Research Article

Using Stacked Generalization Model in Stock Price Forecasting: A Comparative Analysis on BIST100 Index

Ahmed İhsan ŞİMŞEK[1]

[1] Assist. Prof., Firat University, aisimsek@firat.edu.tr, ORCID: 0000-0002-2900-3032

Abstract: Investing in financial markets requires an adequately planned approach and decision-making process for both individual and institutional investors. The volatility of financial markets is influenced by intricate and constantly evolving factors, prompting investors, analysts, and financial experts to employ progressively sophisticated and data-centric methodologies to precisely forecast future price swings. Deep learning models for stock price prediction demonstrate the ability to comprehend intricate connections by amalgamating extensive datasets. The objective of this essay is to employ various machine learning models using daily data from the BIST100 index, a prominent financial indicator in Turkey. The models under question encompass Support Vector Regression (SVR), K-Nearest Neighbors (KNN), Random Forest (RF), XGBoost and Stacked Generalization. The models' prediction skills were evaluated using RMSE, MSE, MAE, and R2 performance indicators. Based on the observed results, the Stacked Generalization model demonstrated greater performance in making predictions for the analyzed dataset. These findings offer valuable insights that should be considered when selecting models for similar analyses in the future.

Keywords: BIST100, Stacked generalization, Stock prediction, Decision support, Time series
Jel Codes: C45, C67, G17

Hisse Senedi Fiyat Tahmininde Stacked Generalization Modelinin Kullanımı: BIST100 Endeksi Üzerine Karşılaştırmalı Bir Analiz

Öz: Finansal piyasalara yatırım yapmak hem bireysel hem de kurumsal yatırımcılar için önemli bir strateji ve karar alma sürecini gerektirmektedir. Finansal piyasaların oynaklığı, karmaşık ve sürekli gelişen faktörlerden etkilenerek yatırımcıları, analistleri ve finansal uzmanları gelecekteki fiyat dalgalanmalarını kesin olarak tahmin etmek için giderek daha karmaşık ve veri merkezli metodolojiler kullanmaya teşvik etmektedir. Hisse senedi fiyat tahmini için derin öğrenme modelleri, kapsamlı veri kümelerini birleştirerek karmaşık bağlantıları kavrama yeteneği sayesinde giderek daha çok tercih edilmektedir. Bu makalenin amacı, Türkiye'nin önemli finansal göstergelerinden biri olan BIST100 endeksine ait 04.01.2010-29.11.2023 tarihleri arasındaki 3494 günlük veri setini kullanarak en iyi tahmin yeteneğine sahip modeli belirlemektir. Bu kapsamda SVR, KNN, RF, XGBoost ve Stacked Generalization modeli kullanılmıştır. Modellerin tahmin becerileri RMSE, MSE, MAE ve R2 performans göstergeleri kullanılarak değerlendirilmiştir. Modellerin tahmin performanslarından elde edilen sonuçlara göre, Stacked generalization modeli analiz edilen veri kümesi için tahminler yapmada daha yüksek performans göstermiştir. Bu bulgular gelecekte benzer analizler için model seçerken dikkate alınması gereken değerli bilgiler sunmaktadır.

Anahtar Kelimeler: BIST100, Stacked generalization, Hisse senedi tahmini, Karar destek, Zaman serisi
Jel Kodları: C45, C67, G17

Cite: Şimşek, A. İ. (2025). Using stacked generalization model in stock price forecasting: A comparative analysis on BIST100 index. *Fiscaoeconomia, 9*(1), 305-322. https://doi.org/10.25295/fsecon.1444407

Submitted: 28.02.2024
Accepted: 27.09.2024

1. Introduction

In today's financial markets, stock prices have a volatile, complex and constantly changing structure due to the interaction of many factors. The intricacy of this situation has prompted investors, analysts, and financial experts to use increasingly advanced and data-oriented techniques for properly forecasting future price fluctuations. Currently, deep learning technologies are recognized as a potent instrument for comprehending and forecasting the intricacies in financial markets. Deep learning is a specialized field within artificial intelligence that excels in its capacity to analyze vast quantities of data and comprehend intricate patterns. Conventional financial models frequently rely on specific assumptions, which are subject to potential alterations as time progresses. Deep learning surpasses these models by demonstrating its capacity to adjust and acquire knowledge in response to evolving dynamics across time. Deep learning models for stock price prediction possess the capacity to provide more precise and dependable forecasts by assimilating vast datasets and comprehending intricate correlations. Investing in financial markets is typically determined by considerations of risk and return. Deep learning can enhance this decision-making process, assess risk variables more efficiently, and assist investors in making better-informed selections. Deep learning-based stock price prediction in this context enables investors to enhance portfolio management and respond promptly to financial market volatility.

Many different methods are used in forecasting financial markets. These include ARIMA (Box et al., 2015; Kulkarni et al., 2020), GARCH (Gabriel & Ugochukwu, 2012, Chen & Chen, 2015), genetic algorithm (Nikolopoulos & Fellrath, 1994; Kim & Han, 2000), ANN (Roh, 2007; Vijaya et al., 2016; Gurjar et al., 2018; Gaytan et al., 2022; Kurani, 2023); machine learning algorithms (Umer et al., 2019; Khan et al., 2020; Vijh et al., 2020; Rouf et al., 2021; Soni et al., 2022; Kumbure et al., 2022; Ashtiani & Raahmei, 2023; Jorgenson et al., 2023), deep learning algorithms (Jiang, 2021; Mehtab et al., 2021; Hu et al., 2021; Li&Pan, 2022; Shah et al., 2023; Muhammad et al., 2023; Mukherjee et al., 2023) methods can be listed. Today, both individual and institutional investors have progressively employed machine and deep learning methodologies as instruments for guiding their investment strategies within financial markets. Machine learning methods offer the chance to evaluate large amounts of past data and detect patterns that can be effectively used to predict future stock prices. These approaches employ algorithms that can acquire knowledge from current data and adjust their predictive abilities based on new information. Machine learning algorithms has the ability to identify connections and patterns within past stock price data that may be difficult for individual experts to see, provided they are trained on this data. Deep neural methodologies possess the capability to precisely forecast forthcoming stock prices by proficiently modeling intricate nonlinear data patterns. The main goal of a neural network is to create a mapping and emphasize the pertinent information needed to understand a function, hence enhancing the accuracy of the predicted output (Mandic & Chambers, 2001). Financial research employs machine learning and deep learning techniques. Nevertheless, every approach possesses its own benefits and limitations. Currently, stacked generalization models can be employed to enhance the accuracy of predicting. Stacking is a well-established method in machine learning, especially for integrating models. This approach involves using a meta-learning model to effectively combine and incorporate the output generated by a set of basic models. The act of merging various models is frequently known as "model blending" or simply "blending" when the final decision component is represented by a linear model. The notion of layered regression, sometimes known as stacking, was first introduced by Wolpert in 1992. In this specific methodology, the dataset is partitioned into equal segments by a random process. In th-fold cross-validation, one specific set is designated for testing, while the rest sets are allocated for training. By employing subsets that consist of training and testing pairs, various learning models are used to make predictions. Therefore, the previously indicated predictions are used as metadata to create the meta-model. The meta-model employs a winner-takes-all strategy to provide the most precise

prediction (Ganaine et al., 2022). BIST100 index, which represents the top 100 companies listed on the Borsa Istanbul (Istanbul Stock Exchange) based on their market capitalization and liquidity. The BIST100 index is a key benchmark in the Turkish financial market, providing insights into the overall health and trends of the stock market in Turkey. Figure 1 shows the daily historical chart of the BIST100 index.

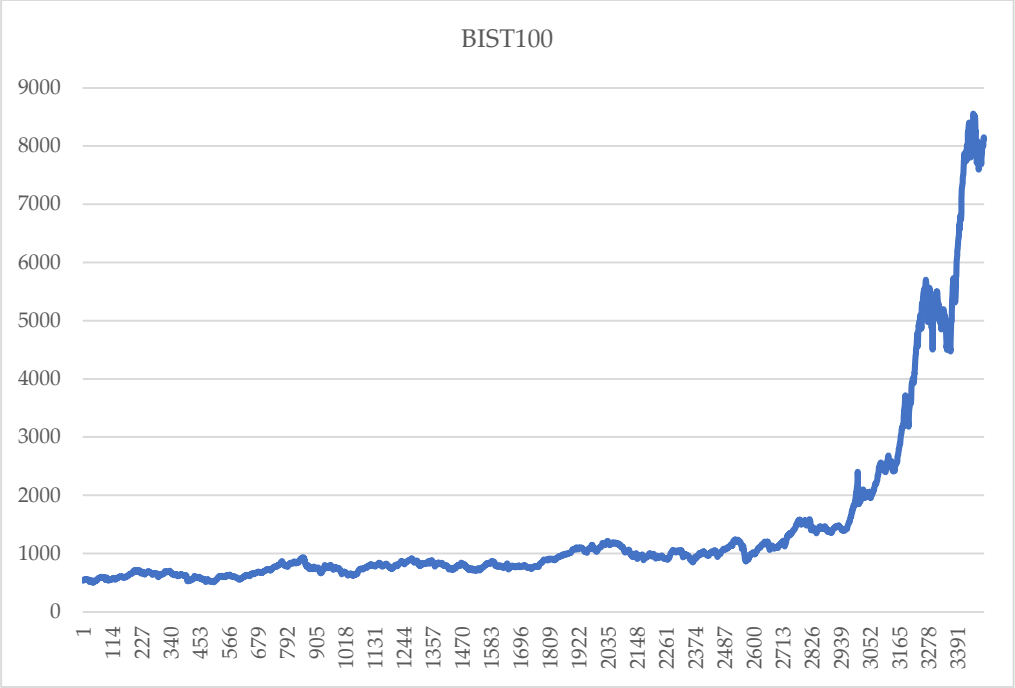

Figure 1. BIST100 Index Price Data from 04.01.2010 to 29.11.2023

Investing in financial markets necessitates a significant strategy and decision-making process for both individual and institutional investors. Performing precise and dependable data analysis is a crucial component in forecasting future price fluctuations in this procedure. Upon reviewing the literature, it is evident that numerous forecasts have been created concerning the BIST100 index utilizing various machine and deep learning techniques (Oztekin et al., 2016; Yakut & Gemici, 2017; Yigit et al., 2020; Kilimci & Duvar, 2020; Sabanci et al., 2023; Armagan, 2023). Nevertheless, none of this research employed the stacking generalization model. In the literature, most of the models used in stock price prediction are based on a single model type, which generally results in limited prediction accuracy. The Stacked Generalization model allows for stronger and more general predictions to be made by combining different model types. This model provides a structure that can compensate for the weaknesses of existing models. Studies conducted using the stacked generalization model can produce superior prediction performance. This study aims to fill the existing gap in the literature.

This article focuses on utilizing different machine learning models to forecast prices based on the daily data of the BIST100 index, a prominent financial indicator in Turkey. The data used spans from 04.01.2010 to 29.11.2023. The main objective of our article is to forecast BIST100 index prices by employing a range of machine and deep learning algorithms, including SVR, KNN, RF, XGBoost and Stacked Generalization. These algorithms will be assessed to determine their predictive capabilities. These models possess distinct characteristics and capabilities, making them effective tools for comprehending and forecasting intricate patterns inside financial markets. The subsequent sections of our paper will adhere to the following fundamental procedures. The study compile our dataset by gathering daily pricing data of the BIST100 index. Furthermore, the analysis preprocess this data to ensure its compatibility with machine

learning models. This article construct several models, including SVR, KNN, RF, XGBoost and Stacked Generalization. Subsequently, this study assess the performance of each model individually. The paper employs RMSE, MSE, MAE, and R2 performance metrics to assess the predictive prowess of our models. These measurements will provide us with insights into the degree of success achieved by each model. Through a comprehensive assessment of the merits, drawbacks, and efficacy of each model, This study address potential applications in forecasting prices in financial markets.

2. Literature Review

Forecasting stock prices in financial markets is a fundamental challenge for investors and finance experts. While standard financial models were previously employed for these forecasts, it is evident that deep learning methods are currently gaining prominence in the realm of financial research. Lately, some academics have been concentrating on utilizing ML approaches to predict stock price patterns. The study conducted by Lv et al. (2019) examines the performance of stocks in large-scale stock datasets, comparing the effects of transaction costs and no transaction costs. Recent advancements in advanced machine learning techniques, such as deep learning, have led to the development of novel algorithms for analyzing and predicting time series data. Kelany et al. (2020) introduce a novel model for assessing the effectiveness of deep learning and other prediction methods in relation to the risk factor of stocks. Artificial neural networks are often regarded as precise prediction models in machine learning. However, their effectiveness in forecasting stock market prices is limited. Enhancing the precision of the model is necessary. The study conducted by Polamuri et al. (2020) focuses on developing a hybrid prediction system for stock market prices using multiple models. A framework called the Stock Market Prices Prediction Framework is offered for this purpose. Additionally, there exists a wide range of literature regarding the technical analysis of stock values. The primary objective of this study is to discern patterns within stock price fluctuations and capitalize on them for financial gain. Mehtab et al. (2020) propose a hybrid approach that integrates machine learning and deep learning approaches to predict stock values. Furthermore, there exists a wide range of literature dedicated to the technical analysis of stock prices. The primary aim of this literature is to discern patterns in the movements of stock prices and capitalize on them for financial gain. Machine and deep learning approaches becoming increasingly prominent as effective techniques for analyzing financial data, encompassing textual, numerical, and graphical information. The study conducted by Li et al. (2021) demonstrates that ensemble deep learning techniques outperform traditional methods in accurately forecasting future stock price patterns. These novel instruments provide exceptional support to investors in making well-informed investment decisions. Sidogi et al. (2022) present a method where they create 12-month rolling characteristics using commonly used sell-side recommendations. These features include analyst coverage, point accuracy, and directional correctness. The authors ensure that their method avoids any biases related to future predictions. Sidogi et al. (2022) present a groundbreaking "AI analyst" that combines predictive characteristics from several analysts through the utilization of machine learning methods. Typically, machine learning is used to forecast the future values of publicly traded stocks by analyzing both current and historical price data. An analysis was performed on the methodologies used to analyze time series data, ranging from conventional linear modeling methods to automated machine learning frameworks, which also include deep learning models (Alsharef et al., 2022). The study conducted by Htun et al. (2023) thoroughly analyzes 32 research projects that utilize a combination of feature analysis and machine learning techniques in various stock market applications. Investors are utilizing deep learning models to predict and evaluate stock and foreign currency markets, taking use of the advantages of artificial intelligence. The usefulness of Machine Learning, Deep Learning, Reinforcement Learning, and Deep Reinforcement Learning in the field of Quantitative Finance (QF) and the Stock Market has been clarified by the authors (Sahu et al., 2023).

Researchers have turned to deep learning to develop an intelligent trading framework in response to the unpredictable and volatility character of the financial market. Yang et al. (2023) suggest employing CNN as the fundamental component of this framework because to its capability to capture the spatial interdependence (i.e., the relationship between rows and columns) within the input data. In contrast to previous deep learning-based trading frameworks, they have devised a unique normalization approach for preparing stock data. Zeng et al. (2023) suggest utilizing the capabilities of CNNs and Transformers to effectively represent and analyze both immediate and prolonged interdependencies in a time series. This approach aims to predict whether the price will increase, decrease, or remain unchanged (flat) in the future. The study emphasized the effectiveness of the proposed methodology in predicting the short-term changes in stock values of S&P 500 companies, when compared to commonly used statistical and deep learning methods. In another study conducted for Borsa Istanbul index prediction, Decision Trees, RF, KNN, Naive Bayes, logistic Regression, SVM and ANN methods were used. In this study, where various technical indicators were used between 01.01.2012-31.12.2021, it was determined that the best prediction result for BIST100 index prediction was produced by the ANN method (Ayyıldız & İskenderoğlu, 2023). Zhang et al. (2023) present an innovative two-stage methodology that employs decomposition and re-estimation techniques to forecast prices in worldwide carbon markets. Once the subsequences have been broken down, utilize six machine learning and deep learning algorithms to combine the data and enable the prediction of the ultimate carbon price values. The increasing growth and high demand in the financial market have led to significant interest in predicting stock price trends among both academia and industry. A deep learning-based system for anticipating stock opening prices was developed and implemented in Mali in 2023. The research conducted by Du et al. (2023) examines the utilization of knowledge-based reinforcement learning and estimation of distribution approach to tackle the issue of flexible job shop scheduling. This paper presents a hybrid multi-objective optimization strategy that integrates the estimation of distribution algorithm with deep Q-network to solve this problem. A tridimensional solution representation is employed to convey the processing sequence, machine allocation, and processing speed allocation.

In this study, in addition to the SVR, KNN, RF and XGBoost models, which are frequently used in the literature in stock price prediction, the stacked generalization method, which increases the prediction performance of existing models, was applied. With stacked generalization, higher accuracy predictions can be achieved by combining the strengths of more than one model. As mentioned above, these methods are frequently used in the literature. The SVR method is also used in the literature for stock price prediction due to its success in capturing nonlinear relationships and processing high-dimensional data. For example, Huang et al. (2020) used the SVR method and SVR-EMD methods for S&P500, HSI and SSE index prediction. Oukhouya & El Himdi (2023) predicted the price of the Morocco Stock Index 20 index using four different machine learning methods, including the SVR method. According to the results, it was revealed that the SVR method together with MLP produced better prediction results than other methods. Jiang et al. (2024) made stock price prediction by comparing 9 different methods. According to the results obtained, it was revealed that the CNN-based method gave the best results. Soepriyanto (2021) compared the KNN method with the Naive Bayes method in stock price prediction to show its applicability. Vinay & Mahaveerakannan (2023) compared the KNN method with logistic regression and showed that the KNN method gave better results than logistic regression in stock price prediction. Ayyıldız & İskerderoğlu (2024) applied various methods such as Decision Trees, KNN, Logistic Regression, Naive Bayes, SV and ANN in their study on the stock indices of G7 countries. According to the results obtained, it was observed that ANN, Logistic Regression and SVM algorithms made the highest accuracy predictions. In studies where stacked generalization methods were used for stock price prediction, it was observed that this method gave better results than classical machine and deep learning methods. For

example, Bhanja & Das (2019) compared the performance of the LSTM-CNN based stacked model they proposed for Combay Stock Exchange prediction with other models in the literature and revealed that the stacked model gave better performance. In another study, Gyamerah et al. (2019) compared KNN, AdaBoost and Stacked models for Nairobi Stock Exchange prediction. It was observed that the stacking model produced better prediction results in all error metrics. Massaoudi et al. (2021) compared XGboost, LightGBM and MLP methods with the stacking model. It was observed that the stacking model gave better results in all error metrics. Mahboob et al. (2023) proposed an MLP-LSTM based stacking model for Karachi Stock Price Index prediction. According to the obtained results, it was observed that CNN-LSTM-RNN stacking model gave the best prediction result with 98.9%. Mandal et al. (2023) proposed an SVR-MLR based stacked generalization model in their study for 6 companies listed in the Indian National Stock Exchange. They compared the proposed model with the study of Henrique et al. (2018). In this study, much better prediction results were produced than the results obtained from the SVR model, which gave the best prediction result. Digi et al. (2024) proposed an LSTM based stacked model and tested this model for 10 companies listed in the Indonesia Stock Exchange. The predictive performance of the stacked model was measured with an average RMSE of 0.00885, average MAE of 0.00800, average MAPE of 0.02496, and an average R2 of 0.9597.

Stock price prediction models in the literature are usually limited to a single method or are compared with a few in a limited way. These approaches usually focus on the weaknesses of existing methods rather than improving the predictive power of the model. One of the reasons why the Stacked Generalization model is rarely preferred in such studies is the difficulty in how to combine the outputs of multiple models and which meta-model to use in this process. The main reason why the Stacked Generalization model was preferred in this study is to create a more robust and generalizable prediction model by combining the strengths of different model types. This model offers flexibility and prediction power that a single model cannot have. One of the biggest challenges in the implementation of the Stacked Generalization model is the selection of the meta-model. There is not enough work on this subject in the literature, and this study aims to fill this gap.

3. Data & Methodology

This study utilized a dataset that spanned from January 4, 2010, to November 29, 2023. The dataset comprised 3494 consecutive days of data for the BIST100 index. 80% of the data was used for training and 20% for testing. The data was acquired from the website investing.com. The analysis utilized five different methodologies. The algorithms being considered are SVR, XGBoost, Random Forest, KNN, and Stacked-Generalization. SVR tries to predict target values by drawing a line or hyperplane from the given data. The goal is to place this line closest to the data. During training, the model uses the training data to determine this line. When training on test data, the model uses this line to make predictions based on new data. The model makes predictions for new inputs based on the line it has learned. KNN is actually a model without a training phase. It stores the training data directly and does not create any models during training. During testing, the model calculates a similarity between a new data point and its nearest neighbors and makes a prediction based on the labels of these neighbors. When new data arrives, the model compares this data with the training data it has stored and makes a prediction based on the labels of its nearest neighbors. Stacked-Generalization aims to make better predictions by using multiple models together. First, each model is trained separately. Then, the prediction results of these models are taken by a new model and the final prediction is made by combining these results. During testing, each model makes a prediction on the test data. These predictions are then combined by the meta model. The model achieves more accurate results by combining the predictions of multiple models. Hyperparameter settings of the models used in the study were determined using “gridsearch” and

"randomsearch" algorithms. In addition, cross-validation was used to increase the generalization ability of the models.

The evaluation of the method's efficacy involved assessing various statistical variables, such as RMSE, MSE, MAE, and R2. RMSE is a measure calculated by taking the square root of the mean square of the difference between the predicted values and the true values. In general, lower RMSE values indicate that the model is performing better. MSE is the mean square of the difference between the predicted values and the true values. A lower MSE indicates that the model is performing well. MAE is the mean absolute value of the difference between the predicted values and the true values. A lower MAE indicates that the model is performing better. R^2 is a statistical measure that shows how much the independent variables explain the variance of the dependent variable. R^2 close to 1 indicates that the model is performing well and generally indicates a good fit. Python 3.11 was used in the study for both data tampering and developing models. The "Pandas" library in Python was used to read data from Excel, and "NumPy" was used for numerical operations and working with arrays. "scikit-learn" was used to create, train and evaluate machine learning models. Graphic visualization was done with Matplotlib and stacking operations were done with MLXtend. The mathematical procedures for computing the statistical parameters given above are represented by Equations 1, 2, 3, and 4 (Harishkumar et al., 2020).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{1}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\hat{y}_i)}{N}} \tag{2}$$

$$MAE = \frac{\sum_{i=1}^{n}|y_i-x_i|}{n} \tag{3}$$

$$R^2 = 1 - \frac{\sum_i(y_i-\hat{y}_i)^2}{\sum_i(y_i-\mu)^2} \tag{4}$$

3.1. SVR Model

The SVR method was first introduced by Cortes & Vapnik in the 1990s (Cortes & Vapnik, 1995). Subsequently, a regression technique known as support vector machine for regression (Vapnik, 1999) was developed. The SVR technique was first developed as a method for classifying data. SVR approach is employed to determine the best hyperplane that efficiently separates multiple parameters. The ideal hyperplane is defined by its largest margin, guaranteeing an equal separation from all variables (Zouzou & Çıtakoğlu, 2021). Equation 12-18 provides a clear overview of the progressive stages of the SVR approach (Xu et al., 2014).

$$f(x) = \omega\,\Phi(x) + b \tag{12}$$

$$L(f(x), y, \varepsilon) = f(x) = \begin{cases} 0 & |y - f(x)| \le \varepsilon \\ |y - f(x)| - \varepsilon & |y - f(x)| > \varepsilon \end{cases} \tag{13}$$

$$\begin{cases} Min.\frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}\xi_i \\ sub.t. \begin{cases} y_i - \omega\Phi(x_i) - b \le \varepsilon + \xi_i \\ -y_i + \omega\Phi(x_i) + b \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \end{cases} \tag{14}$$

$$\omega^* = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\,\Phi(x_i) \tag{15}$$

$$b^* = \frac{1}{N_{nsv}}\left\{\sum_{0<\alpha_i<C}\left[y_i - \sum_{x_i\in SV}(\alpha_i - \alpha_i^*)K(x_i, x_j) - \varepsilon\right] + \sum_{0<\alpha_i<C}\left[y_i - \sum_{x_j\in SV}(\alpha_j - \alpha_j^*)K(x_i, x_j) + \varepsilon\right]\right\} \tag{16}$$

$$K(x_i, x_j) = exp\left(-\frac{\|x-x_i\|^2}{2\sigma^2}\right) \tag{17}$$

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x_i, x) + b^* \tag{18}$$

The main objective of SVR is to discover a linear regression function, denoted as f(x), within a high-dimensional space. Let x be an element from the set of real numbers, representing the sample vector. The function Φ exhibits non-linear properties in its

mapping. Introducing a linear insensitivity loss function, denoted as L(f (x), y, ε), improves the resilience of the optimization issue. Equation 13 provides a numerical representation of this loss function. Equation 14 defines the variables x_i and y_i as the input vector and output value, respectively. The variables are linked to a particular serial number, indicated by the variable i. Both x_i and y_i are elements of the set of real numbers, denoted as R. The input vector has a dimension of d. Here, the variable d represents the number of items in an input vector. The variable n represents the quantity of training samples. The symbol ε represents the measure of regression precision. The variable C indicates a punitive coefficient that measures the magnitude of the penalty imposed on a data sample if its error exceeds the threshold value ε. The variables ξ_i and ξ_i^* are employed as slack variables to impose penalties on the complexity of the fitting parameters. To determine the values of ω and b, it is crucial to solve the optimization issue described in Equation 15 and 16. The symbol N_{nsv} represents the count of support vectors that have been identified. The Lagrange multipliers, denoted as α_i and α_i^*, must satisfy the condition of being non-negative. In this specific scenario, Equation 17 utilizes the kernel function denoted as $K(x_i, x_j)$. The Gaussian kernel function, renowned for its impressive capacity to generalize, is chosen. Equation 18 represents the final regression function.

3.2. KNN Model

The K-nearest neighbor method is a machine learning algorithm that is commonly considered to be straightforward to implement (Aha et al., 1991). The core premise of the KNN algorithm is to categorize an unknown data point by comparing its resemblance to previous data points with known classifications. The technique use a k-value to determine the nearest neighbor, considering only the provided number of closest neighbors. The k-value is utilized to allocate a category to a specific data point in a sample. The use of numerous nearest neighbors is crucial in defining the class to which a specific data point belongs. The name KNN is derived from the fundamental reasoning behind it. The strategy is categorized as a memory-based technique because it necessitates the storage of data points in memory during execution (Phyu, 2009; Amra & Maghari, 2017). This approach is versatile and can be utilized for problems involving regression as well as classification. The underlying principle of KNN is that related items tend to cluster together, and the algorithm identifies these clusters by measuring the spacing between distinct data points. KNN is considered a lazy learner method since it stores the data and conducts operations on it during the stage of classification, instead of learning directly from the training dataset. The classification of a new data point is determined by the majority consensus of its nearest neighbors. Adjusting the value of k can significantly affect the precision of the algorithm (Ahmed et al., 2023). The formulas used in the KNN method are given below (Puspitiari & Rustam, 2018).

$$d(p,x) = \sqrt{\sum_{j=1}^{m}\left(x_j - p_j\right)^2} \tag{22}$$

$$\hat{y} = \frac{1}{k}\sum_{i=1}^{k} y_i \tag{23}$$

3.3. Random Forest Model

Random Forest is an ensemble learning algorithm and improves generalization capabilities by using random sampling techniques to overcome the limitations of decision trees (Wang et al., 2020). By combining predictions from multiple sub-decision trees, Random Forest reduces the influence of outliers in individual trees, thus resulting in more accurate overall predictions (OuYang, 2024). The Random Forest model has advantages such as its ability to manage lots of input variables efficiently, which makes it appropriate for situations involving high-dimensional data. Furthermore, Random Forest models have a reputation for being resilient as it comes to handling missing data, continuing to produce accurate predictions even though there is insufficient data (Xing et al., 2022). The

algorithm performs well in a variety of prediction tasks since it can successfully detect nonlinear connections among variables (Liu, 2024). In addition, the random forest model has some disadvantages. A disadvantage is that training the model takes a lot of time because it requires creating a lot of decision trees, that can cause processing times to increase. Furthermore, Random Forest performs well in minimizing overfitting problems, but compared to simpler models like linear regression, it may produce a more complex model framework that is difficult to comprehend due to the ensemble learning concept it utilizes (Elmuna, 2023).

There are basically four stages in the Random Forest model. First, a different sub-data set is created for each decision tree by randomly and repeatedly selecting samples from each training dataset. Then, these decision trees are divided into branches according to the selected features and grown until they reach the specified criterion (Liu et al., 2024). Then, the created decision trees make predictions for the test data set and all of these predictions are averaged and the final prediction is obtained. Then, the prediction performance of the Random Forest model is evaluated using performance metrics such as MSE, MAE, RMSE, R2 (Hutagalung et al., 2023).

3.4. XGBoost Model

XGBoost works with the gradient boosting technique, which is one of the ensemble learning methods among machine learning algorithms. This algorithm builds weak prediction models sequentially, usually using decision trees, and combines them to create a strong and accurate model. XGBoost is known for its ability to effectively process complex data sets and provide high prediction performance (Hadaya et al., 2022). XGBoost model has some advantages. Firstly, its ability to minimize bias and variance-related errors, which helps in preventing overfitting and improving model performance through its iterative learning process (Yu et al., 2022). To get the best performance out of XGBoost on different datasets, it's essential to fine-tune its hyperparameters. This means tweaking factors like the learning rate, tree depth, and regularization to enhance the model's accuracy and ensure it adapts well to the specific characteristics of the data (Schimohr et al., 2022). Additionally, XGBoost provides valuable insights into the importance of each variable, helping users understand how different features influence the model's predictions. This feature is especially useful for understanding the data's underlying patterns and making better decisions about which features to focus on and how to optimize the model (Burgess-Hull et al., 2022). The stages of the XGBoost model are shown below in Equation 24-31.

$$\hat{y}_i = \emptyset(x_i) = \sum_{k=1}^{K} f_k(y_i), f_k \in \mathcal{F} \quad (24)$$

$$min\, L^{(t)}(\, y_i, \hat{y}_i{}^{(t)}) = min\left(\sum_{i=1}^{n} \iota\left(y_i, \hat{y}_i{}^{(t)}\right) + \sum_{k=1}^{t} \Omega(f_k)\right) \quad (25)$$

$$\Omega(\, f) = \gamma T + \frac{1}{2}\lambda w^2 \quad (26)$$

$$min\, L^{(t)} = min\left(\sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)\right] + \Omega(f_t)\right) \quad (27)$$

$$g_i = \partial_{\hat{y}_i{}^{(t-1)}} l(y_i, y_i^{t-1}) \quad (28)$$

$$h_i = \partial^2_{y_i{}^{t-1}}\, l(y_i, y_i^{t-1}) \quad (29)$$

$$w_j^* = -\frac{\sum g_i}{\sum h_i + \lambda} \quad (30)$$

$$obj^* = -\frac{1}{2}\sum_{j=1}^{T} \frac{(\sum g_i)^2}{\sum h_i + \lambda} + \gamma . T \quad (31)$$

In Equation 24, the predicted value ($\hat{y}_i$) in the XGBoost model is the sum of the outputs of all trees. Here, f_k is the function of the k-th tree and $\mathcal{F}$ represents the space containing the functions of all possible decision trees. In other words, each f_k is a decision tree that transforms a data point into a prediction. Equation 25 expresses the optimization problem of the XGBoost model. Equation 26 defines the complexity of the created XGBoost model. Equation 27 defines the loss function. Equations 28 and 29 take the first and second order derivatives of this loss function. Equation 30 finds the optimal weights

of the leaves in the decision tree. Finally, the total loss function optimized with Equation 31 balances the prediction performance of the model and obtains the best prediction results.

3.5. Stacked Generalization Model

Stacked generalization, also known as stacking, is an ensemble machine learning approach that combines the forecasts of numerous machine learning models in a logical manner. The staking models are composed of many level-0 models, also known as base models, and a level-1 model, referred to as the meta-model. The meta-model aggregates the forecasts generated by the initial models. The input datasets of the base model and meta-model are not the same; the meta-model takes the outcomes of the base models (predictions on the train datasets) as its inputs (Fereydooni & Mahootchi, 2023). Stacking involves aggregating the predictions of multiple base learners by utilizing another machine learning approaches. This approach involves fully utilizing all available approaches to analyze the learning set and make predictions based on it. Undoubtedly, a meta-classifier constructed using the Stacking technique possesses the capability to outperform some ensemble models based on trees, such as RF and XGBoost, as well as cutting-edge deep learning models. However, the financial industry has paid less heed to this algorithm (Jiang et al., 2020). Figure 2 displays the stacked generalization model structure using in this study.

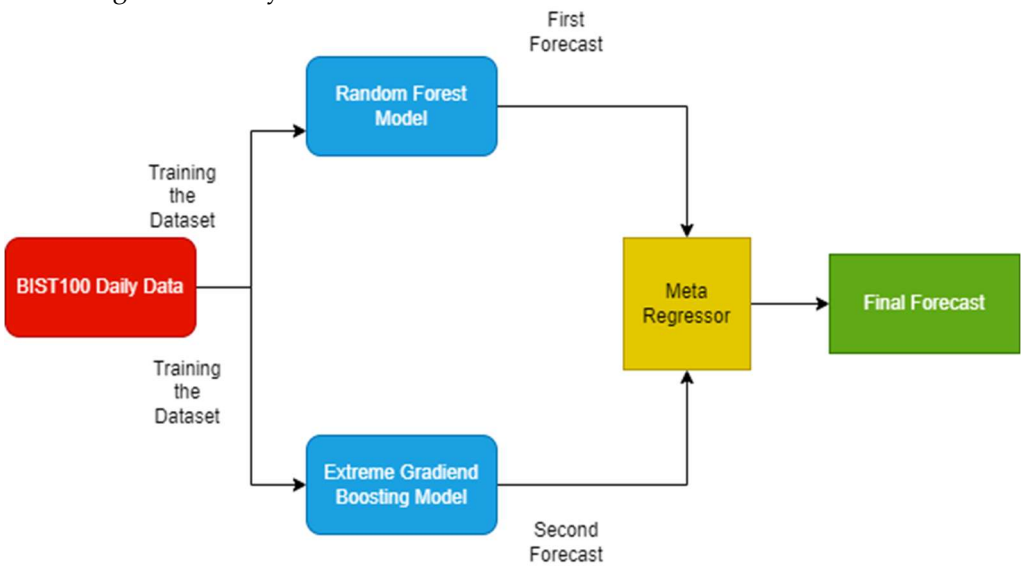

Figure 2. Stacked Generalization Model

This study involves the creation of a stacking regression model. Stacking is an ensemble modeling approach that utilizes the forecasts of various initial models and merges these predictions with a meta-model. The following are the procedural steps employed in this particular instance. To begin with, two fundamental regression models were constructed utilizing the RandomForestRegressor and GradientBoostingRegressor algorithms utilizing the "mlxtend.regressor" module. Random Forest is an ensemble learning method that makes predictions by building multiple decision trees. The model builds many decision trees during training and makes the final prediction by average prediction in regression problems and majority vote in classification problems. XGBoost is an implementation of gradient boosting algorithms and is a high-performance modeling technique. This model builds weak learners sequentially to reduce the error rate. A StackingRegressor is a model that is designed to aggregate the forecasts made by multiple basic models. The primary models consist of RandomForestRegressor and GradientBoostingRegressor. The meta-regression algorithm employed as LinearRegression. The StackingRegressor model was trained using the provided training

data. The StackingRegressor model, which has undergone training, generated forecasts for both the training and test datasets. Subsequently, the outcomes prior to the scaling procedure are transformed and measurement metrics for error are computed for both the training and test datasets. After completing these procedures, the forecasts generated by the RandomForestRegressor and GradientBoostingRegressor models were merged utilizing the LinearRegression meta-model. The efficacy of these merged forecasts was subsequently assessed.

4. Results and Discussion

Investing in financial markets requires a significant planning and choice-making procedure for both individual and institutional investors. Performing precise and dependable data analysis is a crucial component in forecasting future price fluctuations in this procedure. The BIST 100 is a stock market index that encompasses the 100 largest publicly traded firms in Turkey, which are listed on the Borsa Istanbul. The significance of this index lies in its ability to serve as a key metric that accurately represents the overall performance of the Turkish economy. The BIST 100 index comprises equities of prominent corporations spanning several sectors. The success of companies in the BIST 100 index is typically contingent upon the Turkish economy, the global financial market, fiscal policies, and other relevant variables. The index offers investors a gauge to comprehend and assess the economic condition in Turkey. This paper focuses on utilizing machine learning models to forecast prices based on a dataset consisting of 3494 daily observations of the BIST100 index. The BIST100 index, a key financial metric in Turkey, will be referred to as BIST100 in the dataset. The dataset spans from January 4, 2010, to November 29, 2023. The main objective of our article is to forecast the prices of the BIST100 index by employing various machine and deep learning approaches, including KNN, XGBoost, SVR, Random Forest and, Stacked Generalization.

Table 1 shows the statistical coefficients obtained by the models during the training phase.

Table 1. Statistical Results of Training Phase

Model	Training MSE	Training RMSE	Training MAE	Training R2
KNN	0.00017	0.01284	0.00945	0.99520
XGBoost	0.00004	0.00638	0.00478	0.99882
SVR	0.00131	0.03614	0.03019	0.96193
Random Forest	0.00002	0.00455	0.00346	0.99940
Stacked	0.00002	0.00428	0.00319	0.99947

The performance measures of the models in the table are illustrated with important metrics used to evaluate the success of the models. Upon examining the data from Table 1, the KNN model performs quite well with low error rates and a high R^2 value. However, it has slightly higher error values compared to other models. XGBoost is one of the models with the lowest error values and shows a very high R^2 value. This shows that the model performs very well on the training data and explains the data very well. The SVR model has higher error rates than other models and its R^2 value is relatively lower. This shows that the model is weaker in explaining the data than other models. The Random Forest model shows excellent performance with very low error rates and a very high R^2 value. It can be said that it works almost without errors on the training data. The Stacked model, as a combination of XGBoost and Random Forest models, offers the lowest error rates and the highest R^2 value. This shows that the model explains the data very well and makes the most accurate predictions. In general, all models perform admirably, however the Stacked Model appears to be the most suitable for the dataset. This model outperforms other models in terms of R2, RMSE, MSE, and MAE metrics. Following the completion of the

training phase, the model proceeded the testing phase. Table 2 illustrates the statistical coefficients obtained by the models during the testing phase.

Table 2. Statistical Results of Testing Phase

Model	Test MSE	Test RMSE	Test MAE	Test R2
KNN	0.00730	0.70259	0.86474	0.81007
XGBoost	0.00071	0.06938	0.08523	0.95809
SVR	1.81776	0.96947	0.56569	0.83350
Random Forest	0.00004	0.00690	0.08498	0.89291
Stacked	0.00002	0.00688	0.00385	0.98024

Performance on the test set is critical to assessing the model's ability to generalize, that is, how well it can perform on new and unseen data. When Table 2 is examined, the KNN model shows an average performance on the test set. The R^2 value is quite good, but the error rates are higher than the other models. This may indicate that the KNN model is less sensitive to new data than the other models. XGBoost shows a very good performance with low error rates and a high R^2 value. It is seen that this model is generally strong and has a high generalization ability. The SVR model has much higher error rates than the other models in terms of MSE and RMSE. Although the R^2 value is at a reasonable level, the generalization ability of SVR is weak when compared to the other models. It is seen that this model does not perform well on the test set. Random Forest shows a strong performance on the test set with very low error rates and a good R^2 value. The generalization ability of this model is quite high and gives good results against new data. The Stacked model has the lowest error rates and the highest R^2 value among all the models. The performance of this model on the test set is excellent and stands out as the model with the highest generalization ability. As a result, the Stacked model performs best on the test set with the lowest error rates and the highest R^2 value. This shows that the model is very sensitive to new data and very successful in its predictions.

Finally, cross-validation was performed in the study. K-Fold Cross-Validation is a method used to evaluate the generalization ability of a model. It divides the data into K equal parts and each part is used as the test set, while the remaining K-1 parts are used as the training set. This process is repeated K times and all the performance metrics obtained are averaged. This method helps to evaluate the performance of the model on different data sets and to understand whether the model is prone to overfitting. The 5-fold cross-validation results are shown in Table 3.

Table 3. Statistical Results of Kfold-5 Phase

Model	5-Fold CV Mean MSE	5-Fold CV Mean RMSE	5-Fold CV Mean MAE	5-Fold CV Mean R2
KNN	0.02641	0.97660	0.95354	0.78512
XGBoost	0.00060	0.00478	0.10753	0.93389
SVR	1.93815	1.14367	3.12354	0.82500
Random Forest	0.00004	0.00745	0.09512	0.87401
Stacked	0.00003	0.00743	0.00512	0.97209

Table 3 shows the average performance metrics obtained with 5-Fold Cross-Validation of five different models. When Table 3 is examined, the KNN model has higher error rates than the other models and its R^2 value is relatively low. This shows that the performance of the model on different data sets is not consistent and its generalization ability is limited. The XGBoost model shows a strong performance with low error rates

and a high R^2 value. The generalization ability of this model is quite good and gives consistently good results on different data sets. The SVR model has quite high error rates and a low R^2 value. This model produces very high errors during cross-validation and its generalization ability seems weak. This shows that the model cannot show consistent performance on different data sets and is potentially prone to over-learning. The Random Forest model has low error rates and a good R^2 value. The generalization ability of this model is quite good, but it performs slightly lower compared to XGBoost and Stacked models. Stacked model has the lowest error rates and the highest R^2 value among all models. This model also gives the best results during cross-validation and its generalization ability is very high. In conclusion, Stacked model has the lowest error rates and the highest R^2 value among all models. This model also gives the best results during cross-validation and its generalization ability is very high.

The 30-day forecast results of the Stacked model, which gave the best results, between 30.11.2023 and 10.01.2024 are shown in Table 4.

Table 4. Prediction For Next 30 Days Generated by Stacked Model

Date	Predicted BIST100 INDEX
11.29.2023	7,828
11.30.2023	7,819
12.01.2023	7,797
12.04.2023	7,801
12.05.2023	7,809
12.06.2023	7,820
12.07.2023	7,830
12.08.2023	7,836
12.11.2023	7,836
12.12.2023	7,836
12.13.2023	7,818
12.14.2023	7,811
12.15.2023	7,822
12.18.2023	7,832
12.19.2023	7,839
12.20.2023	7,834
12.21.2023	7,850
12.22.2023	7,888
12.25.2023	7,929
12.26.2023	7,961
12.27.2023	7,982
12.28.2023	7,973
12.29.2023	7,970
1.02.2024	7,961
1.03.2024	7,982
1.04.2024	7,985
1.05.2024	7,978
1.08.2024	7,984
1.09.2024	7,984
1.10.2024	7,976

Figure 3 shows the training and test graphs of the stacked model.

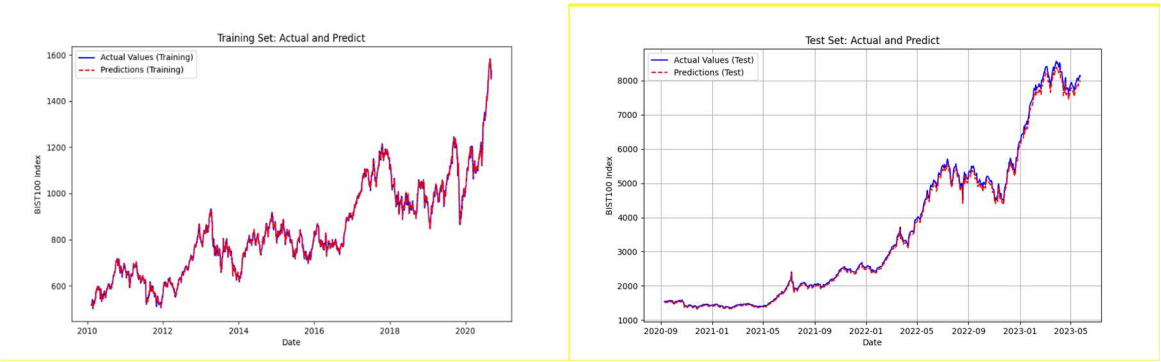

Figure 3. Stacked Generalization Model Training and Test Results

When the literature is examined, it is observed that the stacked generalization model generally performs better than single algorithms (Bhanja & Das, 2019; Gyamerah et al., 2019; Massaoudi et al., 2021; Mahboob et al., 2023; Mandal et al., 2023; Digi et al., 2024). In this study, similar to other studies in the literature, it was observed that the stacking model produced better results than other single algorithms. When compared with the studies in the literature, it is observed that hyperparameter optimization was not performed in some studies (Bhanja & Das, 2019; Gyamerah et al., 2019; Massaoudi et al., 2021). In addition, in this study, cross-validation was performed in addition to other studies in the literature and the generalization ability of the proposed models was demonstrated.

The results obtained have important practical implications for both academic and practitioners in financial markets. The superior performance of the Stacked Generalization model used in this study compared to other models indicates that this model can be used more widely in future stock price predictions. These results can help investors obtain more reliable and accurate predictions in their decision-making processes. The use of the Stacked Generalization model has the potential to reduce uncertainty in investment strategies. This model combines the strengths of various algorithms in different market conditions and provides more balanced and general estimates. Especially in markets with high volatility, the integration of this model into investment strategies can minimize risks and maximize potential returns in portfolio management. These findings also provide important guidance for financial analysts and strategy development teams. Especially when working with large data sets and complex market dynamics, the use of methods such as the Stacked Generalization model can strengthen analysis processes and enable more accurate strategic decisions. This can contribute to the creation of a more effective and efficient investment environment in financial markets in general.

To summarize, this study has demonstrated that the Stacked model outperforms and exhibits robust performance in its predictions for the examined dataset. These findings offer crucial insights that might be considered when selecting models for similar analyses in the future. Further examination is required to further analyze additional optimal combinations of hyperparameters for each model, which could potentially enhance the performance of the models. Utilizing a more extensive data set can enhance the capacity of models to make accurate predictions across a wider range of scenarios. Furthermore, there exist several machine and deep learning techniques apart from the models employed in this investigation. The performance metrics derived from the application of these strategies can be compared to those produced in this study. In addition, the inclusion of novel functionalities that integrate investor behavior or enhance the modeling of such behaviors might facilitate a more comprehensive comprehension of market dynamics.

References

Aha, D., Kibler, D.W. & Albert, M.K. (1991). Instance-based learning algorithms. *Mach Learn, 6*, 37–66

Ahmed, R., Bibi, M. & Syed, S. (2023). Improving heart disease prediction accuracy using a hybrid machine learning approach: A comparative study of SVM and KNN algorithms. *International Journal of Computations, Information and Manufacturing (IJCIM), 3*(1), 49-54.

Amra, I. A. A. & Maghari, A. Y. (2017, May). Students performance prediction using KNN and Naïve Bayesian. *2017 8th International Conference on Information Technology (ICIT)* (p. 909-913). IEEE.

Armağan, İ. Ü. (2023). Price prediction of The Borsa Istanbul Banks Index with traditional methods and artificial neural networks. *Borsa Istanbul Review*.

Ashtiani, M. N. & Raahmei, B. (2023). News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Systems with Applications*, 119509.

Ayyildiz, N. & Iskenderoglu, O. (2023). Prediction of stock index movement directions using machine learning methods: An application on developing countries. *Journal of Financial Economics and Banking, 4*(2), 68-78.

Ayyildiz, N. & Iskenderoglu, O. (2024). How effective is machine learning in stock market predictions?. *Heliyon, 10*(2). https://doi.org/10.1016/j.heliyon.2024.e24123

Bhanja, S. & Das, A. (2019). Deep learning-based integrated stacked model for the stock market prediction. *Int. J. Eng. Adv. Technol, 9*(1), 5167-5174.

Box, G. E., Jenkins, G. M., Reinsel, G. C. & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Burgess-Hull, A. J., Brooks, C. S., Epstein, D. H., Gandhi, D. & Oviedo, E. (2022). Using machine learning to predict treatment adherence in patients on medication for opioid use disorder. *Journal of Addiction Medicine, 17*(1), 28-34. https://doi.org/10.1097/adm.0000000000001019

Chen, M. Y. & Chen, B. T. (2015). A hybrid fuzzy time series model based on granular computing for stock price forecasting. *Information Sciences, 294*, 227-241.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*, 273-297.

Diqi, M., Ordiyasa, I. W. & Hamzah, H. (2024). Enhancing stock price prediction using stacked long short-term memory. *IT Journal Research and Development, 8*(2), 164-174.

Du, Y., Li, J. Q., Chen, X. L., Duan, P. Y. & Pan, Q. K. (2022). Knowledge-based reinforcement learning and estimation of distribution algorithm for flexible job shop scheduling problem. *IEEE Transactions on Emerging Topics in Computational Intelligence*.

Elmuna, E. A. F., Chamidy, T. & Nugroho, F. (2023). Optimization of the random forest method using principal component analysis to predict house prices. *International Journal of Advances in Data and Information Systems, 4*(2), 155-166. https://doi.org/10.25008/ijadis.v4i2.1290

Fereydooni, A. & Mahootchi, M. (2023). An algorithmic trading system based on a stacked generalization model and hidden Markov model in the foreign exchange market. *Global Finance Journal, 56*, 100825.

Gabriel, A. M. & Ugochukwu, W. M. (2012). Volatility estimation and stock price prediction in the Nigerian stock market. *International Journal of Financial Research, 3*(1), 2.

Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M. & Suganthan, P. N. (2022). Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence, 115,* 105151.

Gaytan, J. C. T., Ateeq, K., Rafiuddin, A., Alzoubi, H. M., Ghazal, T. M., Ahanger, T. A., Chaudhary, S. & Viju, G. K. (2022). AI-based prediction of capital structure: Performance comparison of ANN SVM and LR models. *Computational Intelligence and Neurosciene, 2022*.

Gurjar, M., Naik, P., Mujumdar, G. & Vaidya, T. (2018). Stock market prediction using ANN. *International Research Journal of Engineering and Technology, 5*(3), 2758-61.

Gyamerah, S. A., Ngare, P. & Ikpe, D. (2019, May). On stock market movement prediction via stacking ensemble learning method. *2019 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)* (p. 1-8). IEEE.

Hadaya, J., Verma, A., Sanaiha, Y., Ramezani, R., Qadir, N. & Benharash, P. (2022). Machine learning-based modeling of acute respiratory failure following emergency general surgery operations. *Plos One, 17*(4), e0267733. https://doi.org/10.1371/journal.pone.0267733

Harishkumar, K. S., Yogesh, K. M. & Gad, I. (2020). Forecasting air pollution particulate matter (PM2. 5) using machine learning regression models. *Procedia Computer Science, 171*, 2057-2066.

Henrique, B. M., Sobreiro, V. A. & Kimura, H. (2018). Stock price prediction using support vector regression on daily and up to the minute prices. *The Journal of Finance and Data Science, 4*(3), 183-201.

Htun, H. H., Biehl, M. & Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. *Financial Innovation, 9*(1), 26.

Hu, Z., Zhao, Y. & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. *Applied System Innovation, 4*(1), 9.

Huang, Y., Deng, C., Zhang, X. & Bao, Y. (2020). 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

Hutagalung, S. V., Yennimar, Y., Rumapea, E. R., Hia, M. J. G., Sembiring, T. & Manday, D. R. (2023). Comparison of support vector regression and random forest regression algorithms on gold price predictions. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima(JUSIKOM PRIMA), 7*(1), 255-262. https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v7i1.4125

Jiang, M., Liu, J., Zhang, L. & Liu, C. (2020). An improved Stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms. *Physica A: Statistical Mechanics and its Applications, 541*, 122272.

Jiang, W. (2021). Applications of deep learning in stock market prediction: recent progress. *Expert Systems with Applications, 184*, 115537.

Jorgenson, D. W., Weitzman, M. L., ZXhang, Y. X., Haxo, Y. M. & Mat, Y. X. (2023). Can neural networks predict stock market?. *AC Investment Research Journal, 220*(44).

Kelany, O., Aly, S. & Ismail, M. A. (2020). Deep learning model for financial time series prediction. *2020 14th International Conference on Innovations in Information Technology (IIT)* (p. 120-125). IEEE.

Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H. & Alfakeeh, A. S. (2020). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, 1-24.

Kilimci, Z. H. & Duvar, R. (2020). An efficient word embedding and deep learning based model to forecast the direction of stock exchange market using Twitter and financial news sites: A case of Istanbul stock exchange (BIST 100). *IEEE Access, 8*, 188186-188198.

Kim, K. J. & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert systems with Applications, 19*(2), 125-132.

Kulkarni, M., Jadha, A. & Dhingra, D. (2020). Time series data analysis for stock market prediction. *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*.

Kumbure, M. M., Lohrmann, C., Luukka, P. & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications, 197*, 116659.

Kurani, A., Doshi, P., Vakharia, A. & Shah, M. (2023). A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. *Annals of Data Science, 10*(1), 183-208.

Li, Y. & Pan, Y. (2022). A novel ensemble deep learning model for stock prediction based on stock prices and news. *International Journal of Data Science and Analytics*, 1-11.

Liu, C. (2024). Implied volatility forecasting for American options based on random forest regressor, linear regression model. *Advances in Economics Management and Political Sciences, 85*(1), 154-160. https://doi.org/10.54254/2754-1169/85/20240867

Lv, D., Yuan, S., Li, M. & Xiang, Y. (2019). An empirical study of machine learning algorithms for stock daily trading strategy. *Mathematical Problems in Engineering*.

Mahboob, K., Shahbaz, M. H., Ali, F. & Qamar, R. (2023). Predicting the Karachi Stock Price index with an Enhanced multi-layered Sequential Stacked Long-Short-Term Memory Model. *VFAST Transactions on Software Engineering, 11*(2), 249-255.

Mandal, U., Chakraborty, A., Mahato, P. & Das, G. (2023). LinVec: A stacked ensemble machine learning architecture for analysis and forecasting of time-series data. *Indian Journal of Science and Technology, 16*(8), 570-582.

Mandic, D. P. & Chambers, J. (2001). *Recurrent neural networks for prediction: Learning algorithms architectures and stability*. John Wiley & Sons, Inc.

Massaoudi, M., Refaat, S. S., Chihi, I., Trabelsi, M., Oueslati, F. S. & Abu-Rub, H. (2021). A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for Short-Term Load Forecasting. *Energy, 214*, 118874.

Mehtab, S. & Sen, J. (2020). Stock price prediction using convolutional neural networks on a multivariate timeseries. arXiv preprint arXiv:2001.09769.

Mehtab, S., Sen, J. & Dasgupta, S. (2020). Robust analysis of stock price time series using CNN and LSTM-based deep learning models. *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)* (p. 1481-1486).

Mehtab, S., Sen, J. & Dutta, A. (2021). Stock price prediction using machine learning and LSTM-based deep learning models. *Machine Learning and Metaheuristics Algorithms, and Applications: Second Symposium, SoMMA 2020* (p. 88-106). Chennai, India, October 14–17, 2020, Revised Selected Papers 2. Springer Singapore.

Muhammad, T., Aftab, A. B., Ibrahim, M., Ahsan, M. M., Muhu, M. M., Khan, S. I., & Alam, M. S. (2023). Transformer-based deep learning model for stock price prediction: A case study on Bangladesh stock market. International Journal of Computational Intelligence and Applications, 2350013.

Mukherjee, S., Sadhukhan, B., Sarkar, N., Roy, D. & De, S. (2023). Stock market prediction using deep learning algorithms. *CAAI Transactions on Intelligence Technology, 8*(1), 82-94.

Nikolopoulos, C. & Fellrath, P. (1994). A hybrid expert system for investment advising. *Expert Systems, 11*(4), 245-250.

Oukhouya, H. & El Himdi, K. (2023, April). Comparing machine learning methods—SVR, XGBOOST, LSTM, and MLP—for forecasting the Moroccan stock market. *Computer Sciences & Mathematics Forum* (Vol. 7, No. 1, p. 39). MDPI.

OuYang, Z. (2024). Research on the diamond price prediction based on linear regression, decision tree and random forest. *Highlights in Business, Economics and Management, 24*, 248-257. https://doi.org/10.54097/13ccwv59

Oztekin, A., Kizilaslan, R., Freund, S. & Iseri, A. (2016). A data analytic approach to forecasting daily stock returns in an emerging market. *European Journal of Operational Research, 253*(3), 697-710.

Phyu, T. N. (2009). Survey of classification techniques in data mining. *Proceedings of the International Multiconference of Engineers and Computer Scientists* (Vol. 1, No. 5, pp. 727-731). Citeseer.

Polamuri, S. R., Srinivas, K. & Mohan, A. K. (2020). Multi model-based hybrid prediction algorithm (MM-HPA) for stock market prices prediction framework (SMPPF). *Arabian Journal for Science and Engineering, 45*, 10493-10509.

Puspitasari, D. A. & Rustam, Z. (2018). Application of SVM-KNN using SVR as feature selection on stock analysis for Indonesia stock exchange. *AIP Conference Proceedings* (Vol. 2023, No. 1). AIP Publishing.

Roh, T. H. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications, 33*(4), 916-922.

Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S. & Kim, H. C. (2021). Stock market prediction using machine learning techniques: a decade survey on methodologies, recent developments, and future directions. *Electronics, 10*(21), 2717.

Sabancı, D., Kılıçarslan, S. & Adem, K. (2023). An application on forecasting for stock market prices: hybrid of some metaheuristic algorithms with multivariate adaptive regression splines. *International Journal of Intelligent Computing and Cybernetics, 16*(4), 847-866.

Sahu, S. K., Mokhade, A. & Bokde, N. D. (2023). An overview of machine learning, deep learning, and reinforcement learning-based techniques in quantitative finance: Recent progress and challenges. *Applied Sciences, 13*(3), 1956.

Schimohr, K., Doebler, P. & Scheiner, J. (2022). Prediction of bike-sharing trip counts: comparing parametric spatial regression models to a geographically weighted XGBOOST algorithm. *Geographical Analysis, 55*(4), 651-684. https://doi.org/10.1111/gean.12354

Shah, J., Vaidya, D. & Shah, M. (2022). A comprehensive review on multiple hybrid deep learning approaches for stock prediction. *Intelligent Systems with Applications*, 200111.

Sidogi, T., Mongwe, W. T., Mbuvha, R. & Marwala, T. (2022). Fusing sell-side analyst bidirectional forecasts using machine learning. *IEEE Access, 10*, 76966-76974.

Soepriyanto, B. (2021). Comparative analysis of K-NN and naïve bayes methods to predict stock prices. *International Journal of Computer and Information System (IJCIS), 2*(2), 49-53. https://doi.org/10.29040/ijcis.v2i2.32

Soni, P., Tewari, Y. & Krishnan, D. (2022). Machine learning approaches in stock price prediction: A systematic review. *Journal of Physics: Conference Series* (Vol. 2161, No. 1, p. 012065). IOP Publishing.

Umer, M., Awais, M. & Muzammul, M. (2019). Stock market prediction using machine learning (ML) algorithms. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 8*(4), 97-116.

Vapnik, V. (1999). *The nature of statistical learning theory*. Springer science & business media.

Vijaya, P., Raju, G. & Ray, S. K. (2016). Artificial neural network-based merging score for Meta search engine. *Journal of Central South University, 23*, 2604-2615.

Vijh, M., Chandola, D., Tikkiwal, V. A. & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia Computer Science, 167*, 599-606.

Vinay, N. & Mahaveerakannan, R. (2023). Analyze the lack of accuracy in stock price prediction using novel k-nearest neighbors regression compared with logistic regression to improve accuracy. *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)* (p. 1-5). IEEE.

Wang, J., Li, C., Li, J., Qin, S., Wang, J., Chen, Z., … & Wang, G. (2020). Development and internal validation of risk prediction model of metabolic syndrome in oil workers. *BMC Public Health, 20*(1). https://doi.org/10.1186/s12889-020-09921-w

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks, 5*(2), 241-259.

Xing, F., Luo, R., Liu, M., Zhou, Z., Xiang, Z. & Duan, X. (2022). A new random forest algorithm-based prediction model of post-operative mortality in geriatric patients with hip fractures. *Frontiers in Medicine, 9*. https://doi.org/10.3389/fmed.2022.829977

Xu, Z., Gao, Y. & Jin, Y. (2014). Application of an optimized SVR model of machine learning. *International Journal of Multimedia and Ubiquitous Engineering, 9*(6), 67-80.

Yakut, E. & Gemici, E. (2017). Predicting stock return classification through LR, C5. 0, CART and SVM methods, and comparing the methods used: an application at BIST in Turkey. *Ege Academic Review, 17*(4), 461-479.

Yang, K., Zhang, G., Bi, C., Guan, Q., Xu, H. & Xu, S. (2023). Improving CNN-base stock trading by considering data heterogeneity and burst. arXiv preprint arXiv:2303.09407.

Yiğit, Ö. E., Alp, S. & Ersoy, Ö. Z. (2020). Prediction of BIST price indices: A comparative study between traditional and deep learning methods. *Sigma Journal of Engineering and Natural Sciences, 38*(4), 1693-1704.

Yu, W., Lu, Y., Shou, H., Xu, H., Shi, L., Geng, X., … & Song, T. (2022). A 5-year survival status prognosis of nonmetastatic cervical cancer patients through machine learning algorithms. *Cancer Medicine, 12*(6), 6867-6876. https://doi.org/10.1002/cam4.5477

Zeng, Z., Kaur, R., Siddagangappa, S., Rahimi, S., Balch, T. & Veloso, M. (2023). Financial time series forecasting using CNN and Transformer. arXiv preprint arXiv:2304.04912.

Zhang, X., Li, Z., Zhao, Y. & Wang, L. (2023). Carbon trading and COVID-19: A hybrid machine learning approach for international carbon price forecasting. *Annals of Operations Research*, 1-29.

Zouzou, Y., & Çıtakoğlu, H. (2021). Reference evapotranspiration prediction from limited climatic variables using support vector machines and Gaussian processes. *Avrupa Bilim ve Teknoloji Dergisi*, (28), 346-351.

Conflict of Interest: None.

Funding: None.

Ethical Approval

Author Contributions: Ahmed İhsan ŞİMŞEK (100%)

Çıkar Çatışması: Yoktur.

Finansal Destek: Yoktur.

Etik Onay: Yoktur.

Yazar Katkısı: Ahmed İhsan ŞİMŞEK (%100)