ESKİŞEHİR TEKNİK ÜNİVERSİTESİ BİLİM VE TEKNOLOJİ DERGİSİ
B- TEORİK BİLİMLER

Eskişehir Technical University Journal of Science and Technology B- Theoretical Sciences

Estuscience – Theory, 2026, 14 (1) pp. 1-19 , DOI: 10.20290/estubtdb.1700080

RESEARCH ARTICLE

FORECASTING DIFFERENT CURRENCIES USING MACHINE LEARNING METHODS

Hazim İŞCAN [1], Samet Kaan KANAK [2, *]

[1] Department of Computer Engineering, Faculty of Computer and Information Sciences, Konya Technical University, Konya, Türkiye.
hiscan@ktun.edu.tr - 0000-0002-3698-3745

[2] Department of Computer Engineering, Institute of Graduate Education, Konya Technical University, Konya, Türkiye.
samet.kanak@spk.gov.tr - 0000-0002-9415-9017

Abstract

There are a wide variety of investment instruments in the time we live in. Among these instruments, gold is the oldest investment instrument. The Dollar index is an index that shows the strength of the dollar against a major basket of the most popular currencies. The Nasdaq 100 Index is an index of the world's largest non-financial companies listed on the Nasdaq stock exchange. Bitcoin was the first cryptocurrency to emerge in 2009 and has been a popular investment vehicle in recent years. In this study, both the default and hyperparameter-optimized versions of five machine learning techniques — Linear Regression, Lasso, Decision Tree, Random Forest, and XGBoost — were employed to forecast the values of Gold, Bitcoin, the U.S. Dollar Index, and the Nasdaq 100 Index over two-, five-, and ten-year periods spanning 2015–2025. One currency is selected as the dependent variable, and three currencies are selected as independent variables. The dependent variable (currency) is estimated using the independent variables. The predictive capabilities of the models were evaluated using the RMSE, MSE, MAE, and R^2 performance metrics. According to the results, the Opt_Lasso model achieved the best forecasting performance among all models. A review of the existing literature reveals that most prior studies have primarily focused on predicting just a single dependent variable using various independent variables. In contrast, the present study distinguishes itself by performing simultaneous forecasts for four different currencies within a single application framework, thereby extending beyond previous research and offering a novel contribution to the field of multivariate prediction modeling.

Keywords

Currency forecast,
Decision tree,
Lasso,
Random forest,
XGBoost

Time Scale of Article

Received :16 May 2025
Accepted : 08 December 2025
Online date : 25 February 2026

1. INTRODUCTION

Before the invention of money, objects were exchanged. After money was invented, it began to be used as a substitute for the objects used in barter. In this way, trade became easier and faster, contributing to the development of commerce. Today, when it comes to money, not just a single type comes to mind. Gold, silver, dollars, euros, stocks, and Bitcoin, as well as other cryptocurrencies, that have recently become popular, are all used as forms of money.

Gold is the oldest and most valuable metal currency. It is also a widely used metal because it can be made into jewelry. In a study conducted by the Capital Markets Association of Turkey in 2011, gold was "the most preferred investment instrument” [1].

The dollar is the official currency of the United States of America and is the most widely used currency worldwide. The Dollar Index is an index measures the strength of the dollar against a major basket of the most popular currencies. It was developed in 1973 by the U.S. Federal Reserve to calculate a trade-weighted average value of the dollar against global currencies.

*Corresponding Author: samet.kanak@spk.gov.tr

Stock exchanges, on the other hand, are organized markets where all types of securities, shares, bonds, or commodities are traded by registering them on the stock exchange and operate under the supervision of the relevant state regulatory and audit institutions. The Nasdaq 100 Index is an index of the world's largest non-financial companies on the Nasdaq exchange in the U.S., predominantly consisting of technology companies.

Cryptocurrencies are not physical like gold or the dollar. They are also not subject to centralized control like stock exchanges. Due to these features, cryptocurrencies have introduced a novel perspective to financial markets and have gained significant popularity in recent years. In a 2021 study, Turkey ranked first in Europe and fourth in the world in cryptocurrency usage [2]. According to a worldwide study conducted in October 2022 by the Global Web Index (GWI) in London, Turkey has the highest number of people investing in cryptocurrencies among internet users aged 16-64 [3]. Bitcoin, the first cryptocurrency, emerged in 2009 and remains the most valuable of all cryptocurrencies.

Human beings have always wanted to make use of the money they have. They make these decisions using different currencies. The widespread use of computer technology today can help answer the question of how to invest money. Machine learning is the process of creating insights and making predictions from data using various mathematical operations with the help of computers.

In this study, a dataset was created using the values of the Dollar Index, Nasdaq 100 Index, gold and Bitcoin prices taken from investing.com between 2015 and 2025 to assist investors and make future forecasts. Using this dataset, both the default and hyperparameter-optimized versions of five machine learning methods — Linear Regression, Lasso, Decision Tree, Random Forest, and XGBoost — were applied for predictive modeling. In this way, it is possible to predict the prices of these three currencies for the coming years by utilizing their past prices.

Parisi et al. used artificial neural networks to determine the direction of change in gold prices. The dataset consisted of gold prices and the Dow Jones Industrial Average. Using this dataset, they compared artificial neural networks with ARIMA (Autoregressive Integrated Moving Average), a classic time series forecasting method. The findings indicated that ANNs outperformed ARIMA models, achieving a higher accuracy rate of 60% [4].

Aygören et al. forecasted the IMKB 100 index using the independent variables of gold price, interest rate, volume of bilateral transactions between banks, and the dollar exchange rate between 1995 and 2010. ARIMA, Newton and artificial neural networks were used as methods. It was found that forecasting with artificial neural networks was more successful than the others [5].

Sureshkumar and Elango used an artificial neural network multilayer perceptron (MLP) model to forecast a stock on the National Stock Exchange. To make this prediction, data from Tata Consultancy Services (TCS) between 2009 and 2011 were used as the dataset. The MLP model achieved a 93% prediction accuracy [6].

Kocatepe used data from thirteen variables (Turkish inflation, U.S. inflation, BIST100 index, crude oil price, Dollar Index, Turkish lira/dollar exchange rate, Standard & Poor's 500 index, Turkish interest and bond rates, U.S. interest and bond rates, silver and copper prices) between 2007 and 2015 to forecast gram gold prices in Turkey. He used artificial neural networks for forecasting and obtained an accuracy rate of 75.24% [7].

Yalçın analyzed the relationship between the BIST-30 index, dollar, euro, and gold prices in 2010-2014. He used multiple linear regression and logistic regression methods. In the study, where the dependent variable was the BIST30 index, multiple linear regression achieved an accuracy score of 50%, while logistic regression achieved an accuracy score of 80% [8].

Karakoyun forecasted Apple stock and Bitcoin prices in the U.S. Nasdaq stock market using ARIMA and Long Short-Term Memory (LSTM) methods. The study was conducted separately for the stock and Bitcoin. Data for the stock between 1984 and 2018 and the data of Bitcoin between 2013 and 2017 were used. The results showed that the LSTM method outperformed other approaches in both markets, achieving an accuracy rate of 94.22% for Apple and 98.6% for Bitcoin [9].

Öndes and Oğuzlar conducted a forecasting study using the Levenberg-Marquardt algorithm of artificial neural networks, based on data of gold, oil, and silver prices, USD/EURO parity, Euronext100 and the Dow Jones Industrial index between 2005 and 2017. As a result of the study, they achieved an accuracy rate of 81.43% [10].

Ustalı predicted 22 stocks traded in BIST-30 using artificial neural networks, random forest, and XGBoost algorithms. They created the dataset with values from 2010 to 2019. XGBoost achieved a more successful result than artificial neural networks and the random forest algorithm, with a 75% accuracy rate [11].

Şimşek conducted a comparative analysis of SVR, KNN, Random Forest, XGBoost, and Stacked Generalization models using BIST100 index values from January 4, 2010, to November 29, 2023. According to error metrics and accuracy rates, the Stacked Generalization method demonstrated the best performance, yielding the lowest error values and the highest accuracy across all evaluation criteria. XGBoost produced better results compared to Random Forest, while KNN and SVR showed relatively lower performance [12].

Hansun et al. examined Bitcoin, Ethereum, Cardano, Tether, and Binance Coin cryptocurrencies obtained from the Binance exchange between September 2014 and October 2021. Using this data, prediction studies were conducted with LSTM, Bi-LSTM, and GRU artificial neural network models. The performance metrics of each model, such as accuracy and processing time, were evaluated. The Bi-LSTM and GRU models yielded similar results in terms of accuracy. The GRU model was faster than LSTM and showed lower variance. They concluded that the Bi-LSTM and GRU models were effective for cryptocurrency price prediction, and that GRU's faster and more consistent performance would be particularly advantageous for real-time applications [13].

Ata and Erbudak conducted a study on exchange rate prediction using numerical data obtained from the Central Bank of the Republic of Turkey between January 2020 and May 2021. The dataset consisted of 1,352 exchange rate records, and four different machine learning methods were employed for prediction purposes. These methods included decision trees, linear regression, support vector machines, and Gaussian regression. The classification accuracy results obtained from the study were as follows: decision trees – 99.84%, linear regression – 99.18%, support vector machines – 93.72%, and Gaussian regression – 86.83%. These results indicate that the decision tree method outperformed the others in terms of prediction accuracy [14].

Ullah et al. conducted a study to forecast the future value of the Chinese Yuan (CNY) against the U.S. Dollar (USD). The study utilized a three-year dataset covering the period from April 25, 2020, to May 26, 2023. Two models, Long Short-Term Memory (LSTM) and Extreme Gradient Boosting (XGBoost), were employed for the prediction task. According to the results, the LSTM model outperformed the XGBoost model in terms of performance metrics such as RMSE, MAE, MSE, and MAPE. These findings suggest that the LSTM model is more suitable than XGBoost for modeling financial time series data [15].

Meng et al. utilized transformer-based architectures and deep learning models to improve the prediction accuracy of the RMB/USD exchange rate. The dataset was obtained from the WIND Financial Terminal. During the model training process, z-score normalization was applied. For forecasting, models such as TSMixer, FEDformer, LSTM, PatchTST, TimesNet, Transformer, MLP, TCN, and iTransformer were used. The results showed that transformer-based models achieved lower MAE and MSE values compared to traditional deep learning models such as MLP, TCN, and LSTM. This indicates that transformer-based models perform better than conventional deep learning approaches in this context [16].

Panda et al. developed a model (CNN-RF) that combines the Random Forest (RF) approach with Convolutional Neural Networks (CNN) for exchange rate prediction. The CNN-RF model integrates the strengths of both methods to produce more consistent results. The performance of the CNN-RF model was evaluated using a range of metrics, including R^2, MAPE, CEV, MSE, RMSE, and MAE. The proposed model was compared with Multi-Layer Perceptron (MLP), Autoregressive Integrated Moving Average (ARIMA), and Recurrent Neural Network (RNN) models. According to the study's results, the CNN-RF model outperformed the other three models in terms of prediction accuracy for exchange rates [17].

Lakhak compared the prediction performance of the ARIMA model with recurrent neural networks (RNN) and long short-term memory (LSTM) machine learning algorithms for predicting USD/EUR exchange rate data. The models were evaluated and compared based on the shape of the curves, Mean Absolute Error (MAE), and Mean Squared Error (MSE). According to the results of the study, the LSTM model outperformed both the ARIMA and RNN models in terms of performance, especially when considering the minimum MSE [18].

Olsen et al. compared the ability of the LSTM model to forecast EUR/USD implied volatility with that of Random Forest (RF), AR-GARCH, HAR, and MIDAS models. The predictive performance of the models was evaluated using metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). According to the study's findings, the AR-GARCH model outperformed the LSTM model for long-term forecasts, while the LSTM model delivered better performance than the other models for short-term forecasts. Among all the models, the Random Forest produced the least accurate predictions [19].

Tang and Xie proposed an innovative intelligent forecasting framework that combines optimal signal decomposition with multiple deep learning models. In their study, the Grey Wolf Optimization (GWO) algorithm was used to optimize noise parameters during the decomposition process, while Bi-LSTM, GRU, and FNN were employed to model the decomposed sub-series. The Zebra Optimization Algorithm (ZOA) was applied to generate the final forecasts. The dataset consisted of daily closing prices of EUR/USD, GBP/USD, and USD/JPY currency pairs from January 2013 to September 25, 2024. The proposed method was compared against ten alternative approaches, including traditional single predictors and advanced models combined with data decomposition. According to the comparison results, the proposed framework outperforms existing methods in terms of both forecasting accuracy and robustness [20].

Rahat et al. conducted a study using USD/BDT exchange rate data from 2018 to 2023, obtained from Yahoo Finance, applying Long Short-Term Memory (LSTM) and Gradient Boosting Classifier (GBC) methods. The performance of LSTM, GBC, and ARIMA was evaluated using several metrics, including directional accuracy, MAE, and RMSE. The LSTM model outperformed both GBC and ARIMA, achieving an RMSE of 0.986, MAE of 0.752, and a directional accuracy of 99.449%. GBC also achieved better values than ARIMA (1.342 RMSE, 1.015 MAE, and 52.30% directional accuracy), with an RMSE of 1.128, MAE of 0.894, and directional accuracy of 40.82% [21].

Liu et al. conducted a study to forecast daily Bitcoin closing prices based on Bitcoin price data from 2013 to 2018. To achieve this, they employed Linear Regression methods (OLS and LASSO), Long Short-Term Memory (LSTM), and Decision Tree algorithms. The collected dataset included approximately 3,000 trading days. The Linear Regression (LR) model was used to capture the temporal linearity in the Bitcoin time series, while LASSO was applied to prevent overfitting. Decision Trees were utilized for financial time series forecasting, with hyperparameter tuning performed using the GridSearch method. According to the experimental results, the LASSO model demonstrated the best performance in terms of MAE, while the Linear Regression model achieved the highest confidence score of 0.9998 [22].

2. MATERIALS AND METHODS

2.1. Data Set

The dataset used in this study was obtained from the Investing.com platform (https://www.investing.com). Investing.com is a reputable financial data provider widely used by investors and analysts internationally, offering reliable and up-to-date information. This platform sources its data directly from relevant stock exchanges and financial market providers and makes it available to investors and researchers. Therefore, data reliability and integrity comply with accepted standards in the international finance literature. Due to these features, Investing.com was considered an appropriate and reliable data source for this study.

The dataset includes data from January 1, 2015, to January 1, 2025, covering Bitcoin (USD), gold (USD per ounce), the Dollar Index, and the Nasdaq 100 Index. These four variables are assets that play a significant role in global financial markets and can influence investor behavior.

- Bitcoin: As the leader of the cryptocurrency market, it represents digital assets.
- Gold: A traditional "safe-haven" asset to which investors tend to flock to during times of economic uncertainty.
- Dollar Index (DXY): Reflects the strength of the US dollar relative to other major currencies, offering insights into global economic trends.
- Nasdaq 100: A stock index comprising major technology-heavy US companies, crucial for understanding financial market movements and risk appetite.

The selected date range is from January 1, 2015, to January 1, 2025 (a 10-year period). This period encompasses both the early stages of Bitcoin's entry into financial markets and recent developments, enabling long-term analysis. This period encompasses many significant developments, including economic fluctuations, global crises (e.g., the COVID-19 pandemic), monetary easing policies, and the growing interest in digital assets.

Since the data was obtained from a public source, Investing.com, there are no copyright or special permission requirements. The data has been used solely for academic and analytical purposes and has not been shared with third parties.

2.2. Machine Learning

Machine learning first appeared in 1959 in Arthur Lee Samuel's paper, “Some Studies on Machine Learning Using the Game of Checkers”. He chose this game because good and bad moves can be identified. In this way, it is possible to distinguish between good and bad moves [23]. Samuel's program became a much better player after playing against different people in learning mode. The developed program learned which moves were more successful and adapted its programming to include these strategies. In 1965, the first book on this subject was written by Nilsson [24].

Machine learning consists of two stages. The first stage is the learning phase, during which the model learns from the available data. The second stage is the inference phase, involves making inferences based on the learned information.

Today, machine learning is used in many different fields. For example, in medicine, it is used to detect diseases more easily; in real estate, to predict prices based on the characteristics of the house; and in social media, to recommend advertisements tailored to users' needs. In the sales and retail sector, machine learning is also used to investigate customer churn and its causes. This helps retain customers and reduce losses. In finance, machine learning is used to predict the future by utilizing past data.

2.3. Multiple Linear Regression

Regression plays an important role in many application areas by providing prediction, classification rules and data analytical tools to understand the behavior of different variables [25]. The multiple linear regression model is used to analyze the relationship between data groups with multiple variables [26]. Regression can be applied in scientific research in fields such as ecology, psychology, sociology, economics, physics and similar areas. It can also be used to predict insurance claims, election results, crime rates, and damages caused by natural disasters [27].

The formula for multiple linear regression is shown in Equation 1. In this formula, the values of the independent variables are included in the summation. Each x in the equation represents a different independent variable, and y is the dependent variable. β_0 is intercept of the line on the y-axis. β_1 to β_i are the regression coefficients, indicating the change in the dependent variable y for a one-unit change in the corresponding independent variable x. e represents the error term.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_i x_i + e \tag{1}$$

2.4. Lasso Regression

Lasso (Least Absolute Shrinkage and Selection Operator) regression analysis was developed by Robert Tibshirani in 1996 [28]. In multiple regression, the independent variables are not removed from the model even if they have no effect on the dependent variable. In Lasso regression, however, ineffective independent variables are removed from the model. The removal process is carried out by setting the corresponding coefficient to zero. With this feature, Lasso aims to increase the accuracy score by performing both variable selection and regularization simultaneously [29]. The figure given below shows the slopes of the line according to the value of λ used in Lasso.

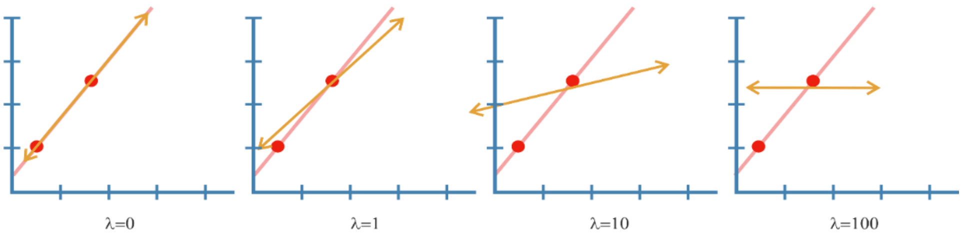

Figure. Lasso λ Values Change [19].

The Lasso estimation equation is shown in Equation 2. In Lasso regression, the L1 shown in Equation 3 is added to the linear regression equation. In this adjustment, the absolute values of the coefficients are multiplied by λ and then summed. λ is the regularization parameter. As λ increases, the slope of the line becomes steeper, and the β coefficients of insignificant variables decrease toward zero. Thus, the insignificant variables are removed from the model. This process is illustrated in Equation 4 [30].

$$\widehat{\beta}_{lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j \right\} \tag{2}$$

$$L_1 = \lambda * \left(|\beta_1| + |\beta_2| + \ldots + |\beta_p| \right) \tag{3}$$

$$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + 0 * x_3 + 0 * x_4 \tag{4}$$

2.5. Decision Tree

A decision tree resembles a tree structure with components such as leaves and branches. This tree contains both decision nodes and leaf nodes. Decision nodes are used for classification, prediction, or decision-making processes on the dataset. Each decision node can split into two or more branches. Leaf nodes contain the outcomes and represent the points where decision are made. The node at the top of the tree is called the root node. To obtain the results, a path is followed from the root node to the leaf nodes [31].

Instead of using linear regression methods, decision trees make predictions based on the average value of the samples that reach each leaf. Another type of tree used for prediction is model tree. In model trees, multiple linear regression models are created at each leaf based on the samples reaching that node. Depending on the number of leaf nodes, a model tree can generate many regression models. This feature can make model trees more complex and difficult to interpret. However, it can also lead to a more accurate model [27].

There are many different algorithms developed for decision trees., and the CART (Classification and Regression Trees) algorithm is one of them. CART is capable of working with both numerical and categorical data and is used to solve regression and classification problems. It employs decision trees to address these problems by splitting the dataset into subsets with two child nodes, thereby forming simple tree structures. These binary tree structures are created using Gini, binomial, ordered binomial, and least squares deviation criteria. The algorithm aims to divide the dataset into homogeneous subgroups [32].

In a dataset with N observations, where each observation consists of p input features and one response variable, the goal when performing a split is to minimize the Sum of Squared Errors (SSE). This relationship is expressed in the Equation 5 [33].

$$\min_{j,s}\left[\min_{C_L}\sum_{x_i\in R_L(j,s)}(y_i-C_L)^2+\min_{C_R}\sum_{x_i\in R_R(j,s)}(y_i-C_R)^2\right] \tag{5}$$

Here, x_i is the input feature or independent variable, and y_i is the target or dependent variable that is predicted based on the input feature xi. *R* represents a subgroup of the dataset after a certain split, with the subscript *R* and *L* indicating the "right" or "left" branches, respectively. C*R* and C*L* denote the average responses in the right and left subgroups after the split. The summation is performed over all observations in the respective subgroups, *R* (right) and *L* (left).

2.6. Random Forest

Random Forest is a powerful ensemble learning method developed by Leo Breiman in 2001, and it is widely used for both classification and regression tasks. It operates by constructing multiple decision trees during the training phase and producing an output that is either the mode of the predicted classes (for classification) or mean of the predictions (for regression) from the individual trees.

The working principle of the algorithm is as follows:

- Bootstrap Sampling: Random subsets of the training data are created with replacement (bootstrap samples). Each decision tree is trained on a different bootstrap sample.
- Random Feature Selection: At each split in a decision tree, a random subset of features is considered instead of all available features. This reduces correlation among trees and increases model diversity.
- Majority Vote / Averaging: All trained trees make independent on new data points. For classification, the class receiving the majority of votes is selected; for regression, the average of all tree predictions is taken as the final output.

Compared to a single decision tree, Random Forest offers the following advantages:

- Improved accuracy and robustness due to the averaging of multiple trees.
- Reduced risk of overfitting.
- Strong performance with large datasets and high-dimensional feature spaces.

Additionally, Random Forest can handle raw data with minimal preprocessing and can also be used to estimate feature importance [34][35].

2.7. XGBoost (Extreme Gradient Boosting)

XGBoost (Extreme Gradient Boosting) is a high-performance ensemble learning algorithm based on decision trees. It is particularly known for its combination of speed, accuracy, and regularization. The algorithm was developed and popularized in 2016 by Tianqi Chen and Carlos Guestrin. XGBoost represents an optimized and extended version of the Gradient Boosting algorithm.

Key Features of XGBoost:

- Gradient Boosting Mechanism: The model learns by sequentially adding a new weak learners (typically decision trees) at each step to minimize the errors made by previous learners.
- Regularization: Incorporate L1 (Lasso) and L2 (Ridge) regularization to control tree complexity, enhancing robustness against overfitting.
- Speed and Efficiency: Supports parallel processing, handles missing values effectively, and optimizes memory usage.
- Feature Importance: Provides the ability to identify which features have the greatest impact on the model's predictions.

Additional Strengths of XGBoost:

- Handling Missing Values: Automatically determines the optimal direction to take when encountering missing data during training.
- Weighted Data Support: Allows assigning different weights to individual instances, which is particularly useful for imbalanced datasets (e.g., fraud detection).
- Early Stopping: Halts training when no improvement is observed in a validation metric for a specified number of rounds, helping to prevent overfitting.
- Tree Pruning: Uses post-pruning rather than pre-pruning, applying constraints such as maximum depth or maximum number of leaves to avoid overly complex trees.

XGBoost is highly popular among practitioners seeking high accuracy in data science competitions such as Kaggle [36].

2.8. Proposed Method

This study was conducted using the Python programming language and the following libraries; pandas, NumPy, Matplotlib, Seaborn, and scikit-learn.

In this study, the currency to be predicted is considered the dependent variable, while the other financial assets (gold, the U.S. Dollar Index, Bitcoin, and the Nasdaq 100 Index) are used as independent variables. The developed application allows the user to select the target currency and then prompts the user to input the latest values of the financial indicators other than the selected target. Once this data is entered, the system automatically performs the corresponding prediction process [37].

Thanks to this structure, a user-centered and flexible prediction system has been established. It is generally observed in the literature that studies tend to focus on only a single fixed dependent variable, with separate models being developed for each dependent variable. In contrast, this study enables different currencies to alternately assume the role of the dependent variable within a single application framework; thus, a prediction infrastructure is created that can respond quickly to different scenarios without the need for retraining the model. In this respect, the study provides a significant innovation and contribution to the literature, both in terms of the modeling approach and user interaction.

In this study, five different modeling approaches were employed for forecasting financial time series: Linear Regression, Lasso Regression, Decision Tree, Random Forest, and Extreme Gradient Boosting (XGBoost). The rationale for choosing these models lies in their representation of different learning paradigms and their ability to jointly capture both linear and nonlinear relationships within time series data. The joint assessment of these five models enabled the capture of both linear trends and complex nonlinear dynamics within financial time series data.

In this study, the dataset was divided into three time horizons: two, five and ten years. Forecasting was performed separately for each subset.

Data Scaling:

In machine learning models, variables often exist on different scales—for instance, one variable may range from 0 to 1, while another may take values in the thousands. Such disparities can lead to imbalances in coefficient estimation and the optimization process. Distance-based algorithms and gradient-based optimization methods are particularly sensitive to these scale differences, which may cause the model to incorrectly assign greater importance to variables with larger numerical values.

To address this issue, both the input variables (xxx) and the target variable (yyy) were standardized to the same scale in this study, ensuring that the model treats all variables with equal sensitivity. Additionally, scaling facilitates faster convergence and improves the numerical stability of the model.

The Z-transformation (standardization) was employed as the scaling method. This process is performed by subtracting the mean of each variable from its individual observations and then dividing the result by the variable's standard deviation (Equation 6).

$$z = \frac{x - \mu}{\sigma} \tag{6}$$

In this equation, x represents the original observation, μ is the mean of the variable, and σ is its standard deviation. Through this standardization process, each variable was rescaled to have a mean of 0 and a standard deviation of 1, ensuring comparability among all variables.

Scaling was automatically performed using the StandardScaler() class. The mean (μ) and standard deviation (σ) were first computed from the training data, and these parameters were subsequently applied to transform the entire dataset. This standardization ensures that all variables are expressed on the same scale, allowing the model to fairly compare variables of different magnitudes. As a result, model parameter estimates become independent of variable scale, the optimization process is accelerated, and the stability of the loss functions is enhanced. Standardizing the data improves both the accuracy and generalizability of the model.

Data Preprocessing:

The datasets used in the analysis consist of Bitcoin, gold, U.S. Dollar Index, and Nasdaq data covering the periods 2023–2025, 2020-2025, and 2015–2025. Due to inconsistencies in trading days across different markets, weekends (Saturday and Sunday) and official holidays—during which other markets remain closed—were removed from the Bitcoin series. This adjustment ensured that all series were temporally aligned to include only common trading days.

After data cleaning, the Z-score method was applied to each variable to reduce the influence of outliers. Observations deviating by more than ±3 standard deviations from the mean were excluded to maintain the statistical integrity of the series.

To capture time series-specific characteristics, three lagged values (lags) and five-day rolling mean variables were generated for each variable. This step allowed the models to learn from both historical trends and short-term fluctuations.

Following the creation of these new variables, missing observations were removed, and only complete data periods were included in the analysis. Subsequently, all variables were standardized using the StandardScaler method to eliminate scale differences and stabilize the model training process.

As a result of these preprocessing steps, a statistically consistent, noise-reduced, and temporally synchronized time series dataset was obtained, covering identical date ranges for all financial assets.

Training and Test Split:

In time series analysis, conventional random data splitting methods are inappropriate, as they can disrupt temporal dependencies. In this study, a Time Series Cross-Validation approach was employed to partition the data into training and test subsets while preserving the chronological order.

The validation process was carried out using the TimeSeriesSplit method, with the dataset divided into five folds (n_splits=5). In each iteration, historical data served as the training set, while the subsequent 60 observations were used as the test set. This procedure allowed for an objective assessment of the model's predictive performance across different periods.

Model Optimization:

In this study, hyperparameter optimization was implemented to improve the overall performance of the forecasting models and to identify their optimal configurations. The optimization process for the Random Forest (RF), XGBoost (XGB), Decision Tree (DT), and Lasso (LASSO) models was conducted using the Optuna library, which employs a trial-and-error–based optimization algorithm.

The performance of machine learning models is determined not only by the choice of algorithm but also by the proper tuning of hyperparameters. Hyperparameters are configuration settings that directly affect the model's learning process and cannot be inferred automatically from the data.

For example:

- In the Random Forest model, the number of trees (n_estimators) and the maximum depth (max_depth),
- In the XGBoost model, the learning rate (learning_rate) and the subsampling ratio (subsample),
- In the Decision Tree model, the branching depth (max_depth), and
- In the Lasso model, the regularization coefficient (alpha / λ)

are hyperparameters that directly affect the model's generalization capability.

Selecting these hyperparameters through trial-and-error or grid search methods is both time-consuming and inefficient, particularly in high-dimensional parameter spaces. Therefore, the Optuna library, which provides a more systematic and adaptive search strategy, was chosen for hyperparameter tuning.

The Optuna library employs a structure similar to Bayesian optimization, learning from the results of previous trials and thereby focusing on the more promising regions of the search space in each new iteration. This approach enables the identification of optimal hyperparameters more quickly and with higher accuracy compared to traditional methods.

For each model, 15 trials (n_trials = 15) were conducted, and in each trial, model performance was evaluated using the coefficient of determination (R^2). Throughout the optimization process, a 3-fold time series cross-validation (TimeSeriesSplit) method, appropriate for the temporal structure of the data, was employed, and the mean R^2 value was maximized.

For the Decision Tree, the default parameter values were set to max_depth = 5 and random_state = 42. During optimization, the hyperparameters were varied within the following ranges: max_depth: 2–10, min_samples_split: 2–10, and min_samples_leaf: 1–5.

For Lasso, the default parameters were set to alpha = 0.01, max_iter = 10000, and random_state = 42. During optimization, the hyperparameter alpha (λ) was varied within the range 0.0001–1.0 on a logarithmic scale. In the Lasso model, the alpha (λ) value serves as the regularization coefficient, determining the degree of coefficient shrinkage. This parameter enhances the model’s resistance to overfitting while reducing the influence of irrelevant variables, thereby improving its generalization capability.

For Random Forest, the default parameters were set to n_estimators = 50, max_depth = 5, and random_state = 42. During optimization, the hyperparameters were varied within the following ranges: n_estimators: 50–100, max_depth: 3–6, min_samples_split: 2–5, min_samples_leaf: 1–3, and max_features: {'sqrt', 'log2'}.

For XGBoost, the default parameters were set to n_estimators = 50, max_depth = 5, learning_rate = 0.1, and random_state = 42. During optimization, the hyperparameters were varied within the following ranges: n_estimators: 50–100, max_depth: 3–6, learning_rate: 0.05–0.2, subsample: 0.8–1.0, colsample_bytree: 0.8–1.0, and min_child_weight: 1–3.

The performances of models trained with both default parameters and those optimized via Optuna were compared using the R^2, RMSE, MAE, and MSE metrics. The model achieving the highest R^2 score was selected as the final forecasting model. This process allowed for a systematic evaluation of model configurations and objectively demonstrated the impact of hyperparameter optimization on prediction accuracy.

3. RESULTS AND DISCUSSION

The performance results obtained from the two-year dataset are presented in Tables 1–4. Specifically, Table 1 reports the forecasting results for gold, Table 2 for the USD Index, Table 3 for Bitcoin, and Table 4 for the Nasdaq 100 Index.

Table 1. Performance results for Gold price forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9589	0,0635	0,0503	0,0041
Linear Regression	0,8377	0,0621	0,0477	0,0041
Default_Lasso	0,1885	0,1537	0,1305	0,0255
Opt_DecisionTree	-1,1748	0,5566	0,4642	0,425
Opt_XGBoost	-1,5877	0,6014	0,5135	0,4893
Default_RF	-1,8246	0,3552	0,2811	0,1619
Opt_RandomForest	-1,9636	0,6421	0,5551	0,5442
Default_XGB	-4,6905	0,4434	0,351	0,2248
Default_DecisionTree	-6,1816	0,4769	0,3931	0,3072

Table 2. Performance results for U.S. Dollar Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9426	0,1856	0,1465	0,0348
Default_Lasso	0,8595	0,2179	0,1676	0,0488
Linear Regression	0,8449	0,1866	0,1468	0,0353
Default_RF	0,7382	0,3188	0,2369	0,1183
Opt_XGBoost	0,7175	0,484	0,3677	0,2803
Default_XGB	0,6103	0,3689	0,277	0,1539
Opt_DecisionTree	0,6086	0,5476	0,4161	0,3396
Default_DecisionTree	0,4638	0,4563	0,33	0,2435
Opt_RandomForest	0,4056	0,7232	0,5899	0,5462

Table 3. Performance results for Bitcoin forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,972	0,0779	0,0573	0,0065
Linear Regression	0,914	0,0741	0,055	0,0059
Default_Lasso	-0,7735	0,2152	0,1877	0,0739
Opt_XGBoost	-1,1118	0,6675	0,502	0,5151
Opt_DecisionTree	-1,1388	0,6673	0,5012	0,5208
Opt_RandomForest	-1,1671	0,6733	0,5151	0,5218
Default_RF	-1,6238	0,5274	0,4334	0,3937
Default_DecisionTree	-1,8053	0,5449	0,4491	0,4093
Default_XGB	-1,8638	0,562	0,4661	0,4548

Table 4. Performance results for Nasdaq 100 Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9631	0,054	0,042	0,003
Linear Regression	0,9168	0,0531	0,0424	0,0029
Default_Lasso	0,8063	0,0819	0,0674	0,0068
Opt_DecisionTree	-1,6556	0,4253	0,3597	0,2185
Opt_RandomForest	-1,6595	0,4413	0,3645	0,214
Opt_XGBoost	-1,8764	0,4436	0,3557	0,218
Default_RF	-3,0385	0,3167	0,2693	0,1116
Default_DecisionTree	-3,1181	0,3385	0,2896	0,118
Default_XGB	-3,5751	0,3549	0,305	0,1431

Upon examining Tables 1–4, it is evident that the Opt_Lasso model demonstrates the highest performance by far across all datasets. This model produces the most accurate and consistent results for all types of forecasts, characterized by high R^2 values and low RMSE, MAE, and MSE metrics. In most cases, Linear Regression ranks second in performance. In contrast, the more complex models (XGBoost, Random Forest, and Decision Tree) generally exhibit poor or even negative R^2 values, indicating inferior performance. This finding suggests that the datasets possess strong linear relationships, making simple linear models more suitable for such data.

The performance results obtained from the five-year dataset are presented in Tables 5–8. Specifically, Table 5 reports the forecasting results for gold, Table 6 for the USD Index, Table 7 for Bitcoin, and Table 8 for the Nasdaq 100 Index.

Table 5. Performance results for Gold price forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9851	0,0512	0,0395	0,0027
Linear Regression	0,9125	0,0613	0,048	0,004
Default_Lasso	0,7605	0,1065	0,0871	0,0118
Opt_DecisionTree	0,316	0,5273	0,3982	0,6509
Opt_XGBoost	0,2567	0,5508	0,42	0,7064
Opt_RandomForest	0,0726	0,6092	0,473	0,8191
Default_DecisionTree	-3,8622	0,3996	0,3511	0,2519
Default_RF	-4,7178	0,4227	0,3678	0,2537
Default_XGB	-6,9939	0,5051	0,4379	0,3379

Table 6. Performance results for U.S. Dollar Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9846	0,0571	0,0435	0,0033
Linear Regression	0,9456	0,0473	0,0371	0,0023
Default_Lasso	0,8909	0,0669	0,0507	0,0046
Default_RF	0,8665	0,075	0,057	0,0059
Default_DecisionTree	0,78	0,0973	0,0768	0,0101
Default_XGB	0,678	0,0979	0,0785	0,01
Opt_DecisionTree	0,6113	0,2882	0,1836	0,1213
Opt_XGBoost	0,5094	0,3232	0,2112	0,1543
Opt_RandomForest	-0,0577	0,4978	0,3744	0,2882

Table 7. Performance results for Bitcoin forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9803	0,0633	0,0459	0,0045
Linear Regression	0,9074	0,0742	0,053	0,0059
Default_Lasso	0,8317	0,1008	0,071	0,0109
Opt_XGBoost	0,7457	0,2764	0,1708	0,1049
Opt_DecisionTree	0,6596	0,2941	0,2031	0,1042
Default_RF	0,3873	0,282	0,21	0,1713
Default_DecisionTree	0,3543	0,2966	0,2202	0,1679
Default_XGB	0,2694	0,3326	0,2454	0,2234
Opt_RandomForest	-0,0302	0,4901	0,3688	0,2847

Table 8. Performance results for Nasdaq 100 Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9841	0,0603	0,0455	0,0037
Linear Regression	0,9477	0,0501	0,0398	0,0027
Default_Lasso	0,8694	0,0806	0,0675	0,0066
Opt_XGBoost	0,0767	0,4752	0,3864	0,3678
Opt_DecisionTree	-0,0304	0,5022	0,4032	0,3877
Default_DecisionTree	-0,2905	0,2245	0,1872	0,0598
Opt_RandomForest	-0,3674	0,5928	0,495	0,5024
Default_RF	-0,4374	0,2405	0,1971	0,068
Default_XGB	-0,7921	0,2838	0,2349	0,0888

As shown in Tables 5–8, the Opt_Lasso model consistently achieved the highest accuracy and the lowest error metrics across all financial assets (Gold, U.S. Dollar Index, Bitcoin, and Nasdaq). The Linear Regression model followed closely, demonstrating stable and reliable performance in second place. Even without hyperparameter optimization, the Default_Lasso models performed at an acceptable level. Conversely, the tree-based models (Decision Tree, Random Forest, and XGBoost) exhibited low or negative R^2 values in most cases, indicating poor explanatory power for the target variable. These findings clearly suggest that the datasets are dominated by linear relationships, making linear models substantially more suitable for such forecasting tasks.

The performance results obtained from the ten-year dataset are presented in Tables 9–12. Specifically, Table 9 reports the forecasting results for gold, Table 10 for the USD Index, Table 11 for Bitcoin, and Table 12 for the Nasdaq 100 Index.

Table 9. Performance results for Gold price forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9907	0,0348	0,0256	0,0013
Linear Regression	0,9246	0,0362	0,0283	0,0014
Default_Lasso	0,8186	0,06	0,0478	0,0037
Default_DecisionTree	-0,4552	0,242	0,1969	0,1031
Default_RF	-0,6443	0,2489	0,1991	0,1014
Opt_DecisionTree	-0,6959	0,4696	0,3405	0,2859
Opt_XGBoost	-0,8454	0,485	0,3532	0,3004
Opt_RandomForest	-1,0499	0,539	0,3919	0,3625
Default_XGB	-1,6929	0,3084	0,2599	0,1554

Table 10. Performance results for U.S. Dollar Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9875	0,0606	0,0459	0,0038
Linear Regression	0,9119	0,0581	0,0455	0,0034
Default_Lasso	0,8443	0,0823	0,0637	0,0069
Default_RF	0,8144	0,0903	0,0695	0,0083
Default_DecisionTree	0,7216	0,1051	0,083	0,0111
Default_XGB	0,6409	0,1276	0,1029	0,0184
Opt_DecisionTree	0,4968	0,3312	0,2343	0,135
Opt_XGBoost	0,4592	0,3119	0,2202	0,1402
Opt_RandomForest	-0,6558	0,5367	0,4464	0,4273

Table 11. Performance results for Bitcoin forecasting.

Model	R^2	RMSE	MAE	MSE
Linear Regression	0,9209	0,0608	0,0461	0,0042
Default_Lasso	0,8299	0,0884	0,0646	0,0089
Default_RF	0,7288	0,116	0,0897	0,0159
Default_DecisionTree	0,6026	0,1375	0,107	0,0231
Default_XGB	0,509	0,167	0,142	0,0415
Opt_Lasso	-0,1566	0,1416	0,1112	0,0317
Opt_DecisionTree	-0,5662	0,5342	0,3878	0,4808
Opt_XGBoost	-0,5997	0,535	0,3838	0,4997
Opt_RandomForest	-0,6435	0,5728	0,4233	0,5308

Table 12. Performance results for Nasdaq 100 Index forecasting.

Model	R^2	RMSE	MAE	MSE
Opt_Lasso	0,9908	0,0356	0,0267	0,0014
Linear Regression	0,9152	0,029	0,0227	0,0009
Default_Lasso	0,7882	0,0476	0,0402	0,0023
Default_DecisionTree	-0,7567	0,1443	0,1199	0,0281
Default_RF	-1,1443	0,148	0,1243	0,0292
Default_XGB	-1,4768	0,1591	0,1365	0,0339
Opt_DecisionTree	-1,7133	0,5672	0,4691	0,4441
Opt_XGBoost	-1,8759	0,583	0,4809	0,4581
Opt_RandomForest	-1,967	0,6043	0,5032	0,49

As observed in Tables 9–12, the Opt_Lasso model consistently achieved the highest predictive accuracy and the lowest error metrics in the modeling tasks for Gold, the U.S. Dollar Index, and the Nasdaq 100 Index. The Linear Regression model followed closely, providing stable and reliable results, and even achieved the highest predictive accuracy and the lowest error metrics in the Bitcoin forecasting task. The Default_Lasso models also performed at a satisfactory level, further supporting the presence of a strong linear structure in the data. Conversely, the tree-based models (Decision Tree, Random Forest, and XGBoost) yielded low or even negative R^2 values in most cases, regardless of whether they were optimized or used with default settings. These findings indicate that the underlying data are largely linear in nature and that more complex models tend to introduce unnecessary complexity, leading to overfitting rather than performance improvement.

Table 13 presents a comparative analysis of the findings of this study with those reported in relatively similar research on financial time series forecasting, based on the R^2, RMSE, MAE, and MSE metrics. Due to the large size of the table, only the Opt_Lasso model — identified as the best-performing forecasting model in this study — has been included. The comparative analysis demonstrates that Opt_Lasso delivers strong and competitive performance across all metrics, achieving results that are comparable to or superior to those reported in previous studies.

Table 13. Comparison with relatively similar studies.

Authors	Dependent Variable	Independent Variable(s)	Method	R²	RMSE	MAE	MSE
Hansun et al. [13]	Bitcoin	Bitcoin	LSTM	-	2518,0217	1617,7592	-
			Bi-LSTM	-	2222,7354	1422,1933	-
			GRU	-	1777,3069	1167,3461	-
Şimşek [12]	BIST100	BIST100	Stacked	0,98024	0,00688	0,00385	0,00002
Ata and Erbudak [14]	Exchange rate	Exchange rate	Decision Tree	1	0,1522	0,00976	0,00023
			Linear Reg.	0,2	0,66752	0,5407	0,44558
			SVM	0,99	0,64769	0,6418	0,41951
			Gauss	0,75	0,33726	0,28072	0,1135
Ullah et al. [15]	CNY/USD	CNY/USD	LSTM	-	0,055	0,043	0,003
			XGboost	-	1,119	0,111	0,701
Meng et al. [16]	RMB/USD	StockIndex	TSMixer	-	-	0,032	0,002
		ExchangeRates	FEDformer	-	-	0,033	0,002
		CurrencyMarket	iTransformer	-	-	0,035	0,002
			PatchTST	-	-	0,039	0,003
			TimesNet	-	-	0,038	0,003
			Transformer	-	-	0,042	0,004
			MLP	-	-	0,053	0,006
			TCN	-	-	0,049	0,004
			LSTM	-	-	0,047	0,005
Panda et al. [17]	GBP/USD	GBP/USD	CNN-RF	0,9665	0,2329	0,122	0,2334
	AUD/USD	AUD/USD	CNN-RF	0,9665	0,3879	0,2275	0,3334
	EUR/CAD	EUR/CAD	CNN-RF	0,97	0,6079	0,3523	0,344
Lakhal [18]	USD/EUR	USD/EUR	LSTM	-	-	0,0179	0,00033
			RNN	-	-	0,0505	0,0029
			ARIMA	-	-	0,0164	0,00046
Olsen et al. [19]	EUR/USD	EUR/USD	LSTM	-	0,1995	0,0992	0,0398
			RF	-	0,201	0,1124	0,0404
			GARCH	-	0,1916	0,0972	0,0367
			HAR	-	0,195	0,1197	0,038
			MIDAS	-	0,1943	0,1012	0,0377
Tang and Xie [20]	EUR/USD	EUR/USD	OCEEMDAN	0,9551	-	2,6501	2,1076
	GBP/USD	GBP/USD	OCEEMDAN	0,9231	-	1,5493	1,6174
	USD/JPY	USD/JPY	OCEEMDAN	0,918	-	1,1659	1,0945
Rahat et al. [21]	USD/BDT	USD/BDT	LSTM	-	0,9858	0,752	-
			ARIMA	-	1,342	1,015	-
			GBC	-	1,128	0,894	-
Li et al. [22]	Bitcoin	Bitcoin	Naive Linear Reg.	-	-	0,962	-
			Lasso	-	-	0,918	-
			Decision Tree	-	-	1,3502	-
Our approach	Gold	Dollar Index Bitcoin Nasdaq 100	Opt_Lasso	0,9907	0,0348	0,0256	0,0013
	Dollar Index	Gold Bitcoin Nasdaq 100	Opt_Lasso	0,9875	0,0606	0,0459	0,0038
	Bitcoin	Gold Dollar Index Nasdaq 100	Linear Reg.	0,9209	0,0608	0,0461	0,0042
	Nasdaq 100	Gold Dollar Index Bitcoin	Opt_Lasso	0,9908	0,0356	0,0267	0,0014

4. CONCLUSION

In this study, the values of various financial assets — namely Gold, the U.S. Dollar Index, Bitcoin, and the Nasdaq 100 Index — were forecasted using both the default and hyperparameter-optimized versions of five different machine learning models: Linear Regression, Lasso Regression, Decision Tree, Random Forest, and XGBoost. The primary objective was to evaluate and compare the predictive performance of linear and nonlinear modeling approaches across different financial instruments.

To achieve this, datasets covering three distinct time horizons — two years, five years, and ten years — were constructed for each asset. These datasets were then used to generate forecasts and to assess how the length of historical data influences model accuracy and generalization capability. The findings reveal that the best predictive performance was obtained with the ten-year dataset. Models trained on the ten-year data achieved higher accuracy levels than those trained on the two- and five-year datasets. This improvement can be attributed to the longer time horizon capturing broader market dynamics and trends, reducing short-term noise, and providing the models with more comprehensive information for learning underlying patterns.

In the analysis encompassing all periods and asset types, the Opt_Lasso model clearly demonstrated the highest performance. Its R^2 values ranged between 0.98 and 0.99 in most tables, while RMSE and MAE values remained very low. These results indicate that the Opt_Lasso model provides the greatest stability and generalizability in terms of both accuracy and error metrics.

The Linear Regression model consistently ranked second, exhibiting reliable and stable predictive power, with R^2 values exceeding 0.90 in nearly all cases. The Default_Lasso models also achieved satisfactory performance without hyperparameter optimization, with R^2 values around 0.80 in most instances, suggesting the presence of strong linear relationships in the datasets.

In contrast, the tree-based methods (Decision Tree, Random Forest, and XGBoost) generally produced low or even negative R^2 values, even when optimized. This outcome indicates that the data do not exhibit complex nonlinear structures but are instead predominantly linear, driven largely by trends and correlations.

The forecast values in this study are for academic information purposes and do not constitute investment advice.

When examining the existing literature, studies on forecasting financial time series typically focus on a specific currency or financial indicator as a single dependent variable, with separate models developed for each asset. However, in this study, the prediction of different financial instruments such as gold, the dollar index, Bitcoin, and the Nasdaq 100 Index is performed within a single framework using a holistic approach. In this regard, the study differs from existing approaches in the literature not only by conducting multi-dependent variable forecasting but also by addressing assets with diverse characteristics within the same modeling system. This multidimensional forecasting approach represents a significant methodological and practical innovation, particularly given the increasing integration of financial markets. Thus, the originality of the study extends beyond the singular forecasting tendencies commonly observed in the literature.

CONFLICT OF INTEREST

The authors stated that there are no conflicts of interest regarding the publication of this article.

CRediT AUTHOR STATEMENT

Samet Kaan Kanak: Investigation, Methodology, Writing – Review & Editing. **Hazim İşcan**: Supervision, Visualization, Conceptualization.

REFERENCES

[1] TSPAKB. Capital Market Perception and Investment Potential Determination Research. Türkiye Sermaye Piyasaları Birliği. https://www.tspb.org.tr/wp-content/uploads/2023/05/ETM_baskanin_konusmalari_SP_algi_anketi_basin_sunumu_13042011.pdf. Publication at April 2011, Accessed at 25.10.2024.

[2] Buchholz K. These are the countries where cryptocurrency use is most common. World Economic Forum. https://www.weforum.org/stories/2021/02/how-common-is-cryptocurrency. Publication at 18.02.2021, Accessed at 10.10.2024.

[3] Kemp S. Digital 2022: Big Rise in Cryptocurrency Ownership, 2022. https://datareportal.com/reports/digital-2022-big-rise-in-cryptocurrency-ownership/ Accessed at 06.11.2024)

[4] Parisi A, Parisi F, Diaz D. Forecasting Gold Price Changes: Rolling and Recursive Neural Network Models. Journal of Multi. Fin. Manag., 2008; 18, 477-487.

[5] Aygören H, Sarıtaş H, Moralı T. Forecasting ISE 100 Indice Using Artificial Neural Networks And Newton Numerical Search Models. International Journal of Alanya Faculty of Business, 2012; 4(1), 73-88.

[6] Sureshkumar KK, Elango NM. Performance Analysis of Stock Price Prediction using Artificial Neural Network. Global Journal of Computer Science and Technology, 2012; 12 (1), 19-25.

[7] Kocatepe Cİ, Yıldız O. Forecasting Of The Direction Changes In The Gold Price In Turkey With Artificial Neural Network By Using Economic Indices. Duzce University Journal of Science and Technology 2016; 4(3), 926-934.

[8] Yalçın S. Dollar, Euro and Gold Price Changes Trends On The Bist 30 Effects In An Empirical Application. MS thesis, İstanbul Arel Üniversitesi, Sosyal Bilimler Enstitüsü, İstanbul, 2016; 171.

[9] Karakoyun EŞ. Time Series Prediction In Real Time Using Deep Learning. MS thesis, Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2018; 71.

[10] Öndes H, Oğuzlar A. Estimated Gold Price (TL / KG) With Artificial Neural Networks. Akademik Bakış Uluslararası Hakemli Sosyal Bilimler Dergisi, 2019; (72), 249-262.

[11] Koç Ustalı N, Tosun N, Tosun Ö. Stock Price Forecasting Using Machine Learning Techniques. Eskişehir Osmangazi University Journal of Economics and Administrative Sciences, 2021; 16(1), 1-16.

[12] Şimşek Aİ. Using Stacked Generalization Model in Stock Price Forecasting: A Comparative Analysis on BIST100 Index. Fiscaoeconomia, 2025; 9(1), 305-322.

[13] Hansun S, Wicaksana A, Khaliq A. Multivariate cryptocurrency prediction: comparative analysis of three recurrent neural networks approaches. Journal of Big Data, 2022; 9(1), 50-65.

[14] Ata O, Erbudak AE. Exchange Rate Forecasting Application with Data Mining and Machine Learning. Fırat Üniversitesi Mühendislik Bilimleri Dergisi. Eylül 2022; 34(2), 553-563. https://doi.org/10.35234/fumbd.1078207.

[15] Ullah U, Rehman D, Khan S, Rashid H, Ullah I. Forecasting Foreign Exchange Rate with Machine Learning Techniques Chinese Yuan to US Dollar Using XGboost and LSTM Model. IRASD Journal of Economics, 2024; 6(3), 761–775. https://doi.org/10.52131/joe.2024.0603.0238

[16] Meng S, Chen A, Wang C, Zheng M, Wu F, Chen X, Ni H, Li P. Enhancing Exchange Rate Forecasting with Explainable Deep Learning Models. 2024. https://doi.org/10.20944/preprints202410.0812.v1.

[17] Panda MM, Panda SN, Pattnaik PK. Convolutional Neural Network – Based Algorithm for Currency Exchange Rate Prediction. International Journal on Recent and Innovation Trends in Computing and Communication, 2023; 11(10), 551–559. https://doi.org/10.17762/ijritcc.v11i10.8525

[18] Lakhal S. Forecasting the EUR/USD Exchange Rate Using ARIMA and Machine Learning Models. Data and Metadata. 2024; 3. https://doi.org/10.56294/dm2024.368.

[19] Olsen A, Djupskås G, de Lange P, Risstad M. Forecasting implied volatilities of currency options with machine learning techniques and econometrics models. International Journal of Data Science and Analytics. 2024; 20, 1329-1347. https://doi.org/10.1007/s41060-024-00528-7.

[20] Tang X, Xie Y. Exchange Rate Forecasting: A Deep Learning Framework Combining Adaptive Signal Decomposition and Dynamic Weight Optimization. International Journal of Financial Studies. 2025; 13(3):151. https://doi.org/10.3390/ijfs13030151

[21] Rahat MdY, Gupta RD, Rahman NR, Pritom SR, Shakir SR, Showmick MdIH, Hossen MdJ. Advancing Exchange Rate Forecasting: Leveraging Machine Learning and AI for Enhanced Accuracy in Global Financial Markets. arXiv.org, 2025; papers 2506.09851.

[22] Liu S, Wu K, Jiang C, Huang B, Ma D. Financial Time-Series Forecasting: Towards Synergizing Performance And Interpretability Within a Hybrid Machine Learning Approach. arXiv.org, 2023; papers 2401.00534v1.

[23] Samuel AL. Some Studies in Machine Learning Using the Game of Checkers. IBM Journal of Research and Development, 1959; 3(3), 210-229.

[24] Nilsson NJ. Learning machines foundations of trainable pattern-classifying systems. McGraw-Hill, 1965: 137.

[25] Şenel T, Cengiz M, Savaş N, Terzi Y. Using Of Generalized Additive Model For Model Selection In Multiple Linear Regression. Erzincan University Journal of Science and Technology, 2014; 2(2), 217-227.

[26] Akış B. Map production methods for determinig the value of the case statistic Selçuklu. MS thesis, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2013: 95.

[27] Lantz B. Machine Learning with R, 4rd edn. Packt Publishing, England, 2023; 762.

[28] Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological), 1996; 58(1), 267-288.

[29] Fonti V, Belitser E. Feature Selection using LASSO. Research Paper in Business Analytics, 2017; 25.

[30] Muratlar ER. Alternative Regression Methods: Lasso Regression. Veribilimiokulu. https://www.veribilimiokulu.com/lasso-regresyon. Publication at 17.10.2021 Accessed at 03.10.2024.

[31] Kavzaoğlu T, Çölkesen İ. Classification of Satellite Images Using Decision Trees: Kocaeli Case. Harita Teknolojileri Elektronik Dergisi, 2010; 2(1), 36-45.

[32] Kurt I, Ture M, Kurum AT. Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease. Expert Systems with Applications, 2008; 34(1), 366-374.

[33] Wu T, Chen P, Chen C, Chung M, Ye Z, Li M. Classification and Regression Tree (CART)-based estimation of soil water content based on meteorological inputs and explorations of hydrodynamics behind. Agricultural Water Management, 2024; 299.

[34] Breiman L. Random Forests. Machine Learning, 2001; 45, 5–32 https://doi.org/10.1023/A:1010933404324

[35] Gao X, Wen J, Zhang C. An Improved Random Forest Algorithm for Predicting Employee Turnover. Mathematical Problems in Engineering. 2019; 1-12. https://doi.org/10.1155/2019/4140707

[36] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. 2016; 785-794. https://doi.org/10.1145/2939672.2939785.

[37] Kanak SK. Dolar Endeksi, Nasdaq Endeksi, Estimating The Values Of Dollar Index, Nasdaq Index, Gold and Bitcoin Depending On Each Other By Machine Learning Methods. MSc, Konya Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Konya, Turkey, 2023.