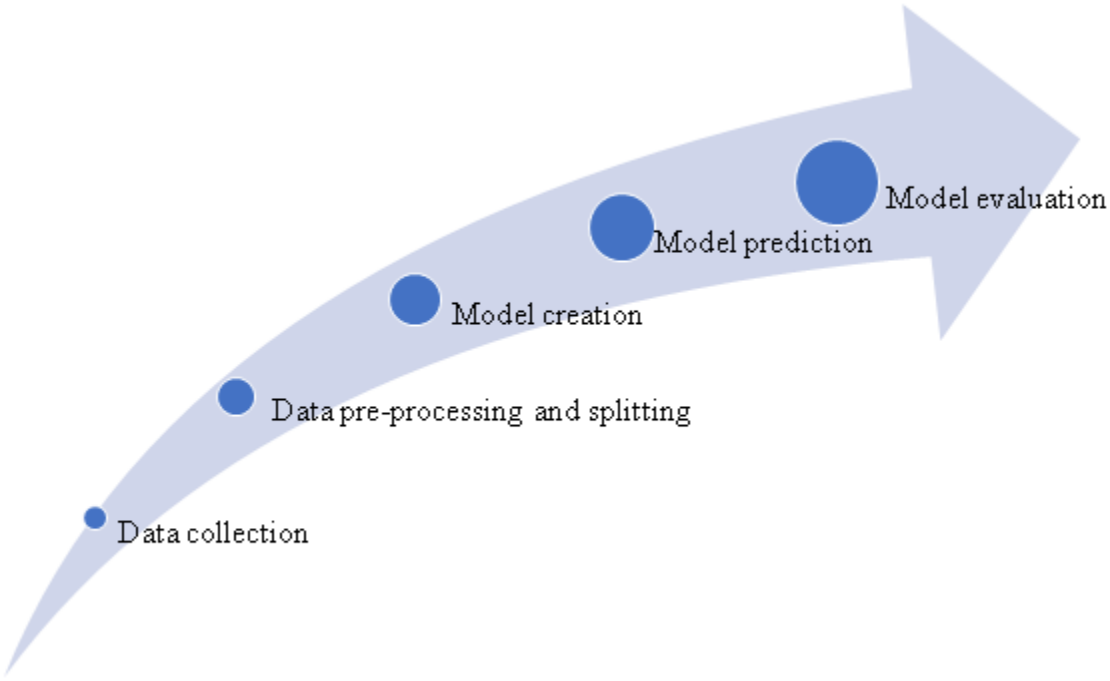
Prediction of Values of Borsa Istanbul Forest, Paper, and Printing Index Using Machine Learning Methods

İlker Akyüz,[a] Kinyas Polat,[b] Selahattin Bardak,[c,*] and Nadir Ersen [d]

*Corresponding author: sbardak@sinop.edu.tr

DOI:10.15376/biores.19.3.5141-5157

GRAPHICAL ABSTRACT



Prediction of Values of Borsa Istanbul Forest, Paper, and Printing Index Using Machine Learning Methods

İlker Akyüz,[a] Kinyas Polat,[b] Selahattin Bardak,[c,*] and Nadir Ersen [d]

It is difficult to predict index values or stock prices with a single financial formula. They are affected by many factors, such as political conditions, global economy, unexpected events, market anomalies, and the characteristics of the relevant companies, and many computer science techniques are being used to make more accurate predictions about them. This study aimed to predict the values of the XKAGT index by using the monthly closing values of the Borsa Istanbul (BIST) Forestry, Paper and Printing (XKAGT) index between 2002 and 2023, and the machine learning techniques artificial neural networks (ANN), random forest (RF), k-nearest neighbor (KNN), and gradient boosting machine (GBM). Furthermore, the performances of four machine learning techniques were compared. Factors affecting stock prices are generally classified as macroeconomic and microeconomic factors. As a result of examining the studies on determining the macroeconomic factors affecting the stock markets, 10 macroeconomic factors were determined as input. The macroeconomic variables used were crude oil price, exchange rate of USD/TRY, dollar index, BIST100 index, gold price, money supply (M2), S&P 500 index, US 10-year bond interest, export-import coverage rate in the forest products sector, and deposits interest rate. It was determined that all machine learning techniques used in the study performed successfully in predicting the index value, but the k-nearest neighbor algorithm showed the best performance with R^2=0.996, RMSE=71.36, and a MAE of 40.8. Therefore, in line with the current variables, investors can make analyzes using any of the ANN, RF, KNN, and GBM techniques to predict the future index value, which will lead them to accurate results.

DOI:10.15376/biores.19.3.5141-5157

Keywords: Machine learning; Forest industry; Index prediction; XKAGT

Contact information: a: Department of Forest Industry Engineering, Karadeniz Technical University, 61100, Trabzon, Türkiye; b: Department of Metallurgical and Materials Engineering, Sinop University, 57000, Sinop, Türkiye; c: Department of Computer Engineering, Sinop University, 57000, Sinop, Türkiye; d: Department of Forestry, Artvin Çoruh University, 08000, Artvin, Türkiye;
**Corresponding author: sbardak@sinop.edu.tr*

INTRODUCTION

The forest-based sector has significant economic and social value on both a national and global scale. This sector, which constitutes 25% of Türkiye's total manufacturing industry, affects many sectors directly and indirectly (Kahraman 2023).In addition, this sector is constantly developing its product diversity with new technologies, and it has a 3.3% share of Türkiye's total exports (Yıldız and Erdoğan 2020; Aegean Exporters' Associations 2024).

The performance of companies in the forest-based sector can be a valuable indicator for understanding the general economic situation and industrial developments. At this

point, the Forestry, Paper, Printing Index (XKAGT) emerges as an important tool to monitor the performance of companies operating in these sectors. As of April 30, 2024, 17 companies are traded in the BIST Forest, Paper, Printing (XKAGT) index, which started to be calculated on Dec 27, 1996. The number of investors of the BIST XKAGT index was 625,342 as of April 30, 2024, and the number of investors of BIST XKAGT index constitutes 7.65% of BIST all index (Borsa Istanbul 2023; Kahraman 2023).

XKAGT directly measures the positive or negative changes that may occur in the wood-based sector and plays important roles in managing strategic investment decisions in this sector (Kaderli *et al.* 2013).

Increasing instability in stock markets means that investors assume risks. Increased instability can also be a situation that can lead investors to losses (Kurt and Senal 2018).Therefore, the prediction of XKAGT values is very important for investors, sector actors, and people and organizations connected to the sector. Since stock indices reflect the values and performances of the stocks in the portfolio, they are an effective indicator in investors' decisions to invest or abandon investment in the capital markets. Additionally, indices are one of the indicators that foreign investors use to analyze the general economic and political performance of the country in which they intend to invest (Bayramoğlu 2007).

Predicting how the stock market will perform is one of the most difficult things to do (Cavalcante *et al.* 2016). Nowadays, artificial intelligence is used to solve uncertainties in matters such as measuring index performances and predicting stock prices (Özcan Akdağ *et al.* 2022). Machine learning is a field of artificial intelligence that is being used as a game changer in predicting stock market index prices (Ravikumar and Saraf 2020). Research on stock market index predictions based on machine learning algorithms has attracted more attention recently (Hu *et al.* 2022). Various machine learning methods, such as artificial neural networks, decision trees, k-nearest neighbors, support vector machines, support vector regression, bagging, gradient boosting, random forest, *etc.*, are being used to predict stock market movement (Ceylan 2018). There is no best algorithm that can predict the stock market movement with high accuracy. Therefore, performance analysis of different machine learning algorithms is needed to reach the best machine learning algorithm that provides the most optimal and precise prediction of stock market movement. The use of different machine learning methods has important practical implications in guiding investors in choosing algorithms, weighing risk-return, and making more rational investments (Sakhare and Imambi 2019; Vijh *et al.* 2020; Hu *et al.* 2022).

When the literature on machine learning applications in the forest-based industry was examined, it was found that the number of studies on stock market index value and stock price predictions in the forest-based industry is limited. In particular, there are studies on price and demand forecasting of forest products. Yıldırım *et al.* (2011) predicted financial return of paper sector in Turkey using artificial neural networks (ANN) and multiple linear regression (MLR). As a result of the study, it was observed that although both methods provide successful results, ANN performed significantly better than MLR. Yücesan *et al.* (2017) predicted the monthly sales of a furniture manufacturer located in the Black Sea region of Turkey with ANN model based on Bayesian rules training in the study.

It has been concluded that the ANN model is an applicable model in predicting the sales of the furniture factory. Verly Lopes *et al.* (2021) predicted Random Length lumber stock price using LSTM artificial recurrent neural networks. As a result of the study, it was observed that the LSTM network can efficiently capture non-linear temporal relationships. Kurniawan *et al.* (2021) aimed to predict the raw paper material of the printing company

with the short-term memory method, and it was determined that the method could be used in raw paper material prediction.

In the master's thesis by Ghorbanali (2022), the prices of products sold from a large furniture company were predicted using different machine learning methods like linear regression, Bayesian ridge regression, light GBM, and XGBoost. Results showed that XGBoost machine learning model achieved better prediction than other models. Yaneva and Kulina (2023) aimed to predict furniture demand with machine learning techniques. For this purpose, daily data of a large furniture manufacturer in Bulgaria was used. In the study predicting timber prices in Poland with artificial neural networks (RBF and MLP) and classical models (ARIMA, ETS, BATS, and TBATS) by Kozuch *et al.* (2023), they stated that neural networks provide more accurate results compared to classical methods that are widely used in predicting timber prices. Bardak (2023) tried to predict the prices of bookcase and dresser type furniture using data that obtained different public e-commerce sites in the United States and deep learning and random forest algorithms. It was concluded that deep learning and random forest algorithms are suitable for predicting furniture prices.

Studies on different stock market index prediction with machine learning methods can be listed as: prediction of Korea Composite Stock Price Index 200 (Pyo *et al.* 2017), prediction of Swedish OMX 30, British FTSE 100, and Australian S&P/ASX 200 indexes (Johnsson 2018), prediction of Japanese Nikkei 225 and Japanese Nikkei 400 indexes (Harahap *et al*. 2020), prediction of the BIST 30, BIST 50 and BIST 100 price indices (Yiğit et al. 2020), prediction of S&P 500 index (Abraham 2021), prediction of National Association of Securities Dealers Automate (NASDAQ), New York Stock Exchange (NYSE), Nikkei, and Financial Time Stock Exchange (FTSE) index (Subasi *et al.* 2021), prediction of BIST Transportation index (Özcan Akdağ *et al.* 2022), prediction of Indian Stock Market Nifty 50 index (Singh 2022), prediction of the Dow Jones stock index movement (Alihodzic *et al.* 2022), prediction of BIST 100 index (Ünvan and Ergenç 2023), prediction of Borsa Istanbul banks index (Armağan 2023), prediction of Taiwan 50 Exchange Traded Funds (ETF) index (Fan *et al.* 2024), prediction of MASI, CAC 40, DAX, FTSE 250, NASDAQ, and HKEX indexes, representing the Moroccan, French, German, British, US, and Hong Kong (Oukhouya *et al.* 2024).

In order to be useful to sector representatives, current and potential investors, and other individuals and organizations affected by changes taking place in the industry in the forest products, furniture, paper and publishing sectors, a prediction study was made on the XKAGT index values by creating artificial neural network, gradient boosting machine, random forest, and k-nearest neighbor models with monthly data based ten macroeconomic variables, and these techniques were compared. Moreover, there are limited prediction studies on stock market price of the forest-based sector using machine learning methods. In this respect, this study contributes to the literature.

EXPERIMENTAL

Data Collection and Processing

In this study, a dependent variable and 10 macroeconomic (independent) variables that were determined to have a direct or indirect effect on the stock were used. In determining the independent variables, studies in the literature (Budak *et al.* 2017; Gürsoy 2019; Durmuş *et al.* 2019; Fattah and Kocabıyık 2020; Özcan Akdağ *et al.* 2022) and expert opinions were used. The independent variables in the machine learning models were

crude oil price, exchange rate of USD/TRY, dollar index, BIST 100 index, gold price, money supply (M2), S&P 500 index, US 10-year bond interest, export-import coverage rate in the forest products sector, and deposit interest rate. Information regarding the variables used in the analysis is given in Table 1.

Table 1. Variables Used in Analysis

Data set	Variables	Access source	Variable type	Date range
Monthly	XKAGT index	Investing.com	Numerical	2002:01-2023:11
	BIST 100 index	Investing.com	Numerical	
	Gold price	Investing.com	Numerical	
	S&P 500 index	Investing.com	Numerical	
	US 10-year bond interest	Investing.com	Numerical	
	Crude oil price	Investing.com	Numerical	
	Exchange rate of USD/TRY	Investing.com	Numerical	
	Dollar index	Investing.com	Numerical	
	Deposit interest rate	Central Bank of the Republic of Türkiye (CBRT)	Numerical	
	Money supply (M2)	Central Bank of the Republic of Türkiye (CBRT)	Numerical	
	Export	Trademap.org	Numerical	
	Import	Trademap.org	Numerical	

Before machine learning models are established, the data set must go through some data preprocessing. Missing data analysis and outlier data analysis were performed from the data preprocessing. It was checked whether there was any missing data in the data set with the isnull parameter of the Python programming language and no missing data was found. Outlier analysis was performed using the boxplot method, and some results are given in Fig. 1. As seen in the figure, it was determined that there were no outlier values. After the preprocessing, the data was split into two: 80% (211) training data and 20% (52) test data.

Modeling

Experiments were conducted using Python coding on a data set consisting of 10 independent and 1 dependent variable. Artificial neural networks, random forest, k nearest neighbors, and gradient boosting machine algorithms were used, and four different models were established. Python programming language was used to apply machine learning techniques. The most common programming language used in 57% of artificial intelligence studies is Python, which has various libraries that can be used in different areas (Kızrak 2018).

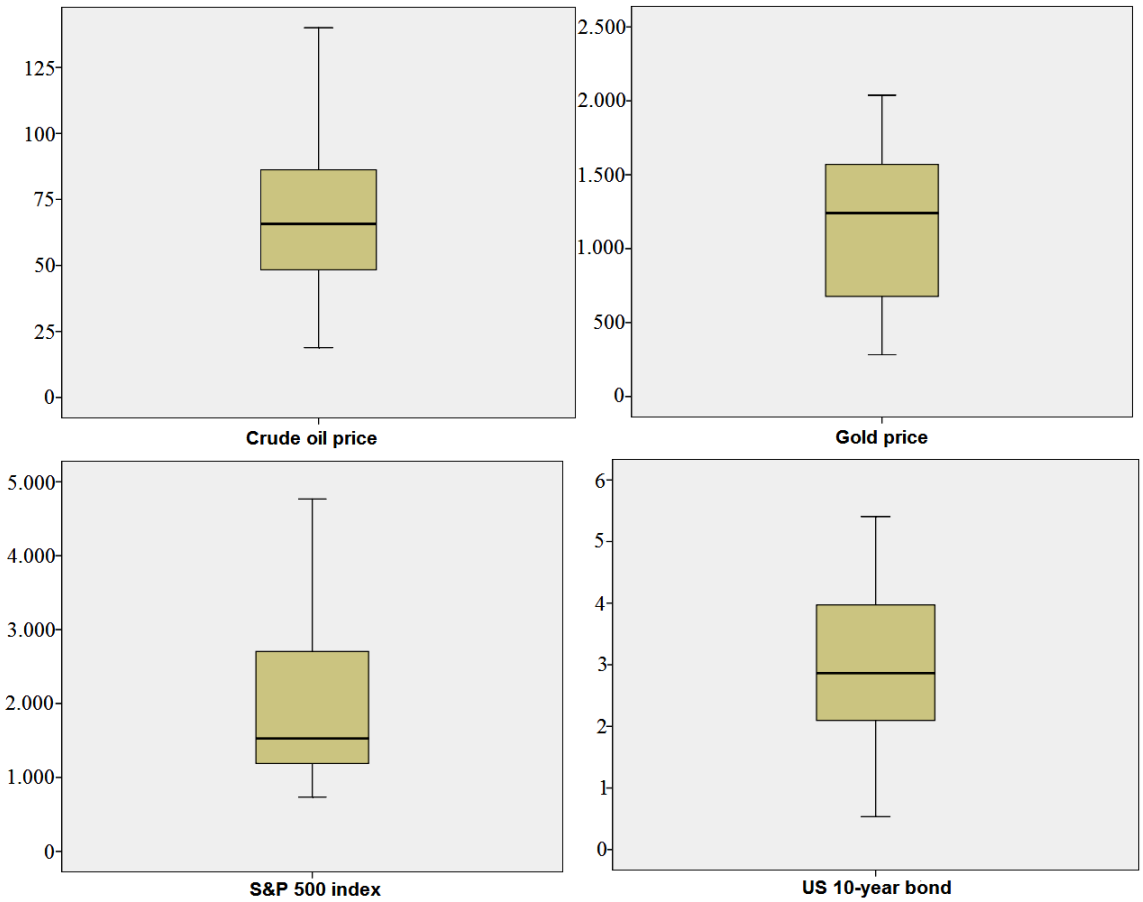


Fig. 1. Outlier analysis of variables

Artificial neural networks (ANN)

The prediction method with artificial neural networks was implemented using pandas, numpy, Pytorch, and Optuna libraries. The pandas and numpy libraries were utilized to manipulate and process datasets. The PyTorch library (Hassan *et al*. 2022) was employed in both the training and testing phases of the model. The use of the Optuna library (Akl *et al*. 2019; Yu and Zhu 2020; Abdolrasol *et al.* 2021; Rimal *et al.* 2024) was employed to optimize hyperparameters, including the learning rate and the quantity of neurons in the hidden layer of the model. In the beginning, the data set was partitioned into targets (y) and features (X) for the purpose of training the model. The data set was subsequently partitioned into test and training sets before being transformed into PyTorch tensors. A batch processing strategy was utilized throughout the model training phase to run over the dataset and modify the model weights. Upon completion of the training process, the model was assessed on the test set by calculating several performance measures (MSE, MAE, and R^2). In this model, both the ADAM optimization technique and the ReLU activation function were implemented (Ahmad *et al.* 2022). The ReLU function was selected for the activation of neurons in the hidden layer as part of the model's design. Weight adjustments during the model's training were performed using the ADAM optimization technique.

The performance of any machine learning model directly depends on the selected hyperparameter. There are different hyperparameter values for the algorithms used in modeling and separate calculations have been made for each algorithm. Table 2 presents

the usage values of hyperparameters used for ANN. As seen in Table 2, hidden size, number epoch and weight hyperparameters were analyzed for the ANN model. Hidden size represents the number of neurons in the hidden layer. This parameter was chosen within the range of 10 to 100. Number epoch specifies the number of forward and backward passes the complete training dataset undergoes through the ANN. The number epoch value was determined within the range of 50 to 200.Weight value was chosen within the range of 0.0001 to 0.2.

Table 2. ANN Hyperparameter Candidates

Hyperparameters	Search space
hidden size	10-100
number epochs	50-200
weights	0.0001-0.2

K nearest neighbor (KNN)

The k-nearest neighbor algorithm, defined by Fix and Hodges in 1951, is a non-parametric machine learning technique used for classification and regression based on linear supervised pattern recognition (Lu and Zhu 2014; Bhuvaneswari and Therese 2015). The application process of the nearest neighbor technique in this study is as follows: first, the pandas library was included into the model to facilitate data manipulation and analysis. Then, utilizing the K-Neighbors Regressor (Yao 2023) from the sklearn.neighbors module, value predictions have been generated according to the closeness of data points. The GridSearchCV framework (Kudari *et al.* 2021) was employed to ascertain and optimize the model's hyperparameters. In the K nearest neighbor algorithm model, learning rate, algorithm, and n neighbors hyperparameters were analyzed and the usage values of the hyperparameters were given in Table 3.

As it has a direct impact on the performance of the model, the selection of the number of nearest neighbors (k) is critical. "k" denotes as "n neighbors" (for regression) in Python coding. Demonstrating sensitivity to noise in the data, a reduced value of 'k' may result in overfitting. A greater value of 'k', on the other hand, might result in underfitting due to the smoothing of the decision boundary. This parameter was chosen within the range of 2 to 9.

The manner in which the distance or uniform weight parameter is utilized to determine the contribution of the neighbors to the output prediction process is how the neighbor contributions are weighted. The process by which each neighbor contributes equally to the forecast is referred to as "uniform weighting." Each neighbor is accorded equal importance in this setting. In contrast, neighbors are weighed according to their distance from the query point; this is referred to as "distance" weighting.

Algorithm is another crucial element. The algorithm employed for calculating the nearest neighbors is determined by this parameter. "Ball tree," "KD tree," and "Brute" are among viable choices. Through testing each of them, the code's auto-function ascertains which one is most appropriate.

Table 3. KNN Hyperparameter Candidates

Hyperparameters	Search space
n neighbors	2, 3, 4, 5, 7, 9
algorithm	Auto, Ball tree, Kd tree, Brute
learning rate	Uniform, Distance

Gradient boosting machine (GBM)

For the GBM model, inaddition to the sklearn.ensemble module's Gradient Boosting Regressor method (Rao *et al.* 2023), the pandas and numpy libraries have been utilized.

Achieving optimal tuning of hyperparameters is critical to improving the accuracy of predictions and the ability of the model to generalize. Improving hyperparameters delineates a domain containing probable hyperparameter values for the Gradient Boosting Model (GBM). This procedure involves the utilization of several hyperparameters, including the minimum sample leaf, learning rate, maximum depth, and n estimators (Table 4).

The number of trees in the forest is expressed with n estimators. In general, a greater number of "n estimators" indicates a model with greater complexity. This parameter was chosen as 100, 200, and 300. Learning rate is the term used to describe the speed at which modifications are implemented at each stage. In this study, learning rates such as 0.01, 0.1, and 0.2 have been tried. The parameter named max depth delineates the maximum depth that a tree inside the model can attain. This parameter was chosen as 3, 4, and 5. Another hyperparameter utilized by decision tree algorithms is "min samples split." It specifies the minimum number of samples that are necessary to further divide an internal node into child nodes. In this study, it was chosen as 2, 3, and 4. A hyperparameter utilized in ensemble methods such as random forests and gradient boosting machines, "min samples leaf" is a component of decision tree algorithms (GBMs). The numbers 1, 2, and 3 were utilized as min samples leaf. GBM is the moniker given to the GradientBoostingRegressor model that is constructed with random state set to 42.

Table 4. GBM Hyperparameter Candidates

Hyperparameters	Search space
n estimators	100, 200, 300
learning rate	0.01, 0.1, 0.2
max depth	3, 4, 5
min samples leaf	1, 2, 3
min samples split	2, 3, 4

Random forest (RF)

In this study, firstly, the data collection was loaded and processed with the pandas library. In the realm of broad mathematical and numerical operations, numpy has been favored. Utilization has been made of the Random Forest Regressor function (El Mrabet *et al.* 2022) from the sklearn.ensemble module. When optimizing hyperparameters for the Random Forest model, several factors must be considered (Contreras *et al.* 2021; Virro *et al.* 2022; Lee *et al.* 2023; Sandunil *et al.* 2023). As can be seen in Table 5, six hyperparameter analyzes were performed to tune the Random forests model. These

parameters are: n estimators, max features, min samples split, learning rate, max depth, and min sample leaf. With the exception of the max feature parameter, all other parameters are identical to those of the BGM model. The "max features" is a hyperparameter used in the Random Forest technique, which governs the quantity of features to be taken into account while constructing trees and identifying the optimal split. The choices are sqrt, log2, and none. The values of other hyperparameters were determined as n estimators: 100, 200, 300, learning rate: 0.01, 0.1, 0.2, max depth: 4 ,6, 8, 10, min sample leaf: 1, 2, 3, min sample split: 2, 5, 10.

Table 5. RF Hyperparameter Candidates

Hyperparameters	Search Space
n estimators	100,200,300
learning rate	0.01,0.1,0.2
max depth	4,6,8,10
max features	sqrt, Log2, None
min samples leaf	1,2,4
min samples split	2,5,10

Evaluation of Models

Coefficient of determination (R^2), mean absolute error (MAE), and root mean square error (RMSE) were used to measure the performance of all machine learning models. Table 6 includes the mathematical formulas and definitions of the performance metrics used in the research. K-cross validation technique was utilized to evaluate model performances more accurately and objectively. In this study, 3-fold cross validation was used on the training set.

Table 6. Performance Metrics Used in the Study (Botchkarev 2018; Correa-Jullian *et al.* 2020; Gao 2023)

Performance metrics	Mathematical formulas	Definitions
Coefficient of determination	$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$	It shows the extent to which the model can explain the deviation between the data.
Mean absolute error	$MAE = \frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}_i \rvert$	It represents the average of the absolute values of the errors.
Root mean square error	$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$	It is used to measure the magnitude of the variance between predicted and actual values.

RESULTS AND DISCUSSION

After preprocessing the data and splitting the dataset into training set and test set, the aim was to make parameter entries for the models and to obtain the best performance with these parameter entries. While selecting the best parameter values, it was taken as a basis that the total error value was minimum. The hyperparameters of the GBM, KNN,

and RF were optimized using GridSearchCV, whereas the hyperparameters of the ANN were optimized using ADAM and Optuna optimization technique. The best values of hyperparameter of all machine learning models used in the study are given in Table 7. When the best values of the parameters of all machine learning models are examined, it was determined that max depth=4, min samples leaf=1, min samples split=4, n estimators=300, and learning rate: 0.1 for GBM, algorithm=auto, weights=uniform, and n neighbors=2 for KNN, learning rate= 0.074, hidden size=58, and number epochs=194 for ANN, and max depth=10, max features= sqrt, min samples leaf=1, min samples split=2, n estimators=100, and learning rate: 0.1 for RF. Predictions were produced for both the training set and the test set using the best parameters.

Table 7. Optimal Model Parameters for All Machine Learning Algorithms

Models	Hyperparameter	Values
GBM	n estimators	300
	learning rate	0,1
	max depth	4
	min samples leaf	1
	min samples split	4
KNN	n neighbors	2
	algorithm	Auto
	learning rate	Uniform
ANN	hidden size	58
	number epochs	194
	weights	0,074
RF	n estimators	100
	learning rate	0,1
	max depth	10
	max features	sqrt
	min samples leaf	1
	min samples split	2

The performances of the prediction results were evaluated with RMSE, MAE and R^2 evaluation criteria. When the performances of the models in Table 8were examined, it was found that the MAE values of GBM is 0.620 for the training phase and 61.6 for the testing phase, the MAE values of RF was 25.25 for the training and 57.72 for the testing, the MAE values of KNN were 36.20 for training and 40.8 for testing, and the MAE values of ANN were 33.48 for the training and 63.31 for the testing. An important criterion used to evaluate the validity of the model was the correlation coefficient (R^2) between the experimental and prediction results. R^2 value takes a value between 0 and 1. If this value approaches 1, the model is quite compatible with the data (Özşahin 2012). When the R^2 values of the models were analyzed, the R^2 values in the training and test data sets of GBM were 0.999 and 0.978, respectively, the R^2 values in the training and test data sets of RF were 0.996 and 0.989, respectively, the R^2 values in the training and test data of KNN were 0.993 and 0.996, respectively. TheR^2 values in the training and test data sets of ANN were 0.997 and 0.991, respectively.

Table 8. Comparison of the Performance of the Models

Models	Training			Test		
	RMSE	MAE	R^2	RMSE	MAE	R^2
GBM	0.781	0.620	0.999	165.6	61.6	0.978
RF	68.00	25.25	0.996	114.4	57.72	0.989
KNN	88.46	36.20	0.993	71.36	40.8	0.996
ANN	56.24	33.48	0.997	104.9	63.31	0.991

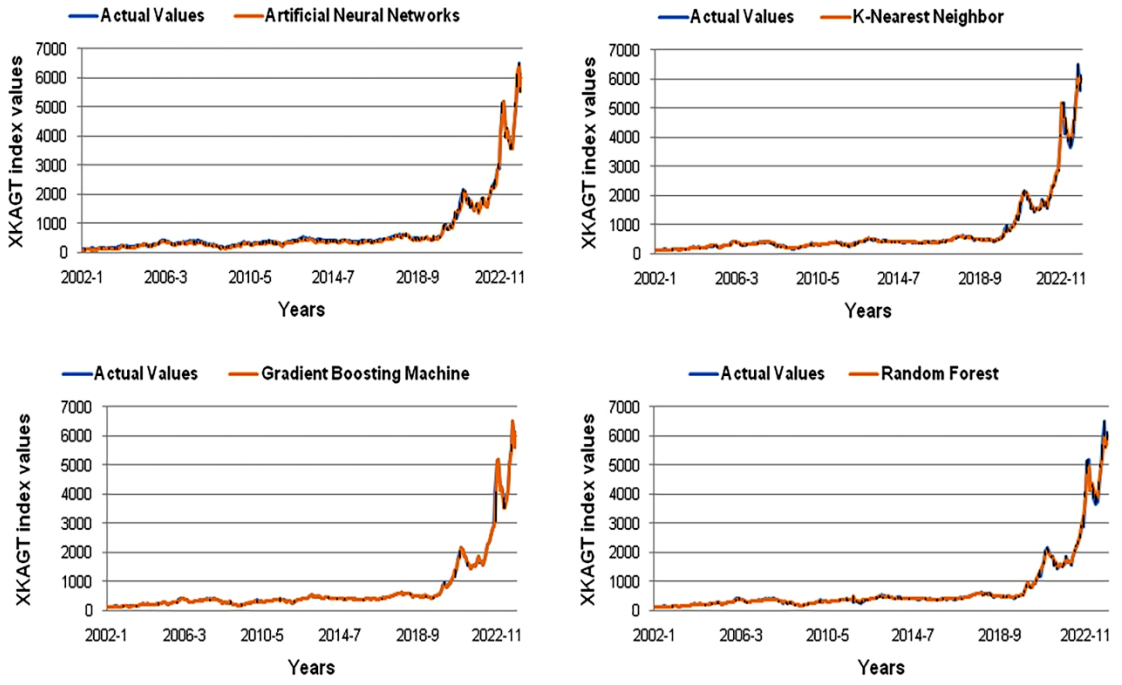


Fig. 2. Comparison of actual values and predicted values obtained with the all machine learning methods

When looking at Fig. 2, it can be seen that the predicted values obtained by GBM, RF, KNN, and ANN models were close to the actual values. Although all models used in index prediction gave successful results, it was observed that GBM's RMSE and MAE values were lower and its R^2 value was higher than other models in the training phase, and that KNN was the model with the best evaluation criteria in the testing phase. In the literature, there are studies on the successful performance of machine learning methods in stock price prediction. In other words, previous study results overlap with the results of this study. Yıldırım *et al.* (2011) found that The ANN and the multiple linear regression methods provide good performance for paper sector financial return prediction and ANN was significantly better than the multiple linear regression model. Roy *et al.* (2020) found that random forest and gradient boosting machine methods provide satisfactory performance in predicting stock prices. Verly Lopes *et al.* (2021) revealed that the LSTM artificial recurrent neural networks predicted the Random Length lumber stock price with low error terms for MSE, RMSE, and MAE. Singh (2021) found that machine learning techniques including ANN, LR, SGD, SVM, AdaBoost, RF, kNN and DT give significant results in Nifty 50 index prediction, but AdaBoost, kNN, RF and DT under performed with increase in the size of data set. Armağan (2023) determined that artificial intelligence-based

deep learning models perform better compared to traditional model in BIST banks index prediction and Convolutional Neural Networks Model (CNNM) gives the best results. Oukhouya *et al.* (2024) reported that machine learning models show very good results in predicting daily prices of stock indices.

CONCLUSIONS

1. The purpose of this study was to guide investors or those considering investing in analyzing and speculating on the stock price trend, which will help them in purchasing stocks to make maximum profits.
2. It is seen that models created with gradient boosting, random forest, k-nearest neighbors and artificial neural networks methods, which are machine learning methods, are successful in predicting the next day's closing values of the BIST Forest, Paper and Printing index. R^2 values of all models were above 0.97 in both training and testing phases.
3. Although all models performed well in index prediction, it was observed that GBM in the training phase and KNN in the testing phase were more successful than the others.
4. Industry actors, current and potential investors can have an idea about future index values by analyzing any of the machine learning techniques used in the study in line with the variables used in the study.
5. The use of only 10 macroeconomic variables and four machine learning methods as input in the study is a limitation of the study. Since stock market index values are affected by many factors in both short and long time periods, it would be meaningful for further research to include and analyze market psychology, and companies' financial ratios such as current ratio, cash ratio, leverage ratio as well as macroeconomic economic variables.
6. Research can be carried out to predict the stock price movements of individual companies included in the XKAGT index.
7. Different studies can be carried out by using different machine learning (*e.g.*, Decision tree, Support Vector Machine-SVM) and deep learning methods (*e.g.*, Convolution Neural Network-CNN, Simple Recurrent Network-SRN) and hybrid the methods (*e.g.*, LSTM-XGBoost).

REFERENCES CITED

Abdolrasol, M. G., Hussain, S. S., Ustun, T. S., Sarker, M. R., Hannan, M. A., Mohamed, R., Abd Ali, J., Mekhilef, S., and Milad, A. (2021). "Artificial neural networks based optimization techniques: A review," *Electronics* 10(21), 1-43. DOI: 10.3390/electronics10212689.

Abraham, A. (2021). "Forecasting stock market prices: A machine learning approach," All Graduate Plan B and other Reports, spring 1920 to spring 2023. 1610. (https://digitalcommons.usu.edu/gradreports/1610), accessed 20 May 2023.

Aegean Exporters' Associations. (2024). "Turkey's Furniture, Paper and Forest Products Export Data For 2022," (https://upload.eib.org.tr), accessed 16 February 2024.

Ahmad, G. N., Fatima, H., Ullah, S., and Saidi, A. S. (2022). "Efficient medical diagnosis of human heart diseases using machine learning techniques with and without GridSearchCV," *IEEE Access* 10, 80151-80173. DOI: 10.1109/ACCESS.2022.3165792

Akl, A., El-Henawy, I., Salah, A., and Li, K. (2019). "Optimizing deep neural networks hyperparameter positions and values," *Journal of Intelligent & Fuzzy Systems* 37(5), 6665-6681. DOI: 10.3233/JIFS-190033

Alihodžić, A., Zvorničanin, E., and Čunjalo, F. (2022). "A comparison of machine learning methods for forecasting Dow Jones stock index," in: *Large-Scale Scientific Computing*, I. Lirkov, S. Margenov (eds.), Springer, Cham. DOI: 10.1007/978-3-030-97549-4_24

Armağan, İ. Ü. (2023). "Price prediction of the Borsa Istanbul banks index with traditional methods and artificial neural networks," *Borsa Istanbul Review* 23(S1), S30-S39. DOI: 10.1016/j.bir.2023.10.005

Bardak, T. (2023). "Predicting prices of case furniture products using web mining techniques," *BioResources* 18(4), 7412-7427. DOI: 10.15376/biores.18.4.7412-7427

Bayramoğlu, M. F. (2007). *Using Artificial Neural Networks Models for Predicting Financial Indexes: An Application on Predicting of Daily Lowest and Highest Values of ISE National 100 Index,* Master's Thesis, Zonguldak Karaelmas University, Zonguldak, Türkiye.

Bhuvaneswari, P., and Therese, A. B. (2015). "Detection of cancer in lung with k-NN classification using genetic algorithm," *Procedia Materials Science* 10, 433-440. DOI: 10.1016/j. mspro.2015.06.077

Borsa Istanbul. (2023). "Indices, BIST Stock Indices," (https://www.borsaistanbul.com/en/index/1/stock-indices), accessed 30 May 2024

Botchkarev, A. (2018). "Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology," (https://arxiv.org/ftp/arxiv/papers/1809/1809.03006.pdf), Accessed 28 September 2023.

Budak, S., Ölmez Cangi, S., and Tuna, İ. (2017). "The effect of basic macroeconomic variables on BIST indexes," *The Journal of Academic Social Science* 55, 199-214. DOI: 10.16992/ASOS.12826

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., and Oliveira, A. L. I. (2016). "Computational intelligence and financial markets: A survey and future directions," *Expert Systems with Applications*, 55, 194-211. DOI: 10.1016/j.eswa.2016.02.006

Ceylan, T. (2018). *Approaches to the Application of Machine Learning in the Retail Sector*, PhD Thesis. Yıldız Technical University, İstanbul, Türkiye.

Contreras, P., Orellana-Alvear, J., Muñoz, P., Bendix, J., and Célleri, R. (2021). "Influence of random forest hyperparameterization on short-term runoff forecasting in an Andean mountain catchment," *Atmosphere* 12(2), article 238. DOI: 10.3390/atmos12020238

Correa-Jullian, C., Cardemil, J. M., Droguett, E. L., and Behzad, M. (2020). "Assessment of deep learning techniques for prognosis of solar thermal systems," *Renewable Energy* 145, 2178-2191. DOI: 10.1016/j.renene.2019.07.100

Durmuş, S., Yılmaz, T., and Şahin, D. (2019). "The effect of macroeconomic indicators on index revenues: BIST example," *EURASIA International Research Journal* 7(16), 870-886.

El Mrabet, Z., Sugunaraj, N., Ranganathan, P., and Abhyankar, S. (2022). "Random forest regressor-based approach for detecting fault location and duration in power systems," *Sensors* 22(2), 458. DOI: 10.3390/s22020458

Fan, M. H., Huang, J. L., and Chen, M. Y. (2024). "Stock and futures market prediction using deep learning approach," in: *Investment Strategies - New Advances and Challenges*, G. Prelipcean, M. Boscoinau (eds.), IntechOpen, London, United Kingdom. DOI: 10.5772/intechopen.114116

Fattah, A., and Kocabıyık, T. (2020). "The effect of macroeconomic variables on stock market indices: A comparison of Turkey and the USA," *Journal of Financial Research and Studies* 12(2), 116-151. DOI: 10.14784/marufacd.691108

Gao, S. (2023). "Trend-based k-nearest neighbor algorithm in stock price prediction," in: *Proceedings of the 3rd International Conference on Digital Economy and Computer Application (DECA 2023)*, Shanghai, China, pp. 746-756.

Ghorbanali, M. (2022). *Automation of Price Prediction Using Machine Learning in a Large Furniture Company*, Master's Thesis. Malmö University, Malmö, Sweden.

Gürsoy, A. (2019). "The effect of macroeconomic variables on stock returns: Case of banking sector," *Journal of Economics and Financial Research* 1(1-2), 1-25.

Harahap, L. A., Lipikorn, R., and Kitamoto, A. (2020). "Nikkei stock market price index prediction using machine learning," *Journal of Physics: Conference Series* 1566, 1-6. DOI: 10.1088/1742-6596/1566/1/012043

Hassan, W. H., Hussein, H. H., Alshammari, M. H., Jalal, H. K., and Rasheed, S. E. (2022). "Evaluation of gene expression programming and artificial neural networks in PyTorch for the prediction of local scour depth around a bridge pier," *Results in Engineering* 13, article 100353.

Hu, Q., Qin, S., and Zhang, S. (2022). "Comparison of stock price prediction based on different machine learning approaches," in: *Proceedings of the 2022 International Conference on Bigdata Blockchain and Economy Management (ICBBEM 2022*), Wuhan, Hubei province, China, pp. 215-231.

Johnsson, O. (2018). *Predicting Stock Index Volatility Using Artificial Neural Networks: An Empirical Study of the OMXS30, FTSE100 & S&P/ASX200*, Master's Thesis. Lund University, Lund, Sweden.

Kaderli, Y., Petek, A., Doğaner, M., and Babayiğit, G. (2013). "The sensitivity to market index and non-systematic risk measurement of sector indices in Borsa İstanbul," *Anadolu University Journal of Social Sciences* 13(3), 55-64.

Kahraman, Y. E. (2023). "The effect of profitability on business success in BIST forest, paper, printing index," *Third Sector Social Economic Review* 58(3), 2125-2143. DOI: 10.15659/3.sektor-sosyal-ekonomi.23.08.2161

Kızrak, A. (2018). "A getting started guide to artificial intelligence and deep learning," (https://ayyucekizrak.medium.com/yapay-zekaya-ba%C5%9Flama-rehberi-91e79d3de8e1), accessed 09 February 2024.

Kozuch, A., Cywicka, D., and Adamowicz, K. (2023). "A comparison of artificial neural network and time series models for timber price forecasting," *Forests* 14(2), article 177. DOI: 10.3390/f14020177

Kudari, J. M., Jebakumari, A., Kumar, S., Jebakumari, S. A., and Sushma, B. S. (2021). "Image classifier using the adam optimizer and the relu activation function,"

International Journal of Advanced Research in Engineering and Technology (IJARET) 12(3), 56-60. DOI: 10.34218/IJARET.12.3.2021.006

Kurniawan, F., Herwindiati, D. E., and Lauro, M. D. (2021). "Raw paper material stock forecasting with long short-term memory," in: *9th International Conference on Information and Communication Technology (ICoICT)*, Yogyakarta, Indonesia, pp. 342-347. DOI: 10.1109/ICoICT52021.2021.9527528

Kurt, F. E., and Senal, S. (2018). "Modeling volatility of insurance shares in Istanbul stock market: An application with arch-m models," *Journal of Süleyman Demirel University Institute of Social Sciences* 32, 314-332.

Le, H. L., Tran, D. H., and Van Chau, D. (2023). "A survey on the impact of hyperparameters on random forest performance using multiple accelerometer datasets,"*International Journal for Computers & Their Applications* 30(4), 351-361.

Lu, L., and Zhu, Z. (2014). "Prediction model for eating property of Indica rice," *Journal of Food Quality* 37, 274-280. DOI: 10.1111/jfq.12095

Oukhouya, H., Kadiri, H., El Himdi, K., and Guerbaz, R. (2024). "Forecasting international stock market trends: XGBoost, LSTM, LSTM-XGBoost, and back testing XGBoost models," *Statistics, Optimization & Information Computing* 12(1), 200-209. DOI: 10.19139/soic-2310-5070-1822

Özcan Akdağ, N., Karaatlı, M., and Kocabıyık, T. (2022). "Forecasting of BIST transportation index with ANN NARX model," *Alanya Academic Review Journal* 6(3), 2721-2746. DOI: 10.29023/alanyaakademik.1088404

Özşahin, Ş. (2012). "The use of an artificial neural network for modelling the moisture absorption and thickness swelling of oriented strand board," *BioResources* 7(1), 1053-1067.

Pyo, S., Lee, J., Cha, M., and Jang, H. (2017). "Predictability of machine learning techniques to forecast the trends of market index prices: Hypothesis testing for the Korean stock markets," *PLOS ONE, Public Library of Science* 12(11), 1-17. DOI: 10.1371/journal.pone.0188107

Rao, N. S. S. V. S., Thangaraj, S. J. J., and Kumari, V. S. (2023). "Flight ticket prediction using gradient boosting regressor compared with linear regression," in: *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, Chennai, India, pp. 1-6.

Ravikumar, S., and Saraf, P. (2020). "Prediction of stock prices using machine learning (regression, classification) algorithms," in: *Proceedings of 2020 International Conference for Emerging Technology (INCET)*, Belgaum, India, pp. 1-5.

Rimal, Y., Sharma, N., and Alsadoon, A. (2024). "The accuracy of machine learning models relies on hyperparameter tuning: student result classification using random forest, randomized search, grid search, bayesian, genetic, and optunaalgorithms," *Multimedia Tools and Applications*, 1-16. DOI: 10.1007/s11042-024-18426-2

Roy, S. S., Chopra, R., Lee, K. C., Spampinato, C., and Mohammadi-ivatlood, B. (2020). "Random forest, gradient boosted machines and deep neural network for stock price forecasting: A comparative analysis on South Korean companies," *International Journal of Ad Hoc and Ubiquitous Computing* 33(1), 62-71. DOI: 10.1504/IJAHUC.2020.104715

Sakhare, N. N., and Imambi, S. S. (2019). "Performance analysis of regression based machine learning techniques for prediction of stock market movement," *International Journal of Recent Technology and Engineering* 7(6S), 655-662.

Sandunil, K., Bennour, Z., Ben Mahmud, H., and Giwelli, A. (2023). "Effects of tuning hyperparameters in random forest regression on reservoir's porosity prediction. Case study: Volve oil field, North Sea," in: *57th US Rock Mechanics/Geomechanics Symposium*, Atlanta, USA. DOI: 10.56952/ARMA-2023-0012.

Singh, G. (2022). "Machine learning models in stock market prediction," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* 11(3), 18-28. DOI: 10.35940/ijitee.C9733.0111322

Subasi, A., Amir, F., Bagedo, K., Shams, A., and Sarirete, A. (2021). "Stock market prediction using machine learning," *Procedia Computer Science* 194, 173-179. DOI: 10.1016/j.procs.2021.10.071

Ünvan, Y. A., and Ergenç, C. (2023). "Stock market forecasting with machine learning: the case of BIST-100 index," *International Research Journal of Modernization in Engineering Technology and Science* 5(6), 1304-1312.DOI: 10.56726/IRJMETS41781

Verly Lopes, D.J., Bobadilha, G.d.S., and Peres Vieira Bedette, A. (2021). "Analysis of lumber prices time series using long short-term memory artificial neural networks," *Forests* 12, article 428. DOI: 10.3390/f12040428

Vijh, M., Chandola, D., Tikkiwal, V. A., and Kumar, A. (2020). "Stock closing price prediction using machine learning techniques," *Procedia Computer Science*, 167, 599-606. DOI: 10.1016/j.procs.2020.03.326

Virro, H., Kmoch, A., Vainu, M., and Uuemaa, E. (2022). "Random forest-based modeling of stream nutrients at national level in a data-scarce region," *Science of the Total Environment* 840, article 156613. DOI: 10.1016/j.scitotenv.2022.156613

Yao, B. (2023). "Walmart sales prediction based on decision tree, random forest, and k neighbors regressor," *Highlights in Business, Economics and Management* 5, 330-335.DOI: 10.54097/hbem.v5i.5100

Yavena, P. E., and Kulina, H. N. (2023). "Furniture market demand forecasting using machine learning approaches," *Journal of Physics: Conference Series* 2675, 1-12. DOI: 10.1088/1742-6596/2675/1/012008

Yıldırım, İ., Özşahin, Ş., and Akyüz, K. C. (2011). "Prediction of the financial return of the paper sector with artificial neural networks," *BioResources* 6(4), 4076-4091. DOI: 10.15376/biores.6.4.4076-4091

Yıldız, B., and Erdoğan, M. (2020). "Measuring financial performance of enterprises operating in forest, paper and printing sector: A TOPSIS application in Borsa Istanbul," *Journal of Business Research-Turk* 12(1), 938-954. DOI: 10.20491/isarder.2020.886

Yiğit, Ö. E., Alp, S., and Öz, E. (2020). "Prediction of BIST price indices: A comparative study between traditional and deep learning methods," *Sigma J Eng Nat Sci* 38(4), 1693-1704.

Yu, T., and Zhu, H. (2020). "Hyper-parameter optimization: A review of algorithms and applications," (https://arxiv.org/pdf/2003.05689.pdf), accessed 26 February 2024.

Yücesan, M., Gül, M., and Çelik, E. (2017). "Application of artificial neural networks using Bayesian training rule in sales forecasting for furniture industry," *Drvna Industrija* 68(3), 219-228.DOI: 10.5552/drind.2017.1706

Article submitted: March 8, 2024; Peer review completed: May 19, 2024; Revised version received and accepted: June 4, 2024; Published: June 13, 2024.
DOI: 10.15376/biores.19.3.5141-5157

APPENDIX

Appendix 1. The running code for GBM process

```
# Hyperparameter adjustment
param_grid = {
   'n_estimators': [100, 200, 300],
   'learning_rate': [0.01, 0.1, 0.2],
   'max_depth': [3, 4, 5],
   'min_samples_split': [2, 3, 4],
   'min_samples_leaf': [1, 2, 3]
}
# Model building by GridSearchCV
gbm = GradientBoostingRegressor(random_state=42)
grid_search = GridSearchCV(estimator=gbm, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)
grid_search.fit(X_train, y_train)
```

Appendix 2. The running code for KNN process

```
# Determining the hyperparameter range for KNN
param_grid_knn = {
  'n_neighbors': [2, 3, 4, 5, 7, 9],
  'weights': ['uniform', 'distance'],
  'algorithm': ['auto', 'ball_tree', 'kd_tree', 'brute']
}
# Hyperparametre optimization with GridSearchCV
knn = KNeighborsRegressor()
grid_search_knn = GridSearchCV(estimator=knn, param_grid=param_grid_knn, cv=3, n_jobs=-1, verbose=2)
grid_search_knn.fit(X_train, y_train)
```

Appendix 3. The running code for ANN process

```
# Objective function for hyperparameter optimization;
def objective(trial):
hidden_size = trial.suggest_int('hidden_size', 10, 100)
lr = trial.suggest_float('lr', 1e-4, 1e-1, log=True)
num_epochs = trial.suggest_int('num_epochs', 50, 200)
   model = ANNModel(X_train.shape[1], hidden_size, 1)
   optimizer = optim.Adam(model.parameters(), lr=lr)
   criterion = nn.MSELoss()
   for epoch in range(num_epochs):
model.train()
      for batch_x, batch_y in train_loader:
optimizer.zero_grad()
```

```
        outputs = model(batch_x)
        loss = criterion(outputs, batch_y)
loss.backward()
optimizer.step()
model.eval()
  with torch.no_grad():
y_pred = model(X_test_t)
mse = mean_squared_error(y_test, y_pred.numpy())
  return mse
# Optuna optimization
study = optuna.create_study(direction='minimize')
study.optimize(objective, n_trials=50)
```

Appendix 4. The running code for RF process

```
# Determining the hyperparameter range
param_grid = {
  'n_estimators': [100, 200, 300],
  'max_features': ['sqrt', 'log2', None],  # 'auto' yerine 'sqrt', 'log2' veya None
  'max_depth': [4, 6, 8, 10],
  'min_samples_split': [2, 5, 10],
  'min_samples_leaf': [1, 2, 4]
}
# Hyperparameter optimization with GridSearchCV
rf = RandomForestRegressor(random_state=42)
grid_search = GridSearchCV(estimator=rf, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)
grid_search.fit(X_train, y_train)
# Retrain the model using the best parameters
best_rf = grid_search.best_estimator_
best_rf.fit(X_train, y_train)
```