{Tanh}^{-1}(p_1 + (p_2 * (x_1 . x_2'))) \quad (14)$$

where x_1 and x_2 are values of independent variables, p_1 and p_2 are the kernel parameters.

The suggested model uses SVR together with BO to forecast MSFT stock. Section 3 describes the mathematical formulation of SVR and optimization methods. The steps involved in predicting the stock price of MSFT shares are given below, and Figure 4 presents the stages that make up the suggested model. The following steps, which can be implemented in MATLAB, are used in the work:

1. Select the MSFT dataset, pre-processing and feature selection.
2. Create training and validation datasets with different ratios 90:10, 80:20, 70:30, 60:40 and 50:50.
3. BO extracts the optimum hyperparameters for the linear, polynomial, RBF, Tanh and aTanh kernels.
4. Create ensemble models for linear polynomial RBF kernels (LPR-SVR-BO) and hyperbolic tangent kernels (HTK-SVR-BO).
5. With various datasets, train and validate the recommended ensemble SVR models.
6. Performance analysis of proposed models LPR-SVR-BO and HTK-SVR-BO.

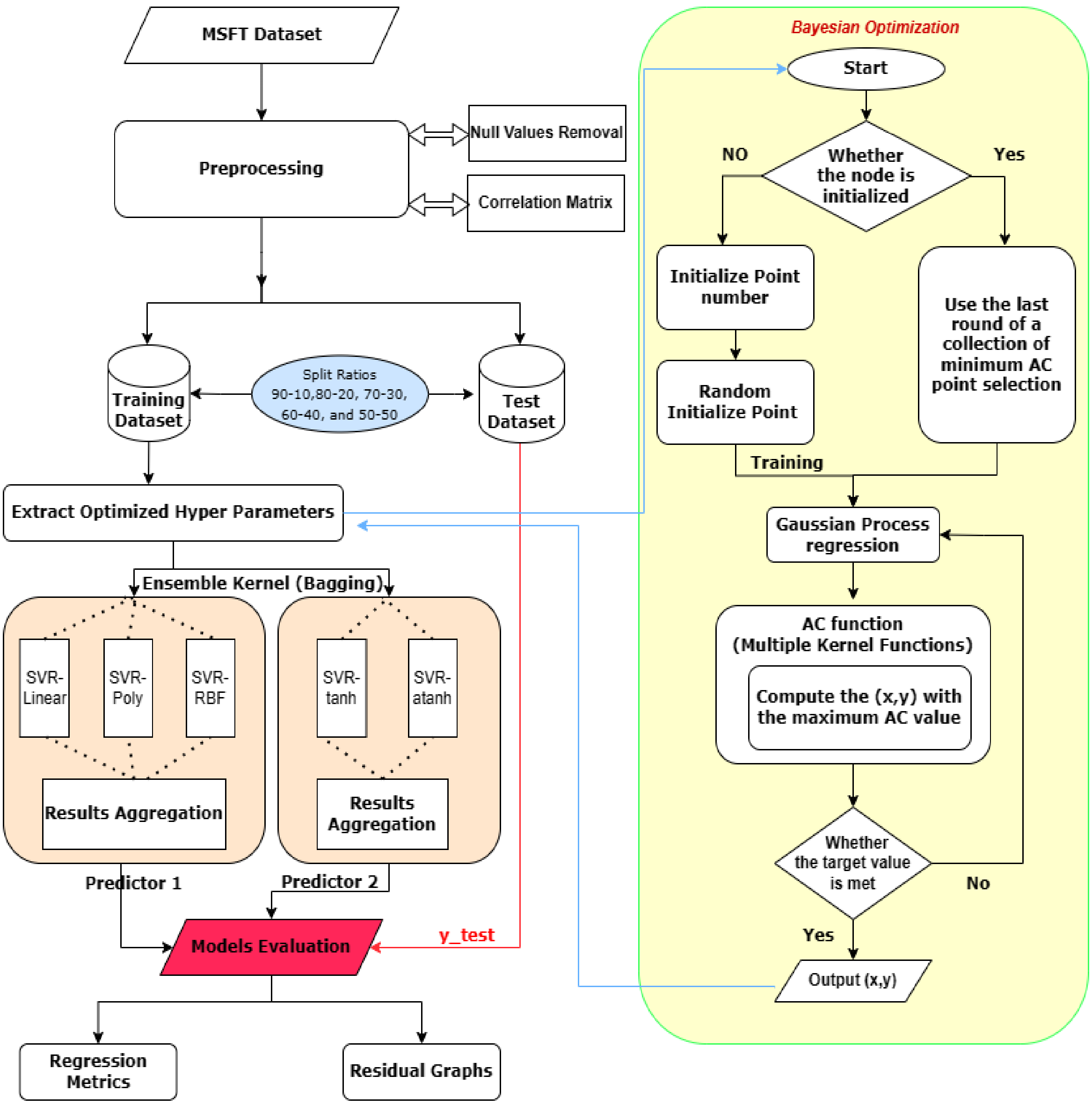


Figure 4. *Optimized ensemble SVR model for Microsoft stock prediction.*

5 Results and Analysis

The MSFT dataset is selected for training and testing datasets with a split ratio of 70% and 30%. These datasets were used for training and testing by Sharma et al. (2021), and they apply a variety of machine learning algorithms such as ARIMA, ANN, SVR (Linear) and SVR (RBF). As a result, the following regression metric results are obtained for RMSE, see Table 2.

Table 2. *Statistical comparison of ARIMA, Neural network, SVR (Linear) and SVR (RBF).*

	ARIMA	**Neural network prediction**	**SVR (Linear)**	**SVR (RBF)**
RMSE	1.67	1.66436	1.65226	1.65
Adjclose (Predicted)	103	106	104	99

This work uses data from the MSFT time series for training and testing. As regression metrics, RMSE is followed by MAE, MSE, R^2 score and MAPE. SVR models are optimized using Bayesian methods with linear, polynomial and RBF kernels. Based on these results, regression metrics can be obtained (Table 3), curves predicted and residual graphs plotted (Figure 6). Tuning the hyperparameters of the kernels using Bayesian optimization techniques improves the results.

Table 3. *Regression metrics for authors' dataset – SVR with Bayesian optimization.*

Kernels	MAE	MSE	RMSE	R^2 score	MAPE
Linear–BO	0.2908	0.15565	0.39452	0.99802	0.25995
Polynomial-BO	0.51001	0.48363	0.69544	0.99385	0.46346
RBF-BO	0.40709	0.30526	0.5525	0.99612	0.36745

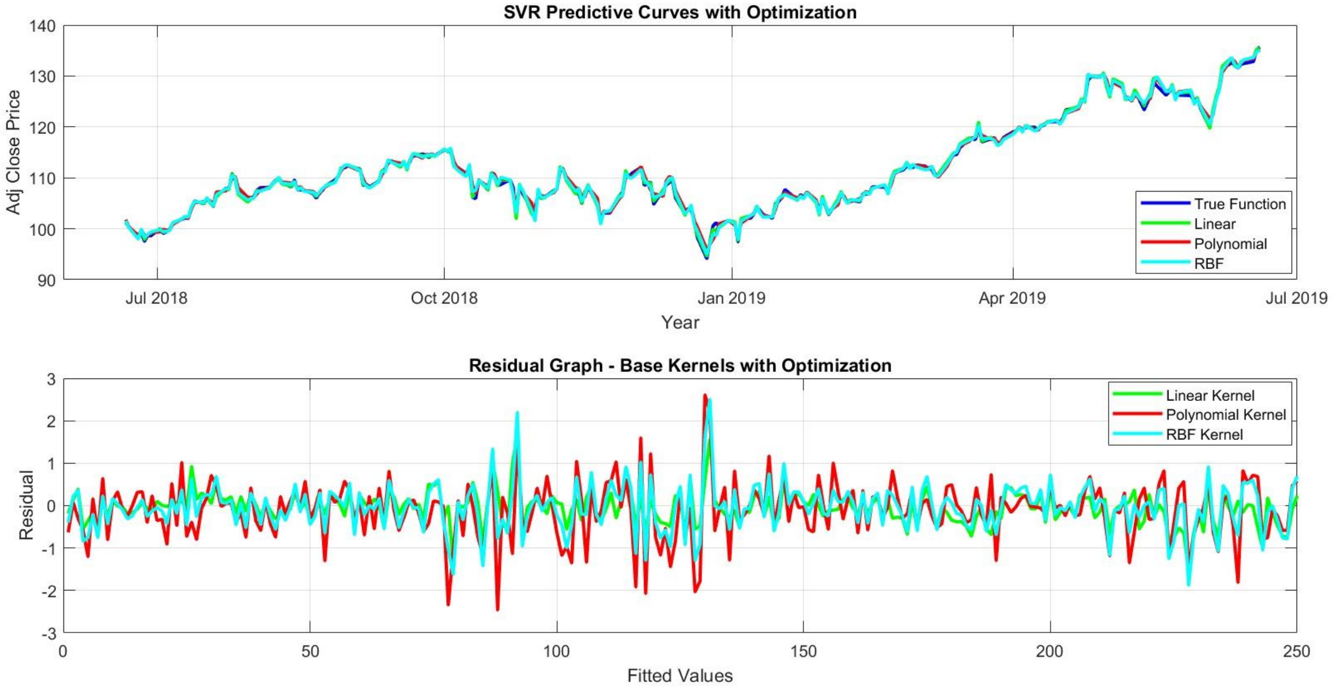


Figure 6. *Predictive and residual graphs for authors' dataset – SVR with Bayesian optimization.*

According to the optimization process, at iteration 0 (Figure 7), both "min observed object" and "estimated min object" points are at the same point in the model. Model parameters are updated as the number of iterations increases to minimize loss. As a result of optimizing, the "min observed object" tracks the lowest observed loss value encountered. "Estimated min objects" predict the location of the lowest loss based on current parameter settings. As the optimization process converges, the "min observed object" and the "estimated min object" should coincide, indicating that the optimized parameters have minimized the loss function.

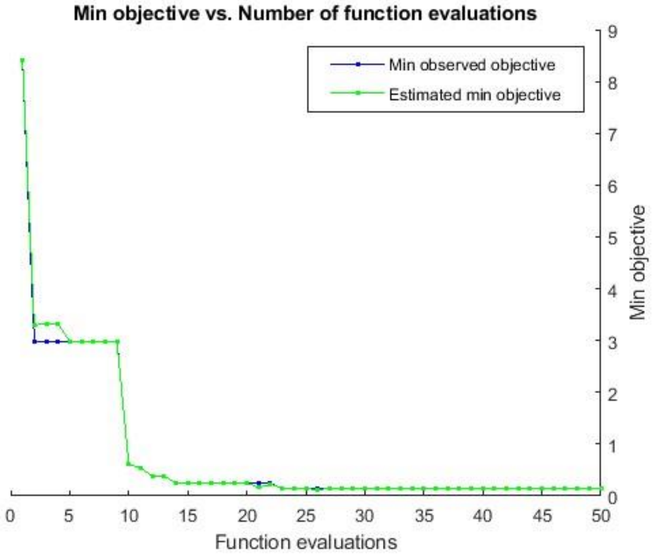
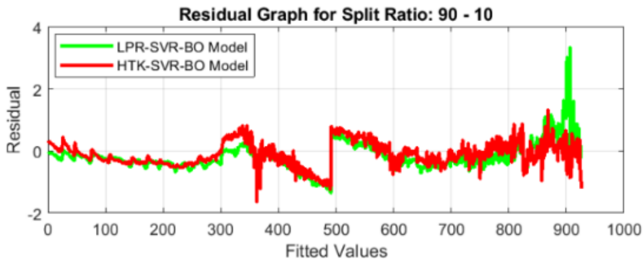
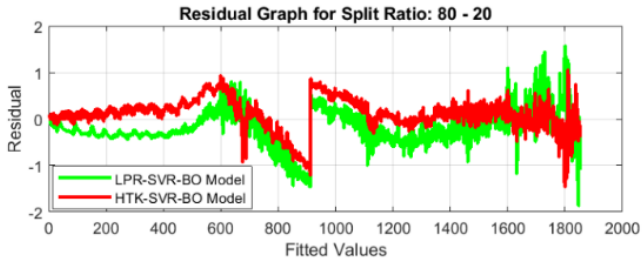


Figure 7. *Bayesian optimization process for tuning hyperparameters of SVR kernels.*

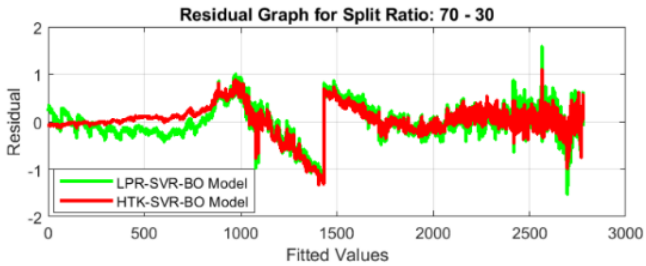
The hyperbolic tangent kernels Tanh() and Tanh-1() are proposed as kernels for support vector regression. An ensemble technique can be used to combine two kernels to obtain both features and outcomes in our work. This model is referred to as HTK-SVR-BO. Compared with the literature results, our results show that SVR linear, polynomial and RBF kernels get better results with optimization. Ensemble methods combine all kernel features into one model. This model is referred to as LPR-SVR-BO.



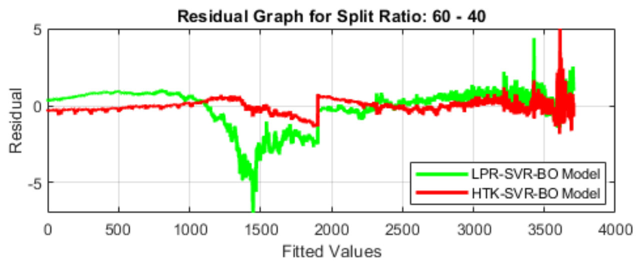
8(a)



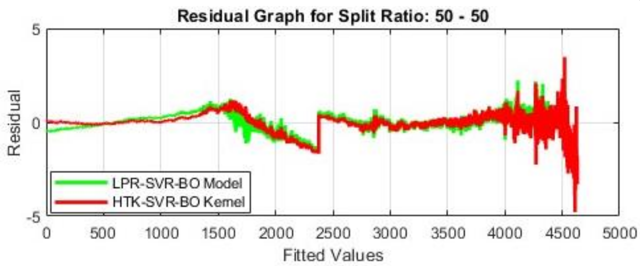
8(b)



8(c)



8(d)

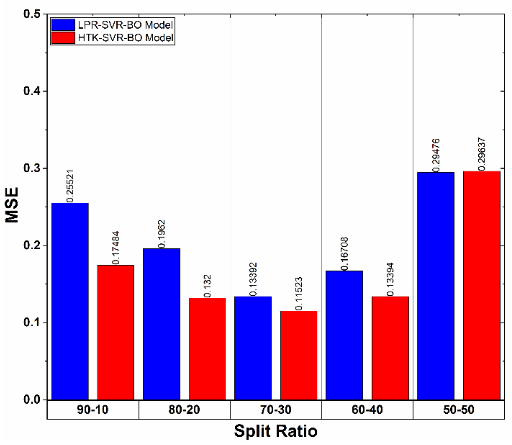


8(e)

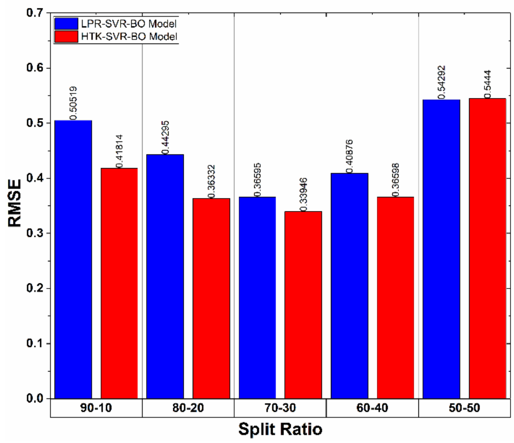
Figure 8(a), 8(b), 8(c), 8(d) and 8(e). *Ensemble model residual graphs with various split ratios of 90:10, 80:20, 70:30, 60:40 and 50:50.*

Ensemble model outcomes are presented in residual graphs for various split ratios of 90:10, 80:20, 70:30, 60:40 and 50:50. In order to separate the dataset into train and test datasets, the holdout partition method is used. After that, random samples are selected from the entire dataset for the train and test datasets. In a 90:10 split ratio, 926 records are in the test dataset and 8339 are in the train dataset. In a similar way, for a split ratio of 80:20, there are 7412 records for trains and 1853 records for tests. There are 6486 train records and 2779 test records for a 70:30 split ratio. There are 5559 train records and 3706 test records for a 60:40 split ratio. There are 4633 train records and 4632 test records for a 50:50 split ratio. A residual graph presents the differences between experimental values (target values) and predicted values. As part of the regression analysis, residual plots are used to diagnose potential problems with a model fit to the data.

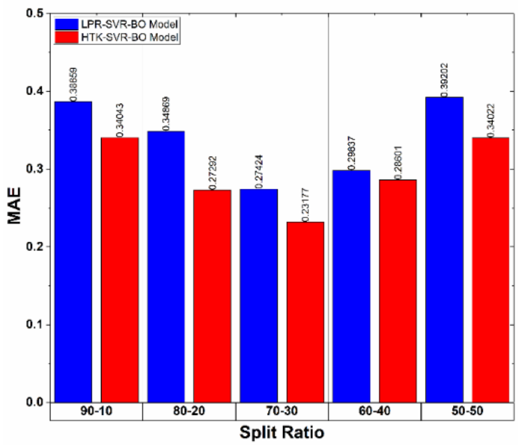
According to the residual graphs in Figures 8(a) – 8(e), the HTK-SVR-BO model is efficient based on the residual graph of both ensemble models. According to Figure 8(a), the LPR-SVR-BO model has more residual errors at the end of the validation dataset. As shown in Figure 8(b), both models have residual errors during validation, but the HTK-SVR-BO model has fewer than the LPR-SVR-BO model. Figure 8(c) shows that the LPR-SVR-BO model has more residual errors at the beginning of the validation. According to Figure 8(d), the LPR-SVR-BO model provides huge residuals when the validation set reaches 1500. Figure 8(e) shows that LPR-SVR-BO starts with more residuals, but at the end of the validation dataset, HTK-SVR-BO shows more residuals. Overall, the HTK-SVR-BO model appears to be efficient.



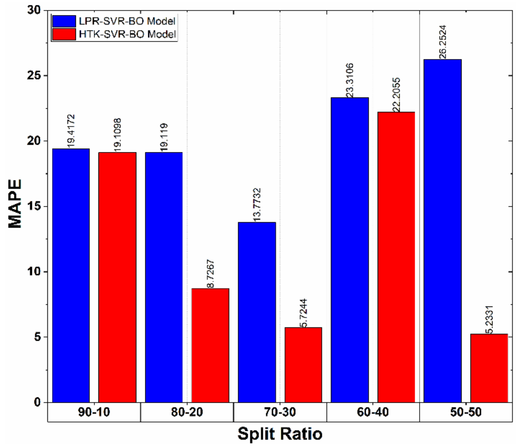
9(a)



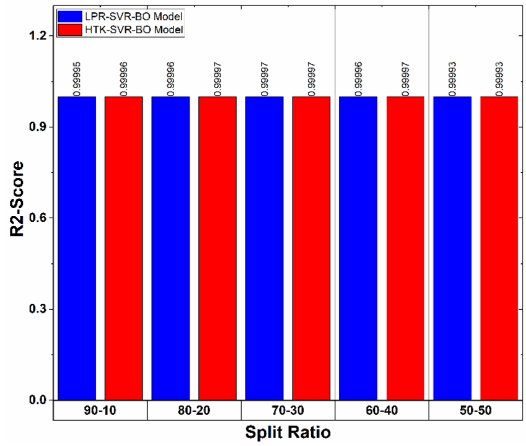
9(b)



9(c)



9(d)



9(e)

Figure 9(a), 9(b), 9(c), 9(d) and 9(e). *Comparison of regression metrics MSE, RMSE, MAE, MAPE and R^2 score for both ensemble models.*

5.1 Comparison

Based on regression metrics MAE, MSE, RMSE, MAPE and R^2 score for the whole dataset with various split ratios, a comparison is made between HTK-SVR-BO, LPR-SVR-BO, SVR (linear) (Sharma et al., 2021) and SVR (RBF) (Sharma et al., 2021). Although SVR (linear) and SVR (RBF) use SVR with a 10-fold technique, they lack an optimization technique. In order to predict the adjusted close price, the proposed methods are compared with SVR (linear) and SVR (RBF) (Table 4). Afterwards, the two ensemble methods are compared. Using HTK-SVR-BO, it is our goal to achieve the lowest values possible for MAE, MSE, RMSE and MAPE at 0.23177, 0.11523, 0.3394 and 5.2331, respectively. Both ensemble methods usually achieve similar results, but HTK-SVR-BO is efficient regarding the R^2 score. All these observations suggest that a split ratio of 70:30 is more effective than other split ratios.

Table 4. *Comparison of ensemble models and SVR in terms of regression metrics.*

Metrics	Split ratio / Model	90:10	80:20	70:30	60:40	50:50
MAE	**LPR-SVR-BO**	0.38659	0.34869	0.27424	0.29837	0.39202
	HTK-SVR-BO	**0.34043**	**0.27292**	**0.23177**	**0.28601**	**0.34022**
	SVR-Linear	2.6853	1.2128	0.6699	0.75213	0.76322
	SVR-RBF	9.6496	4.9962	4.489	4.6531	4.685
MSE	**LPR-SVR-BO**	0.25521	0.1962	0.13392	0.16708	**0.29476**
	HTK-SVR-BO	**0.17484**	**0.132**	**0.11523**	**0.13394**	0.29637
	SVR-Linear	13.649	2.8216	0.61304	0.72516	0.65493
	SVR-RBF	350.25	42.024	28.1	31.282	35.332
RMSE	**LPR-SVR-BO**	0.50519	0.44295	0.36595	0.40876	**0.54292**
	HTK-SVR-BO	**0.41814**	**0.36332**	**0.33946**	**0.36598**	0.5444
	SVR-Linear	3.6906	1.6792	0.78285	0.85143	0.80857
	SVR-RBF	18.691	6.4801	5.2981	5.5899	5.9313
MAPE	**LPR-SVR-BO**	19.4172	19.119	13.7732	23.3106	26.2524
	HTK-SVR-BO	**19.1098**	**8.7267**	**5.7244**	**22.2055**	**5.2331**
	SVR-Linear	141.76	127.34	105.93	131.26	150.39
	SVR-RBF	557.25	501.2	498.09	532.42	556.85
R^2 score	**LPR-SVR-BO**	0.99995	0.99996	0.99997	0.99996	0.99993
	HTK-SVR-BO	**0.99996**	**0.99997**	**0.99997**	**0.99997**	**0.99993**
	SVR-Linear	0.97206	0.97688	0.99291	0.99175	0.99207
	SVR-RBF	0.28696	0.6561	0.67589	0.64442	0.57727

6 Conclusion and Future Directions

This paper presented a comprehensive approach to improving stock price prediction accuracy and reliability for MSFT. An ensemble SVR model was prepared by incorporating the hyperbolic tangent kernel and BO for hyperparameter tuning of kernel parameters. We trained and validated our proposed method with various split ratios. Our proposed method could be trained and validated with various split ratios, but the 70:30 train test ratio yielded the best results for our model. In our comparison, HTK-SVR-BO achieved better regression metrics (MAE, MSE, RMSE, R^2 score and MAPE) than LPR-SVR-BO, with 0.23177, 0.11523, 0.33946, 0.99997 and 5.2331, while LPR-SVR-BO achieved 0.27424, 0.13392, 0.36595, 0.99997 and 13.7732 respectively. Based on the regression metrics, the simulation results prove that the HTK-SVR-BO model is superior to both the LPR-SVR-BO and individual SVR models.

This comparison indicates that the proposed model is more accurate than its counterparts. In this approach, non-linearities can be captured more effectively and features such as handling noisy objective functions, adaptive search and global optimization are incorporated. Consequently, the model may generalize to previously unknown data more effectively, resulting in better accuracy when forecasting future stocks. The present work can be extended by sophisticated techniques, such as hybrid kernels, improved optimization and meta-learning regression algorithms for making more comprehensive financial decisions.

Additional Information and Declarations

Acknowledgments: It has been a privilege and a joy working with Prof. N. Geethanjali throughout my research journey. I am grateful for her guidance, encouragement, and unwavering support. I have been constantly inspired by her dedication and expertise. As a supervisor, she has also been a role model, consistently demonstrating the highest standards of professionalism and commitment.

Conflict of Interests: The authors declare no conflict of interest.

Author Contributions: S. R. T: Conceptualization, Data curation, Investigation, Methodology, Writing – Original draft preparation, Visualization. G. N.: Supervision, Software, Validation, Writing – Reviewing and Editing.

Data Availability: The data that support the findings of this study are openly available in Yahoo finance at https://finance.yahoo.com/quote/MSFT?p=MSFT&.tsrc=fin-srch

References

Alam, S., Sultana, N., & Hossain, S. M. Z. (2021). Bayesian optimization algorithm-based support vector regression analysis for estimation of shear capacity of FRP reinforced concrete members. *Applied Soft Computing*, 105, 107281. https://doi.org/10.1016/j.asoc.2021.107281

Dash, R. K., Nguyen, T. N., Cengiz, K., & Sharma, A. (2021). Fine-tuned support vector regression model for stock predictions. *Neural Computing and Applications*, 35(32), 23295–23309. https://doi.org/10.1007/s00521-021-05842-w

Divina, F., Gilson, A., Gómez-Vela, F., Torres, M., & Torres, J. F. (2018). Stacking Ensemble Learning for Short-Term Electricity Consumption Forecasting. *Energies*, 11(4), 949. https://doi.org/10.3390/en11040949

Gao, X., Shan, C., Hu, C., Niu, Z., & Liu, Z. (2019). An adaptive ensemble machine learning model for intrusion detection. *IEEE Access*, 7, 82512–82521. https://doi.org/10.1109/access.2019.2923640

Gohar, U., Biswas, S., & Rajan, H. (2023). Towards Understanding Fairness and its Composition in Ensemble Machine Learning. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, (pp. 1533–1545). IEEE. https://doi.org/10.1109/ICSE48619.2023.00133

Hu, Y., Su, P., Hao, X., & Tang, F. (2009). The Long-Term Predictive Effect of SVM Financial Crisis Early-Warning Model. In *2009 IEEE First International Conference on Information Science and Engineering*, (pp. 4573–4576). IEEE. https://doi.org/10.1109/ICISE.2009.1230

Huang Y., Deng C., Zhang X. and Bao Y. (2022). Forecasting of stock price index using support vector regression with multivariate empirical mode decomposition. *Journal of Systems and Information Technology*, 24(2), 75–95. https://doi.org/10.1108/jsit-12-2019-0262

Kenfack, P. J., Khan, A., Kazmi, S. M. A., Hussain, R., Oracevic, A., & Khattak, A. M. (2021). Impact of Model Ensemble On the Fairness of Classifiers in Machine Learning. In *2021 IEEE International Conference on Applied Artificial Intelligence (ICAPAI)*, (pp. 1-6). IEEE. https://doi.org/10.1109/ICAPAI49758.2021.9462068

Kumar G., Jain S., & Singh U. P. (2021). Stock market forecasting using computational intelligence: a survey. *Archives of Computational Methods in Engineering*, 28(3), 1069–1101. https://doi.org/10.1007/s11831-020-09413-5

Lin Y., Guo H., & Hu J. (2013). An SVM-based approach for stock market trend prediction. In *2013 IEEE International Joint Conference on Neural Networks (IJCNN)*, (pp. 1-7). IEEE. https://doi.org/10.1109/IJCNN.2013.6706743

Majhi, B., & Anish, C. (2015). Multiobjective optimization based adaptive models with fuzzy decision making for stock market forecasting. *Neurocomputing*, 167, 502–511. https://doi.org/10.1016/j.neucom.2015.04.044

Rubio, L., & Alba, K. (2022). Forecasting selected Colombian shares using a hybrid ARIMA-SVR model. *Mathematics*, 10(13), 2181. https://doi.org/10.3390/math10132181

Sharma D., Singla S. K., & Sohal A. K. (2021). Stock market prediction using ARIMA, ANN and SVR. In *ICCCE 2020*, (pp. 1081–1092). Springer. https://doi.org/10.1007/978-981-15-7961-5_100

Siddique M., Mohanty S., & Panda D. (2019). A hybrid model for forecasting of stock value of tata steel using orthogonal forward selection, support vector regression and teaching learning based optimization. *Far East Journal of Mathematical Sciences*, 113(1), 95–114. https://doi.org/10.17654/ms113010095

Singh, S., Madan, T. K., Kumar, J., & Singh, A. K. (2019). Stock market forecasting using machine learning: today and tomorrow. In *2019 IEEE 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, (pp. 738–745). IEEE. https://doi.org/10.1109/ICICICT46008.2019.8993160

Snoek J., Larochelle H., & Adams R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *2021 ACM Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS'12)*, (pp. 2951–2959). ACM. https://dl.acm.org/doi/10.5555/2999325.2999464

Thumu, S. R., & Nellore, G. (2021). Real-World Research Applications and Directions of Machine Learning Algorithms. *Design Engineering*, 2021(9), 14256–14275.

Turner R., Eriksson D., McCourt M.J., Kiili J., Laaksonen E., Xu Z., & Guyon I. M. (2021). Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020. *Proceedings of Machine Learning Research*, 133, 3–26. http://proceedings.mlr.press/v133/turner21a/turner21a.pdf

Virigineni, A., Tanuj, M., Mani, A., & Subramani, R. (2022). Stock forecasting using HMM and SVR. In *2022 IEEE International Conference on International Conference on Communication, Computing and Internet of Things*, (pp. 1–7). IEEE. https://doi.org/10.1109/IC3IOT53935.2022.9767969

Wu J., Chen X. Y., Zhang H., Xiong L. D., Lei H., & Deng S. H. (2019). Hyperparameter optimization for machine learning models based on Bayesian optimization. *Journal of Electronic Science and Technology*, 17(1), 26–40. https://doi.org/10.11989/JEST.1674-862X.80904120

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415, 295–316. https://doi.org/10.1016/j.neucom.2020.07.061

Editorial record: The article has been peer-reviewed. First submission received on 12 September 2023. Revisions received on 15 October 2023 and 16 November 2023. Accepted for publication on 7 December 2023. The editor in charge of coordinating the peer-review of this manuscript and approving it for publication was Zdenek Smutny.

Acta Informatica Pragensia is published by Prague University of Economics and Business, Czech Republic.

ISSN: 1805-4951