# Evaluating the Forecasting Ability of Deep Learning and Traditional Machine Learning Algorithms in USD-BDT Exchange Rate Forecasting

**By**

**Md. Arman Hosen**
191007512

Department of Computer Science & Engineering
School of Science, Engineering & Technology

Supervised by
Kazi Ekramul Hoque

Lecturer

In partial fulfillment of the requirement for the degree of *Bachelor of Science* in *Computer Science and Engineering*

May, 2023

**East Delta University**

**Noman Society, East Nasirabad, Chittagong - 4209**

# Evaluating the Forecasting Ability of Deep Learning and Traditional Machine Learning Algorithms in USD-BDT Exchange Rate Forecasting

This thesis is submitted in partial fulfillment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering.

**By**

**Md. Arman Hosen**
191007512

Supervised by
**Kazi Ekramul Hoque**

Lecturer

**Department of Computer Science & Engineering**

**School of Science, Engineering & Technology**

**East Delta University, Noman Society, East Nasirabad, Chittagong – 4209**

The undergraduate thesis titled **'Evaluating the Forecasting Ability of Deep Learning and Traditional Machine Learning Algorithms in USD-BDT Exchange Rate Forecasting'** submitted by Md. Arman Hosen (191007512) has been accepted as satisfactory in fulfillment of the requirement for the degree of Bachelor of Science (B.Sc.) in Computer Science and Engineering (CSE) to be awarded by East Delta University.

## Board of Examiners

_______________________

**Mohammed Nazim Uddin, Phd** **Chairman**
Associate Dean & Professor
School of Science, Engineering and Technology
East Delta University

_______________________

**Md. Ishtiaque Aziz Zahed, Phd** **Member**
Associate Professor
School of Science, Engineering and Technology
East Delta University

_______________________

**Linkon Chowdhury** **Member**
Assistant Professor
School of Science, Engineering and Technology
East Delta University

_______________________

**Kazi Ekramul Hoque** **Supervisor**
Lecturer
School of Science, Engineering and Technology
East Delta University

# DECLARATION

I, Md. Arman Hosen (191007512) declare that the work contained in this thesis is original. The information derived from the literature or work has been duly acknowledged and presented in the reference section. No part of this thesis has been submitted elsewhere for the degree, diploma or other similar title of recognition.

Date:

**Md. Arman Hosen**

**191007512**

# ACKNOWLEDGMENTS

I would like to express my deepest gratitude to the Almighty, the Creator of the Universes, for granting me the opportunity to undertake this great accomplishment, for my good health throughout the project, and for providing me with the strength and guidance needed to complete it successfully.

I am immensely grateful to my supervisor, Kazi Ekramul Hoque, for his unwavering support, guidance, and encouragement throughout my thesis project. His constructive criticism and insightful advice kept me motivated and on the right path to success.

I would also like to extend my heartfelt appreciation to my friends and family for their unwavering support and patience throughout the completion of this project. They have been a constant source of inspiration and motivation, and I could not have done this without them.

Lastly, I would like to acknowledge all the individuals who have directly or indirectly contributed to this project. Their help and support have been invaluable, and I am truly grateful for their contributions.

Thank you all for your support, guidance, and encouragement throughout this journey.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| | | |
|---|---|---|
| **FOREX** | : | Foreign Exchange Rate |
| **BRR** | : | Bayesian Ridge Regression |
| **KNN** | : | K-Nearest Neighbor |
| **RF** | : | Random Forest |
| **DT** | : | Decision Tree |
| **XGBoost** | : | Extreme Gradient Boost |
| **AdaBoost** | : | Adaptive Boost |
| **KRR** | : | Kernel Ridge Regression |
| **ARIMA** | : | Autoregressive Integrated Moving Average |
| **RNN** | : | Recurrent Neural Network |
| **LSTM** | : | Long Short-Term Memory |
| **MLP** | : | Multilayer Perceptron |
| **GRU** | : | Gated Recurrent Unit |

# ABSTRACT

The FOREX, or foreign exchange rate, is the relative value of a country's currency to another, and it plays a crucial role in international trade and investment. Accurate predictions of exchange rates are necessary for businesses involved in the global market or with overseas operations to handle currency risk, pricing, and decision of investment. This study examines the fluctuations in exchange rates between two economies by analyzing the forces that drive currency demand and supply on the world market. The aim of this study is to evaluate the forecasting ability of machine learning models in predicting short-term exchange rate fluctuations during sudden changes in the economy. The models were trained using past prices of the Bangladeshi Taka (BDT) with the US Dollar (USD) alone and with additional microeconomic features to create a multivariate dataset. The study compares the predictions made by the different models to determine the most reliable one. The results show that the Bayesian Ridge Regression model performs the best in both cases. The importance of reliable exchange rate predictions cannot be overstated in today's interconnected global economy, making this research valuable to practitioners, policymakers, and academics alike.

**Keywords:** Time-series, Exchange Rate, Regression, Forecasting, Lagged Dataset.

# CHAPTER 1

# INTRODUCTION

## 1.1 Background of Research

Bangladesh is considered to be a country with great potential. Though it is burdened with a huge population, many consider this population an asset. The positive side of this huge population is that it can contribute to economic growth. Speaking of the drawbacks, this population relies heavily on international trade because it may not be viable for Bangladesh to supply all of the everyday basics that people need. Bangladesh, as a result, suffers from an utterly irrational exchange rate. The foreign exchange rate (FOREX) in its simplest sense can be explained as the value of a country's currency relative to another, which is what the word "exchange rate" denotes. As the world economy has become so reliant on international trade, anyone purchasing products or services from a nation must do so in the form of approved local currency of that nation. For example, despite having US dollars (USD), a company that purchases items from Bangladesh must pay its bill in Bangladeshi taka (BDT). In this transaction, the exchange rate of the currencies is crucial. Depending on the exchange rates that is set by the central bank of Bangladesh, that organization can exchange US dollars (USD) for Bangladeshi taka (BDT).

It is well known that the condition of the importing country will worsen with an increase in the exchange rate, and vice versa. In this situation, there is no question that the exchange rate directly affects the economic development of any nation [1].

Predictions of exchange rates can be helpful and required for a range of people and organizations. Businesses that are involved in the global market or have overseas operations may need to estimate exchange rates to handle their currency risk as well as make smart decisions regarding pricing and investing. Exchange rate forecasts can be used by traders of international stocks, bonds, and currencies to support their decisions. In order to manage their foreign exchange reserves and set interest rates, central banks require exchange rate forecasts to supervise their monetary policy decisions. Forecasting currency rates may be necessary for governments to manage their foreign debt, taxation, and trade policies. Exchange rate forecasts might well be beneficial for budgeting and planning expenses for people who go overseas.

For these reasons, predicting exchange rates has always been a field of study for researchers, policymakers, and academics.

## 1.2 Problem Statement

The research that was done for this article has mostly emphasized on the influence of the international economy in the changes in exchange rates of two economies. This was accomplished by taking into consideration the forces that drive currency demand as well as supply on the world market. The demand and supply of Bangladeshi currency on the worldwide market is what determines the value of the Bangladeshi Taka (BDT), which uses a floating exchange rate system [2]. This means that the value of the Bangladeshi Taka is not fixed. In the context of macroeconomic policy, the term "floating exchange rate" refers to a type of exchange rate regime in which the value of a currency fluctuates in reaction to activity taking place on the foreign exchange market [3].

In the past, economists attempted to compare the performance of their statistical exchange-rate approaches utilizing the horse race method, in which they compared how accurately each model predicted the actual values of the currency rate. Machine learning algorithms have introduced a new dimension by enabling devices to self-learn. Algorithms capable of machine learning may perform complex computations more rapidly and make more precise decisions. As a result, machine learning algorithms have supplanted manual accounting methods for estimating currency exchange rates, as they have been demonstrated to be significantly more effective and precise. Various macroeconomic factors influence the volatility of exchange rates, and the purpose of this study is to forecast the USD/BDT exchange rate using machine learning and deep learning algorithms with consideration of macroeconomic theory. A rise in the money supply and the cost of debt repayment causes real currency depreciation, whereas an increase in foreign exchange reserves causes real currency appreciation. In addition, political unrest substantially devalues the national currency [4]. The USD/BDT exchange rate is determined by the dynamics of supply and demand on the foreign exchange market. Supply and demand on the FOREX market are affected by macroeconomic factors of the United States government. These macroeconomic factors of the United States have been taken into account when determining the factors that influence exchange rate fluctuations on the global FOREX market. In this study, several macroeconomic aspects of Bangladesh are considered once more, as they also influence how the Bangladeshi central bank determines the USD/BDT exchange rate. Since all of these variables influence the USD/BDT exchange rate, we combined the macroeconomic variables from the United States and Bangladesh. Because this study incorporates macroeconomic factors and a substantial quantity of economic data, it is possible to predict exchange rates over a longer period of time with greater precision.

## 1.3 Research Goal

The primary objective of the project is to construct a machine learning model for forecasting USD/BDT exchange rates. A more accurate exchange rate prediction model will reduce the risk that investors face on the Forex market. A significant increase in investment will result from a decrease in investor risk. Several Machine Learning algorithms, such as, DT (Decision Tree), KNN (K-Nearest Neighbor), RF (Random Forest), BRR (Bayesian Ridge Regression), KRR (Kernel Ridge Regression), KNN (K-Nearest Neighbor), and various Ensemble models which are XGBoost (Extreme Gradient Boosting), AdaBoost (Adaptive Boosting), Extra Tree and RNN model such as LSTM (Long Short-Term Memory). The outcomes of numerous machine learning models are subsequently evaluated to determine the most efficient model. Using univariate and multivariate time-series data, this study aims to evaluate the performance of various machine learning models in predicting short-term fluctuations in the market. This research establishes a benchmark for future efforts to forecast the USD-BDT exchange rate.

## 1.4 Chapter Outlines

This study is categorized into four different parts. They are:

- Chapter 2 covers some research topics that are related to our chosen research topic.
- Chapter 3 outlines the whole system architecture of proposed model for predicting the exchange rates of USD-BDT.
- Chapter 4 discusses the results and analysis of the different machine learning models.
- Lastly, Chapter 5 includes the conclusion and future scope of the research.

# CHAPTER 2

# LITERATURE REVIEW

I researched a ton of relevant works in the fields of machine learning, neural networks, and economics before I started putting my thesis work into operation. Some noteworthy world related to my work, forecasting of USD/BDT exchange rate are mentioned below:

By examining the effects of numerous macroeconomic variables on the exchange rate, Biswas et al [5] recent study sought to predict the USD/BDT exchange rate. Based on their correlation with the target variable, the BDT price, they looked at 16 macroeconomic variables. The authors applied different deep learning models, including time distributed MLP, to estimate the exchange rate. The best RMSE value for their planned pipeline was 0.1900 when combined with time distributed MLP, while time distributed MLP alone produced an RMSE value of 0.1984. The authors discovered that temporally distributed MLP outperformed alternative baseline models, including Bi-LSTM, LSTM, GRU, CNN, stacked-LSTM, ANN, CNN-LSTM, Encoder-Decoder, SVM, and XGBoost, regardless of the use of their proposed pipeline.

A statistical method was used by Jahanara et al. [6] to analyze the exchange rate of USD-BDT time series. They discovered that there has been an increased tendency in the exchange rate of USD-BDT over time. The authors used 12 distinct iterations of the ARIMA model with various parameters for the moving average (MA), integration, and autoregression (AR). The authors discovered that ARIMA (2,1,1) with drift produced the best outcome out of all feasible ARIMA models. They highlight the significance of statistical models in predicting exchange rate movements in their study, which offers insightful information about the dynamics of the USD-BDT exchange rate.

Md. Shahajada Mia et al. [7] conducted a study comparing the predictive accuracy of statistical models as well as machine learning models, including time-series and neural network models, in forecasting the exchange rate of USD-BDT. They found that the ARIMA (1,1,1) model when applied on the dataset resulted in an RMSE value of 0.325664, whereas the Double Exponential Smoothing model resulted in an RMSE of 0.6639. The neural network model with one input layer and two hidden layers (ANN (1,2)) achieved the best result, with an RMSE of 0.240281. The study's findings suggest that the neural network model fits the exchange rate data well and is capable of forecasting the future trend of exchange rate movements accurately.

Using the annual exchange rates of three nations and the related macroeconomic data, such as relative interest rates, Ramasamy R et al [8] conducted research to investigate the everchanging effects of macroeconomic variables on exchange rate movements. In order to increase the training size and run regressions to examine how these variables affect exchange rates, the authors used the bootstrapping technique. According to their research, macroeconomic factors including tax rates, balance of payments, interest rates, and inflation might have a random effect on the exchange rates. Nonetheless, these factors could be unstable depending on how a nation's economy is doing.

In a study by Refenes et al. [9], they showed that in datasets with nonlinearities, particularly stock indices, neural networks can beat statistical forecasting methods. In contrast to regression models, the researchers discovered that employing neural networks and sensitivity analysis might give users a better grasp of the predictive behavior of the model. According to the study, neural networks can offer a more reliable and precise method for forecasting stock indexes. Sensitivity analysis can be used to determine the main factors influencing stock index movements, which can then be added to the neural network model to increase its precision.

Computational generalization programming (CGP) and recurrent neural networks (RNN) were utilized by Rehman et al. [10] to forecast exchange rates between the Australian dollar (AUD) and three other currencies. To predict exchange rates, the researchers used a Recurrent Neuro-Evolution method. According to the results, the computational method performed better than previous statistical methods because it could choose the optimal feature in real-time, was flexible in its feature selection, and was good at spotting trends. According to the study, predicting currency exchange rates can be done more precisely and effectively by using computer techniques like CGP and RNN. Better forecasting accuracy can result from the flexibility in feature selection and the capacity to choose the optimum feature in real-time. The capability of these techniques to capture more intricate and subtle linkages in currency exchange rates is further highlighted by the success in identifying patterns.

A hybrid GRU-LSTM network was employed by Islam et al. [11] to forecast the various exchange rates of significant currency pairs. However, the hybrid model beat solo GRU and LSTM models in their performance comparison, the researchers observed. According to their findings, a hybrid model might be useful for forecasting the exchange rate of USD/BDT.

In their review study, Pandey et al. [12] assessed the effectiveness of statistical and neural network models in forecasting exchange rates and put forth a hybrid machine that overcomes the drawbacks of both methods. In terms of predictive accuracy, their investigation showed that multilayer neural networks utilizing a BAYESIAN learning strategy beat those using a backpropagation learning approach. This raises the possibility of utilizing hybrid models, which combine the benefits of statistical and neural network techniques for more precise exchange rate prediction.

In one work, Rout et al. [13] projected exchange rates forecasting using an adaptive ARIMA model with differential evolution-based training. This model outperformed rival models in terms of training time and accuracy for both long- and short-term predictions, the researchers found. Convolutional neural networks were utilized by Panda MM, Panda SN, and Pattnaik PK [14] in a different study to predict multi-currency exchange rates. In order to offer optimal weight for their suggested model, they proposed a model that could more effectively utilize multivariate exchange rate data. They did this by using the adaptive learning rate (ADAM) optimization technique.

# CHAPTER 3

# RESEARCH METHODOLOGY

## 3.1 System Architecture of Proposed Model

This section describes the approach and materials recommended for this study. To implement a forecasting method, a suitable dataset has been prepared for training and testing purposes. This involves preprocessing the bank rates of USD-BDT exchange rates, as well as other microeconomic factors, and extracting their respective features. The complete diagram of the system is presented in this part.

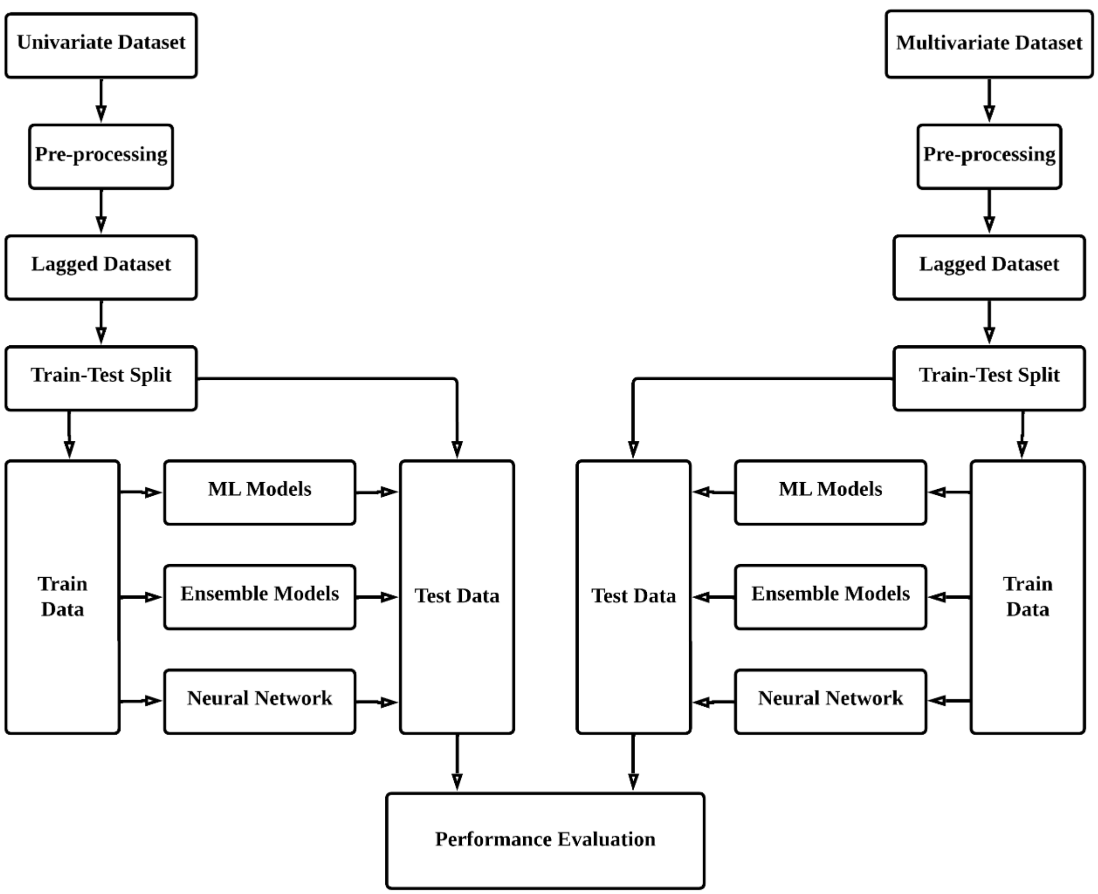


***Figure 3.1 Proposed Model Architecture***

## 3.2 Dataset

A website named Investing (https://www.investing.com/) provided the information for the exchange rate of USD-BDT for the period of January, 2019 to December, 2022 excluding the weekends and public holidays. The official websites of the Bangladesh Bank Bangladesh Bank (bb.org.bd) and the US government's official website (Census.gov) were used to collect additional macroeconomic data for Bangladesh and the US, respectively. The dataset was produced by the simultaneous collection of data on the exchange rate and macroeconomic factors. Using "Price" as the target parameter and the remaining 10 macroeconomic indicators used as features for training the models, the dataset consists of 1012 records of the exchange rate of USD-BDT. These macroeconomic factors are:

- Inflation Rate of Bangladesh
- GDP of USA
- Inflation Rate of USA
- Exports of USA
- Money Supply of USA
- Imports of USA
- USA trading goods with Bangladesh (Imports)
- USA trading goods with Bangladesh (Exports)
- USA trading goods with Bangladesh (Balance)
- Trade Balance of USA

## 3.3 Preprocessing

After collecting the required data, the dataset is checked for NULL values and there was no missing value throughout the dataset. None of column contain categorical values. A rolling/sliding window consists of past 9 days were formed. In order to prevent potential data leakage, the entire dataset was in divided into train and test sets. 80% of the dataset was used for training and the rest of the 20% for testing. Afterward, the dataset was normalized using the MinMax scaler.

## 3.4 Feature Selection

In this research, to find the relationship between the target and the features in case of the multivariate dataset Pearson's, Spearman's, and Kendall’s correlations were used. When the data is homogenous, Pearson's correlation may not be applicable.

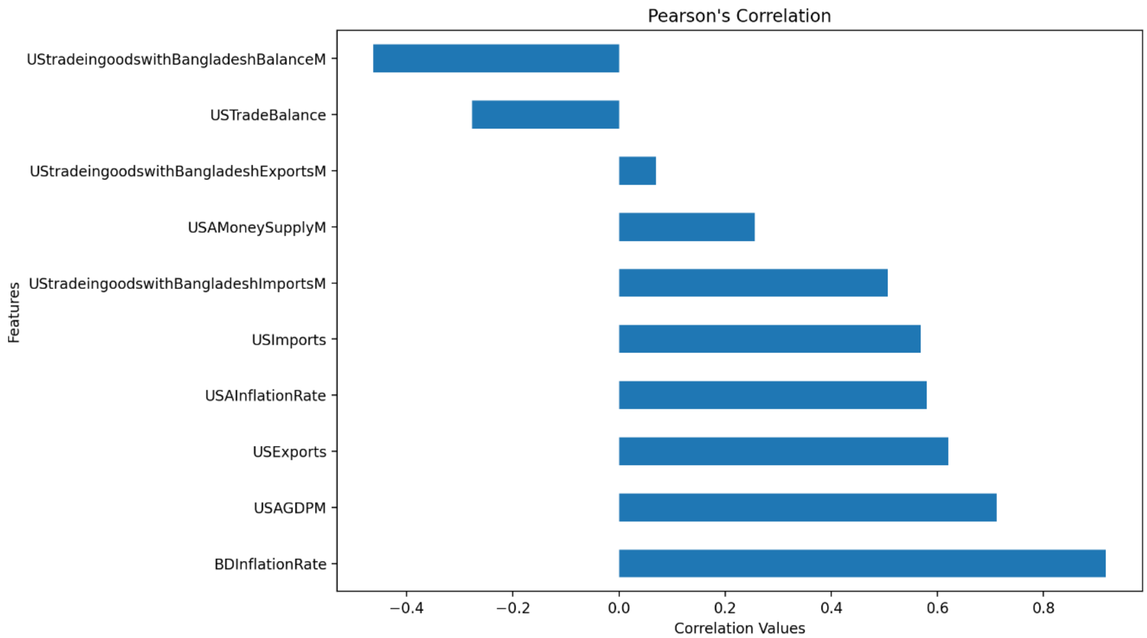


***Figure 3.2 Pearson’s Correlation***

I have used Spearman's correlation because it is a rank-based correlation. Spearman's correlation is the most widely employed approach for determining the correlation between features and the target around the world [5].

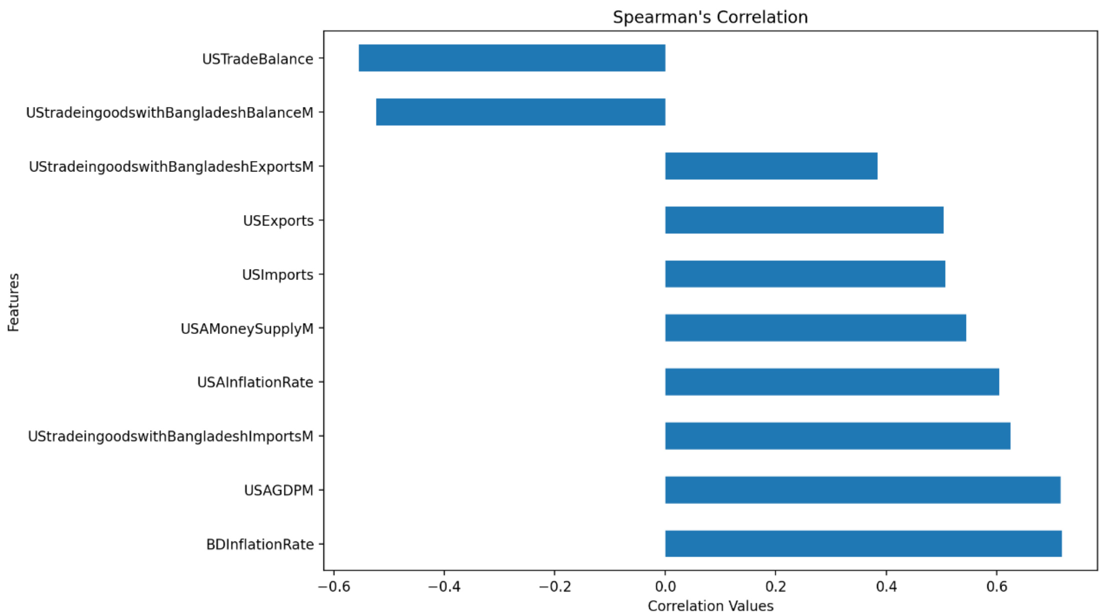


***Figure 3.3 Spearman's Correlation***

Both Pearson's and Spearman's correlation coefficients were computed and compared, and there was difference while selecting the features with higher correlation value than the average of the correlation value. So, Kendall's correlation was calculated and there was no difference between the features selected considering Spearman's and Kendall's correlation.

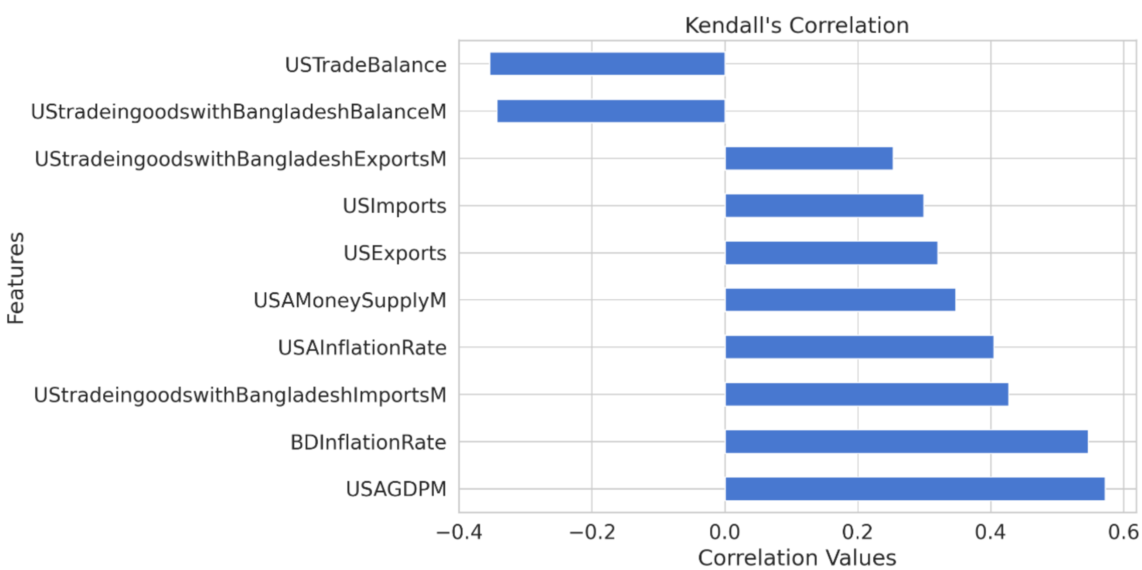


***Figure 3.4 Kendall's Correlation***

## 3.5 Applied Algorithms

For this study, various machine learning models have been implemented such as, DT, KNN, RF, BRR, KRR, XGBoost, AdaBoost, Extra Tree and LSTM.

### 3.5.1 Decision Tree (DT)

A decision tree is an algorithm for machine learning that is commonly used for classification and regression tasks. The algorithm functions by recursively partitioning the data based on the features and determining the optimal division that yields the greatest information gain. The procedure continues until a stopping criterion, such as a maximum depth or a minimum number of instances per leaf node, is met.

In the case of time series prediction, decision trees can be implemented by considering the time series as a sequence of feature vectors and then constructing a decision tree based on these vectors. The tree can then be employed to predict the value of the time series at subsequent time increments. However, that decision trees may not be the optimal algorithm for predicting time series, as they are susceptible to overfitting and may not capture the complex dependencies inherent in time series data.

The hyperparameters and their values used for the univariate implementation of this model are 'criterion' is 'squared_error', 'max_features' is 'auto', 'min_sample_leaf' is 1, 'min_sample_split' is 4 and 'splitter' is random. However, for the multivariate implementation of this model, the hyperparameters and their values were 'criterion' is 'poisson', 'max_features' is 'auto', 'min_sample_leaf' is 1, 'min_sample_split' is 4 and 'splitter' is random.

### 3.5.2 K-Nearest Neighbor (KNN)

K-nearest neighbor (KNN) is a well-known machine learning algorithm that is used for classification and regression tasks. This algorithm operates by finding the k-nearest neighbors of a data point and using their labels or values to predict the label or value of a new data point. The algorithm computes the distance between the new data point and every other data point in the dataset. The labels or values of the k nearest data points are then used to forecast the value of the new data point. KNN can be used to predict time series by treating the time series as a sequence of feature vectors and building a KNN model based on these vectors. The model can then be used to predict the future values of the time series. Notably, KNN may not be the optimal algorithm for time series prediction due to its sensitivity to chaotic data and its potential inability to capture the complex dependencies present in time series data.

The hyperparameters and their values used for the univariate implementation of this model are ‘algorithm’ is ‘ball_tree’ and ‘n_neighbors’ is 5. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘algorithm’ is ‘ball_tree’ and ‘n_neighbors’ is 10.

### 3.5.3 Random Forest (RF)

Random forest is a popular machine learning algorithm used for classification and regression tasks. It operates by constructing a large number of decision trees during training and outputting the class or average prediction for each tree. Randomness is introduced into the tree construction process by the algorithm, both in the selection of features to divide on and in the bootstrapping of training samples. Random forest can be applied to the prediction of time series by treating the time series as a sequence of feature vectors and constructing a random forest model based on these vectors. The model can then be used to predict the future values of the time series.

Random forest is useful for predicting time series because it is less prone to overfitting and can manage high-dimensional data with complex feature interactions.

The hyperparameters and their values used for the univariate implementation of this model are ‘bootstrap’ is False, ‘max_features’ is log2, ‘min_sample_leaf’ is 1, ‘max_depth’ is 20 and ‘n_estimator’ is 100. However, for the multivariate implementation of this model, the hyperparameters and their values where ‘bootstrap’ is True, ‘max_features’ is auto, ‘min_sample_leaf’ is 4, ‘max_depth’ is 20 and ‘n_estimator’ is 50.

### 3.5.4 Bayesian Ridge Regression (BRR)

Bayesian Ridge Regression is a probabilistic machine learning algorithm used in regression tasks for time series prediction. Modeling the relationship between input features and output labels as a linear function with normally distributed errors enables the algorithm to function. Using Bayes' theorem and a prior distribution based on the assumption that the coefficients are normally distributed, the algorithm estimates the posterior distribution of the linear function's coefficients.

Bayesian Ridge Regression is beneficial for predicting time series because it provides probabilistic forecasts, which can be utilized for decision making and uncertainty analysis. In addition, it permits the incorporation of prior knowledge about the distribution of coefficients, which can enhance model performance and generalizability. In addition, it can manage collinear characteristics and prevent overfitting by incorporating regularization.

The hyperparameters and their values used for the univariate implementation of this model are ‘compute_score’ is True, ‘lambda_1’ 1e-08, ‘lambda_2’ is 0.1 4 and ‘tol’ is 1e-05. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘compute_score’ is True, ‘lambda_1’ is 0.1, ‘lambda_2’ is 0.1 and ‘tol’ is 0.1.

### 3.5.5 Kernel Ridge Regression (KRR)

Kernel Ridge Regression (KRR) is an algorithm for machine learning used to estimate time series in regression tasks. It is an extension of ridge regression based on the kernel trick's principles. KRR applies a kernel function to transform the input data into a high-dimensional feature space before applying a linear regression model. This enables KRR to capture intricate nonlinear relationships between input features and output labels. KRR determines the coefficients of the linear function that minimizes both the sum of squared errors between the predicted and actual values and the magnitude of the coefficients. Regularization in KRR is based on the L2-norm of the coefficients, which prevents overfitting and enhances model generalization.

Because it can handle nonlinear relationships between input features and output labels, KRR is beneficial for predicting time series. It is appropriate for modeling nonlinear dynamics because it can capture complex patterns and dependencies in time series data. KRR also permits the use of numerous kernel functions, which can map the input data into high-dimensional feature spaces in order to capture nonlinear relationships more accurately.

The hyperparameters and their values used for the univariate implementation of this model are ‘alpha’ is 1 and ‘kernel’ is linear. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘alpha’ is 0.001 and ‘kernel’ is sigmoid.

### 3.5.6 Extreme Gradient Boosting (XGBoost)

XGBoost (Extreme Gradient Boosting) is a machine learning algorithm that makes predictions using a decision tree ensemble. It is an optimized implementation of gradient boosting that employs multiple regularization techniques to reduce overfitting and enhance performance. XGBoost operates by adding decision trees to the ensemble iteratively, with each new tree attempting to correct the mistakes made by the prior trees. The output of the ensemble is a weighted sum of all the trees' predictions. Each tree's weight is determined by its contribution to reducing the loss function, which quantifies the difference between predicted and actual values.

Because it can manage complex nonlinear relationships between input features and output labels, XGBoost is useful for time series prediction.

The hyperparameters and their values used for the univariate implementation of this model are ‘booster’ is dart, ‘learning_rate’ is 0.001 4 and ‘n_estimators’ is 50. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘booster’ is dart, ‘learning_rate’ is 0.001 and ‘n_estimators’ is 50.

### 3.5.7 Adaptive Boosting (AdaBoost)

Adaboost (Adaptive Boosting) is an algorithm for machine learning that combines weak learners (simple models that perform modestly better than random guesses) to produce a strong learner that can make precise predictions. Adaboost operates by training weak learners iteratively on a weighted version of the training data. At each iteration, the weights of the misclassified samples are elevated, causing subsequent weak learners to place greater emphasis on the misclassified samples. The final forecast is generated by combining the weighted forecasts of all feeble learners. The accuracy of the feeble learners determines the weights assigned to them, with more accurate models receiving greater weights.

Adaboost is useful for predicting time series because it can manage nonlinear relationships between input features and output labels and automatically selects the most informative features.

The hyperparameters and their values used for the univariate implementation of this model are ‘base_estimator’ is DecisionTreeRegressor(max_depth=5), ‘learning_rate’ is 0.001 4 and ‘n_estimators’ is 300. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘base_estimator’ is DecisionTreeRegressor(max_depth=5), ‘learning_rate’ is 0.01 and ‘n_estimators’ is 50.

### 3.5.8 Extra Tree

Extra Trees, also referred to as Extremely Randomized Trees or Extra Random Trees, is a machine learning algorithm that is similar to Random Forest but constructs decision trees differently. Extra Trees constructs multiple decision trees employing random subsets of the input features and a random threshold for each feature, as opposed to Random Forest's optimal threshold. This variability enables Extra Trees to generate a more diverse set of decision trees, thereby reducing overfitting and enhancing the algorithm's generalization capability.

Extra Trees is useful for time series prediction because it can manage complex non-linear relationships between input features and output labels and because it is less susceptible to overfitting than other tree-based algorithms.

The hyperparameters and their values used for the univariate implementation of this model are ‘bootstrap’ is True, ‘max_features’ is auto, ‘min_sample_split’ is 4 and ‘n_estimator’ is 50. However, for the multivariate implementation of this model, the hyperparameters and their values were ‘bootstrap’ is True, ‘max_features’ is auto, ‘min_sample_split’ is 2 and ‘n_estimator’ is 50.

### 3.5.9 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a recurrent neural network (RNN) designed to address the issue of vanishing gradients in conventional RNNs. Because it can model both short-term and long-term dependencies between input features and output labels, LSTM is an effective algorithm for time series prediction. LSTM maintains an internal memory state that can retain information about previous inputs and has a set of gates that regulate the flow of information into and out of the memory state. Based on the task at hand and the significance of the input features, the gates can learn which information to retain and which information to discard.

LSTM is valuable for predicting time series because it can account for both short- and long-term dependencies in the input data. It can also deal with sequences of variable length and absent data, making it a flexible algorithm for a variety of time series prediction tasks. In addition, LSTM can automatically learn which input features are most essential for a given task and generate probabilistic predictions that account for the uncertainty of those predictions.

# CHAPTER 4

# RESULT AND ANALYSIS

## 4.1 Performance Evaluation

In order to conduct the study, the entire dataset was divided into a training dataset and a testing dataset in a ratio of 80:20. The models were trained on the training dataset and then applied to the test dataset to obtain the predicted values. RMSE, MAPE, and R-Squared Error metrics were utilized to assess the models.

### 4.1.1 Root Mean Squared Error (RMSE)

Root Mean Squared Error is a widely used metric for measuring the accuracy of a regression model by comparing the predicted values to the actual values. The root square of the average of the squared differences between the predicted and actual values returns the RMSE. It provides a measure of the magnitude of the model's defects in predicting the outcome variable, with a smaller value indicating a better fit between the predicted and actual values.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

Here, N is the number of test sample.

### 4.1.2 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percent Error is an extensively employed metric for assessing the precision of a regression model. It measures the error by considering the absolute difference between the actual values and predicted values. That absolute difference values are then divided by the actual value and multiply the result by 100.

$$MAPE = \left(\frac{1}{N}\sum_{i=1}^{N}\left|\frac{Actual_i - Predicted_i}{Actual_i}\right|\right) * 100$$

Here, N is the number of test sample.

### 4.1.3 R-squared ($R^2$)

R-squared is a statistical measure which reflects the proportion of the variance in the result variable that can be clarified through the model in the context of evaluating machine learning predicted values. R-squared is computed as the ratio of the total sum of squared differences between the actual values and the mean of the outcome variable to the sum of squared differences between the predicted values and the mean of the outcome variable.

R-squared is an aggregate measure of the model's fit, indicating how well the model explains the variance of the outcome variable. It is an often-used metric in regression analysis because it provides a uniform measurement of the model's performance which can be utilized for comparing models.

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y}_i)^2}$$

Here,

$SS_{RES}$ = Sum of Squares of Residuals

$SS_{TOT}$ = Total Sum of Squares of Errors

## 4.2 Result Analysis

In this section, we discuss the results of the study. Analysis of the models trained on the univariate dataset from Table 4.1 revealed that regression models, specifically BRR and KRR, achieved the best results with RMSE values of 1.2429, 1.1834, and 1.274, respectively. Following the regression models, LSTM produced the most accurate result with an RMSE value of 1.6958. Out of all the models trained on the univariate dataset, BRR was the top performer.

***Table 4.1: Results for univariate forecasting***

| Model | RMSE | MAPE | $R^2$ |
|---|---|---|---|
| DT | 11.5070 | 9.47 % | -2.0681 |
| KNN | 11.5396 | 9.51 % | -2.0855 |
| RF | 11.4812 | 9.44 % | -2.0544 |
| BRR | 1.1834 | 0.63 % | 0.9717 |
| KRR | 1.2740 | 0.68 % | 0.9624 |
| XGBoost | 2.6960 | 2.08 % | 0.8316 |
| AdaBoost | 11.5426 | 9.52 % | -2.0871 |
| Extra Tree | 11.5410 | 9.51 % | -2.0863 |
| LSTM | 1.6958 | 1.12 % | 0.9333 |

Table 4.2 displays the performance measures of the models trained on the multivariate dataset using past 9 days and microeconomic factors as features. While regression models and LSTM had superior results on the univariate dataset, it was found that no models, except for BRR, performed well on the multivariate dataset, with a PMSE value of 1.556 in the test data.

***Table 4.2: Results for multivariate forecasting***

| Model | RMSE | MAPE | $R^2$ |
|---|---|---|---|
| DT | 11.4450 | 9.33 % | -1.9354 |
| KNN | 12.2338 | 10.71 % | -2.3539 |
| RF | 11.3849 | 9.25 % | -1.9046 |
| BRR | 1.5560 | 7.71 % | 0.9457 |
| KRR | 12.3888 | 10.54 % | -2.4395 |
| XGBoost | 11.9745 | 10.01 % | -2.2132 |
| AdaBoost | 11.3724 | 9.24 % | -1.8983 |
| Extra Tree | 11.3716 | 9.23 % | -1.8978 |
| LSTM | 3.8859 | 2.86 % | 0.6616 |

### 4.2.1 Predicted VS Actual Price Graphs

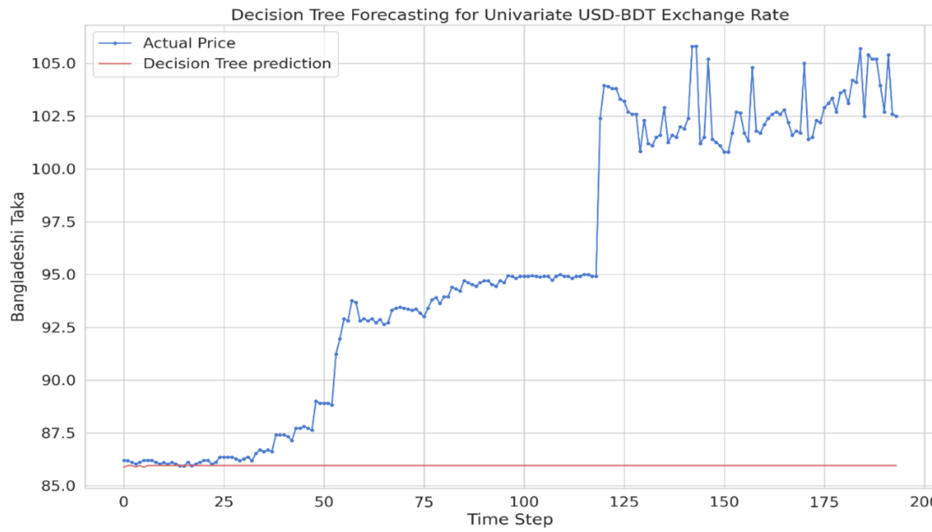


*Figure 4.1 Forecasting of Decision Tree Trained on Univariate Dataset*

The predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.507, 9.47% & -2.0681 respectively.

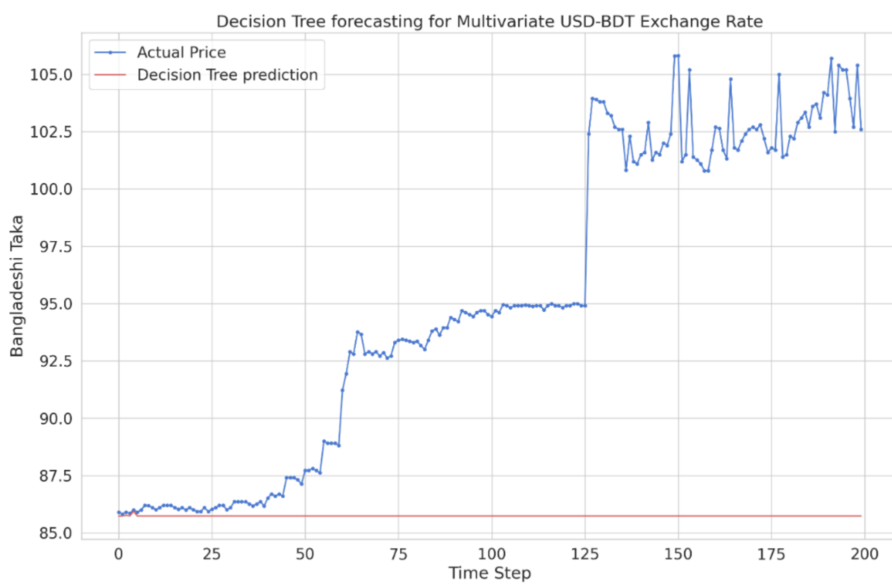


*Figure 4.2 Forecasting of Decision Tree Trained on Multivariate Dataset*

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.445, 9.33% & -1.9354 respectively.

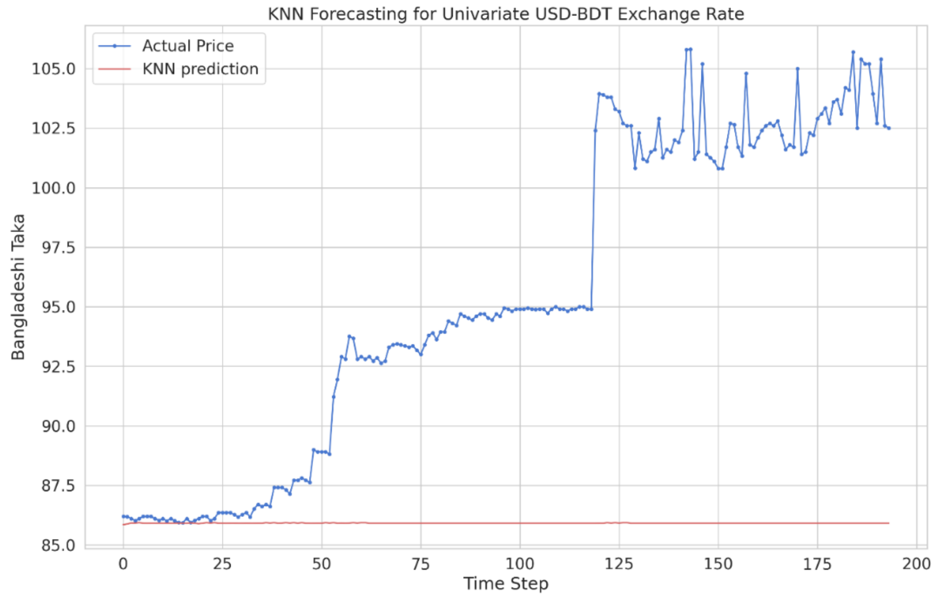


***Figure 4.3 Forecasting of KNN Trained on Univariate Dataset***

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.5396, 9.51% & -2.0855 respectively.

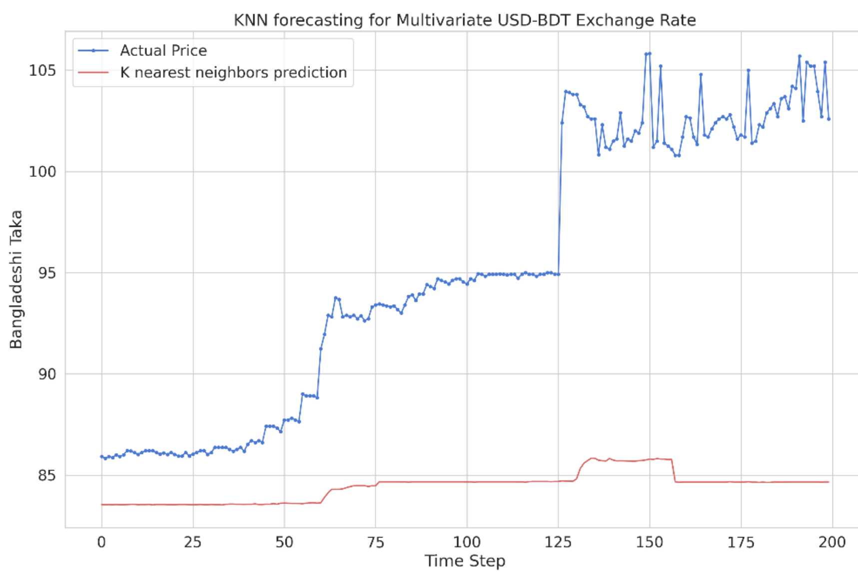


***Figure 4.4 Forecasting of KNN Trained on Multivariate Dataset***

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 12.6126, 10.69% & -2.5648 respectively.

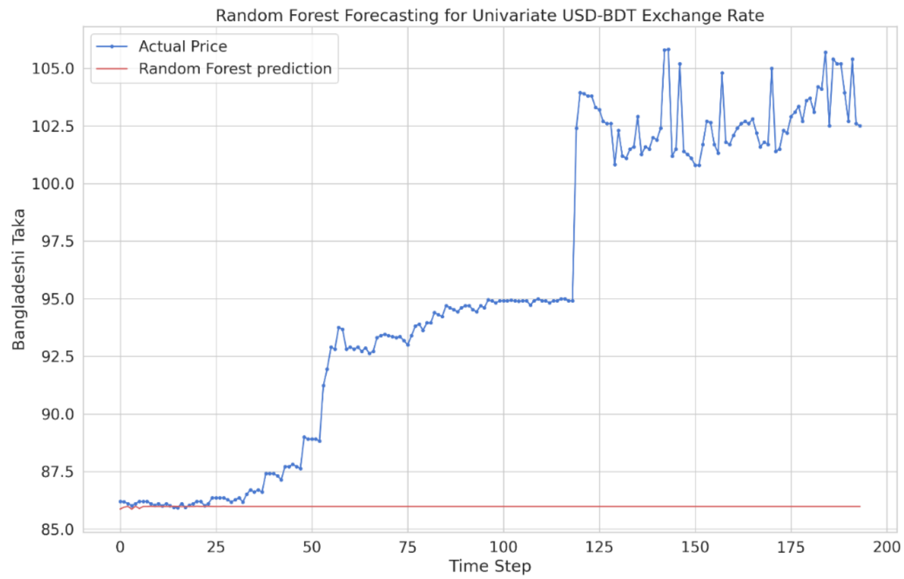


***Figure 4.5** Forecasting of Random Forest Trained on Univariate Dataset*

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.4812, 9.44% & -2.0544 respectively.

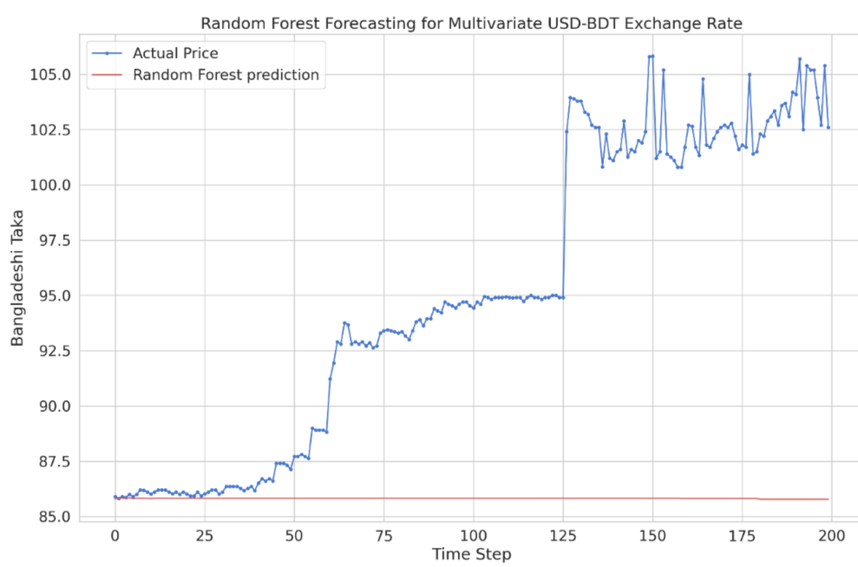


***Figure 4.6 Forecasting of Random Forest Trained on Multivariate Dataset***

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.3849, 9.25% & -1.9046 respectively.

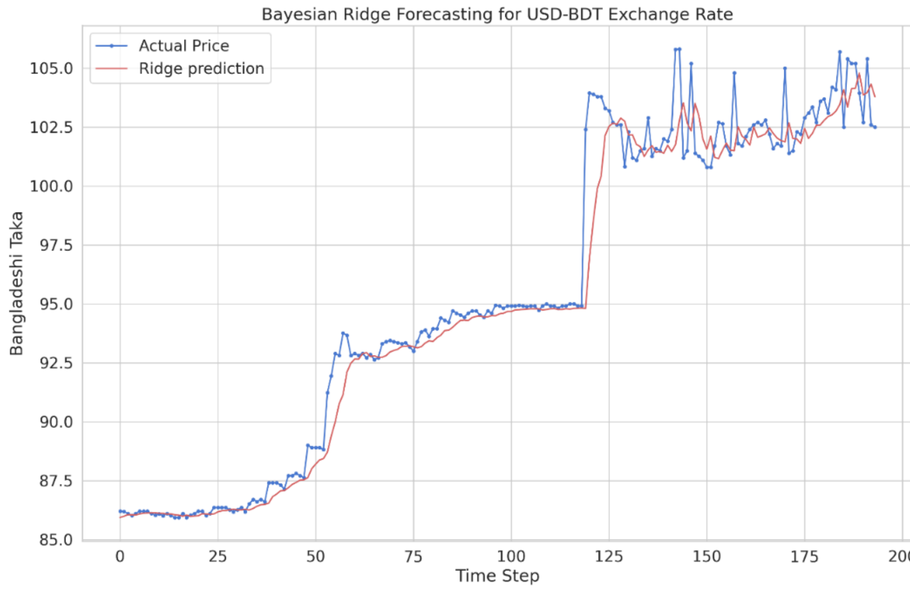


***Figure 4.7* Forecasting of Bayesian Ridge Regression Trained on Univariate Dataset**

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 1.1834, 0.63% & 0.9717 respectively.

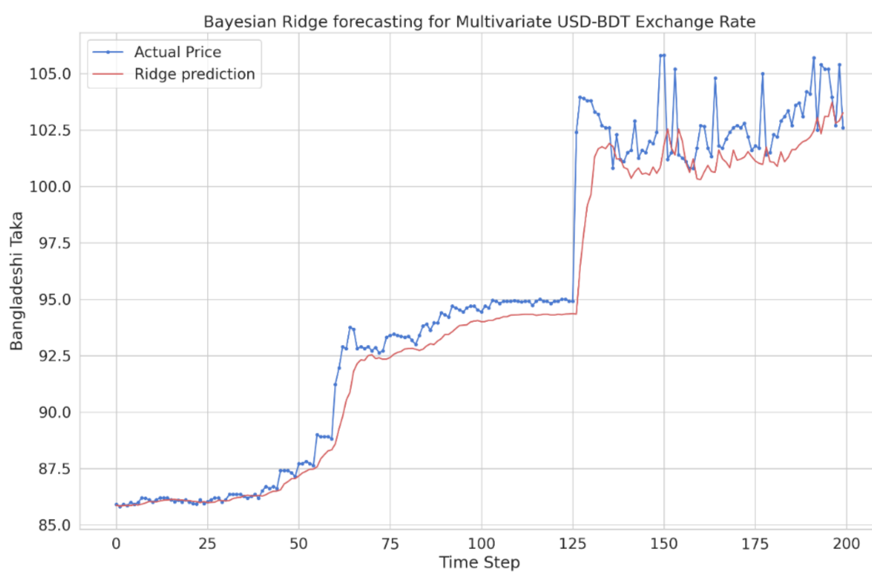


***Figure 4.8* Forecasting of Bayesian Ridge Regression Trained on Multivariate Dataset**

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 1.556, 7.71% & 0.9457 respectively.

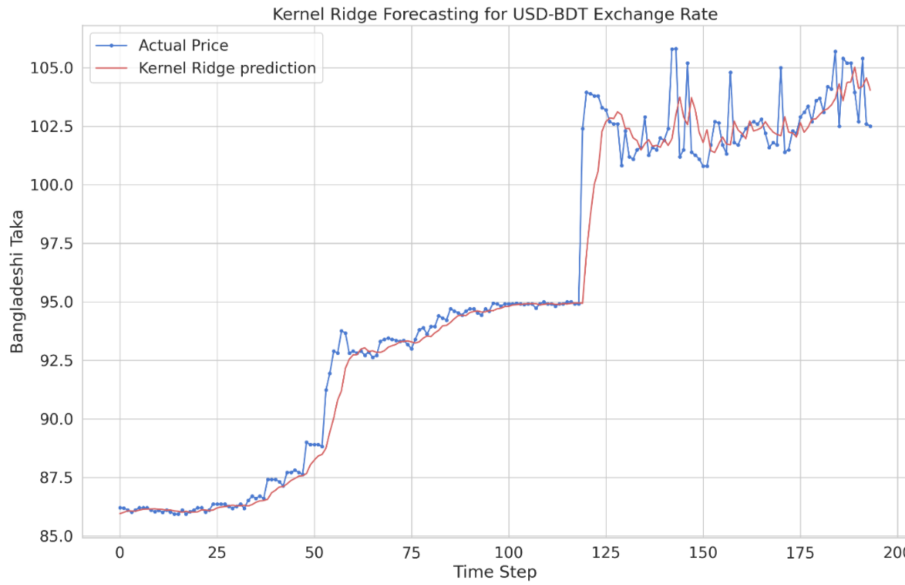


*Figure 4.9* ***Forecasting of Kernel Ridge Regression Trained on Univariate Dataset***

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 1.274, 0.68% & 0.9624 respectively.

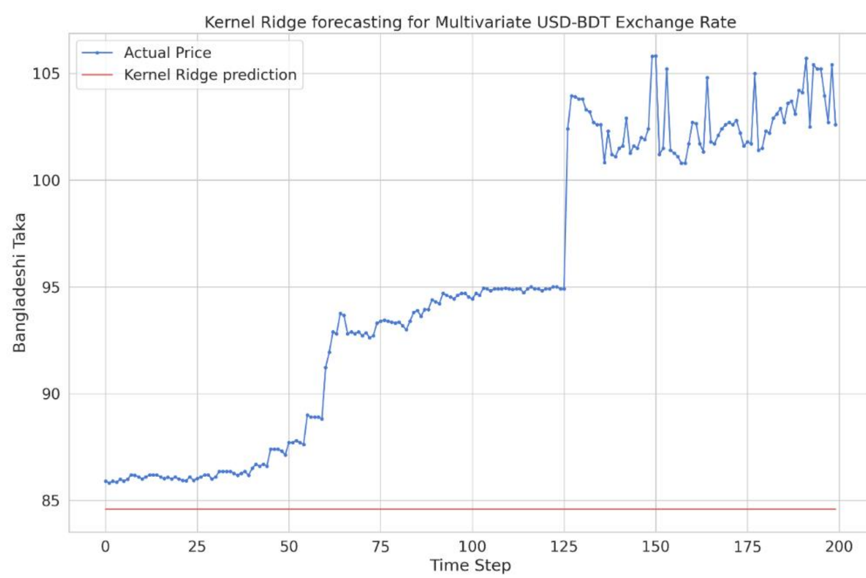


***Figure 4.10 Forecasting of Kernel Ridge Regression Trained on Multivariate Dataset***

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 12.3888, 10.54% & -2.4395 respectively.

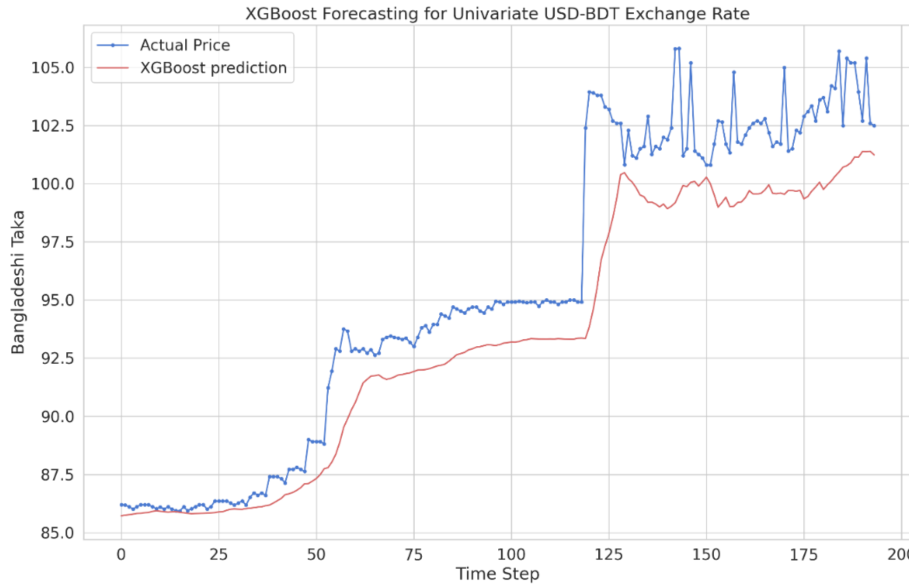


*Figure 4.11 Forecasting of XGBoost Trained on Univariate Dataset*

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 2.696, 2.08% & 0.8316 respectively.

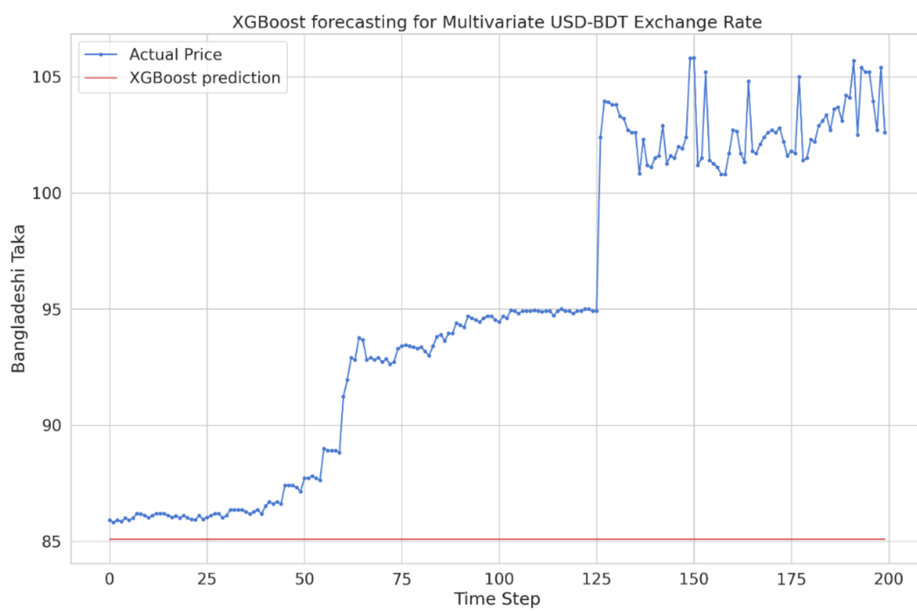


*Figure 4.12 Forecasting of XGBoost Trained on Multivariate Dataset*

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.9745, 10.01% & -2.2132 respectively.

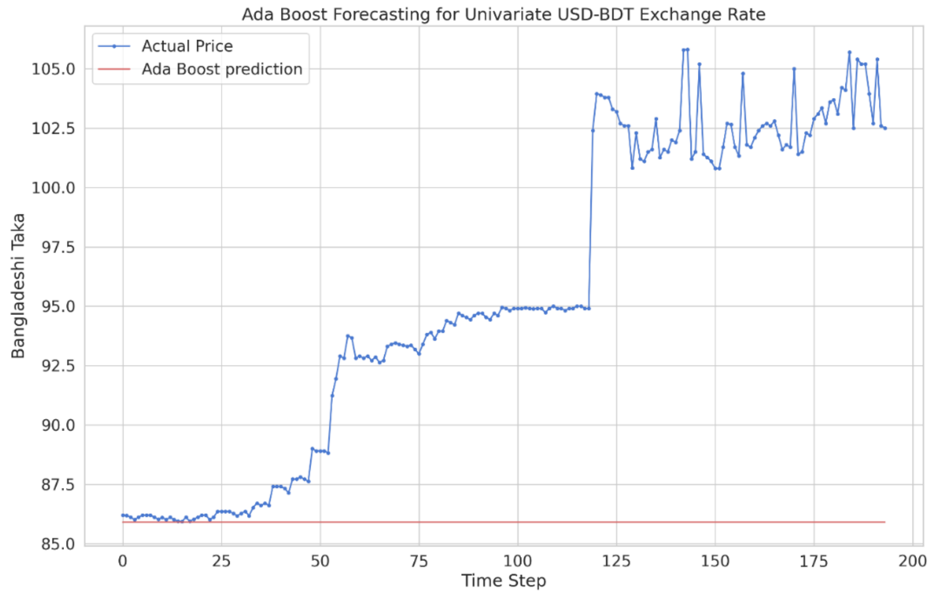


***Figure 4.13 Forecasting of AdaBoost Trained on Univariate Dataset***

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.5426, 9.52% & -2.0871 respectively.

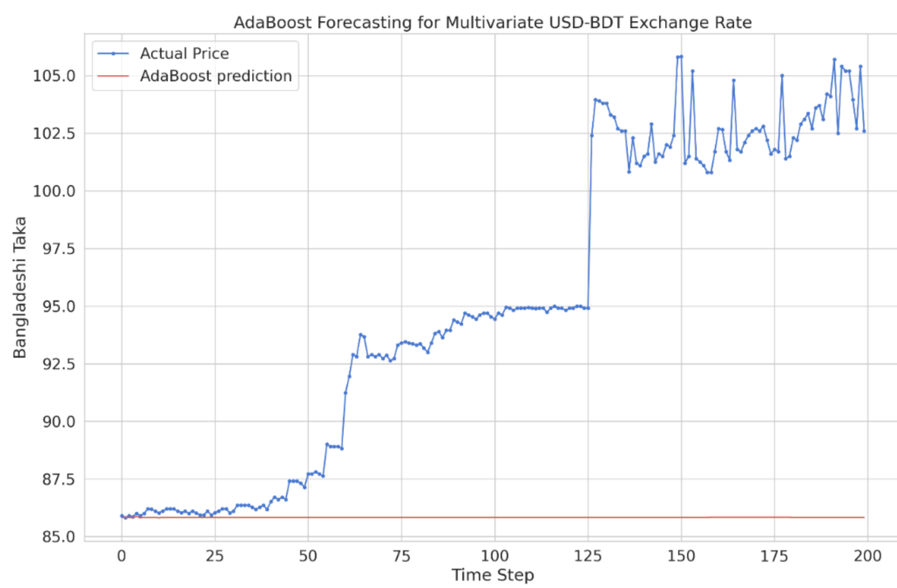


***Figure 4.14 Forecasting of AdaBoost Trained on Multivariate Dataset***

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.3724, 9.24% & -1.8983 respectively.

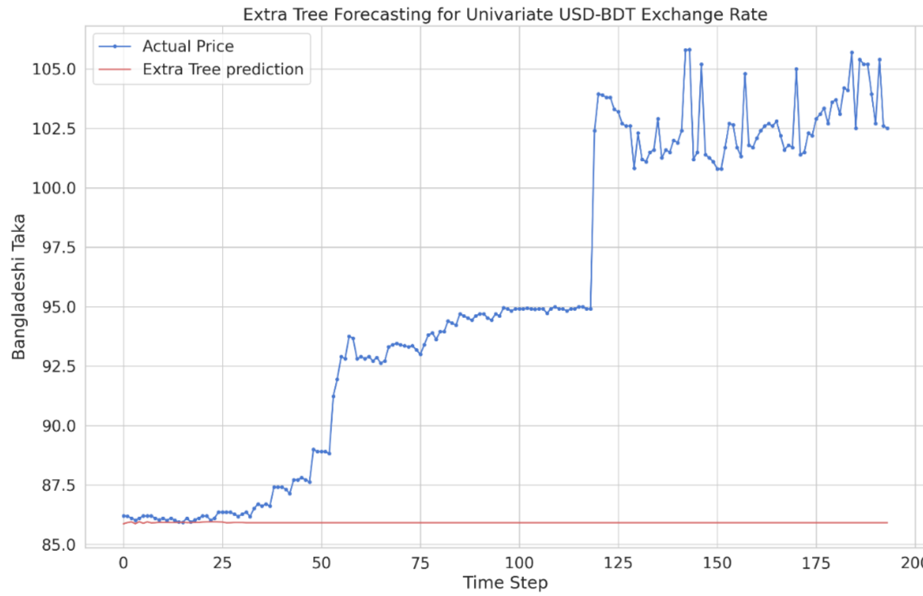


***Figure 4.15 Forecasting of Extra Tree Trained on Univariate Dataset***

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.541, 9.51% & -2.0855 respectively.

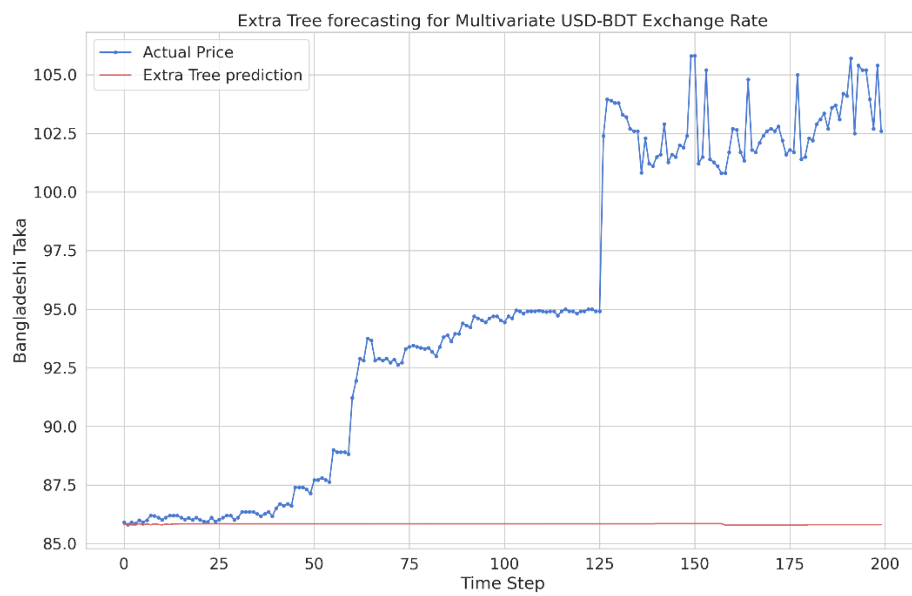


***Figure 4.16 Forecasting of Extra Tree Trained on Univariate Dataset***

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 11.3716, 9.23% & -1.8978 respectively.

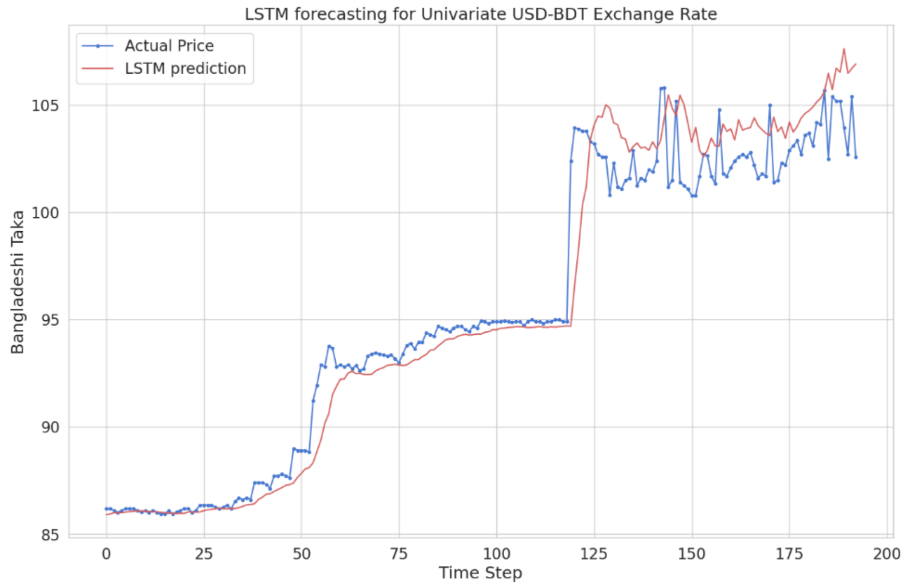


*Figure 4.17 Forecasting of LSTM Trained on Univariate Dataset*

The predicted exchange rate of KNN trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 1.6958, 1.12% & 0.9333 respectively.

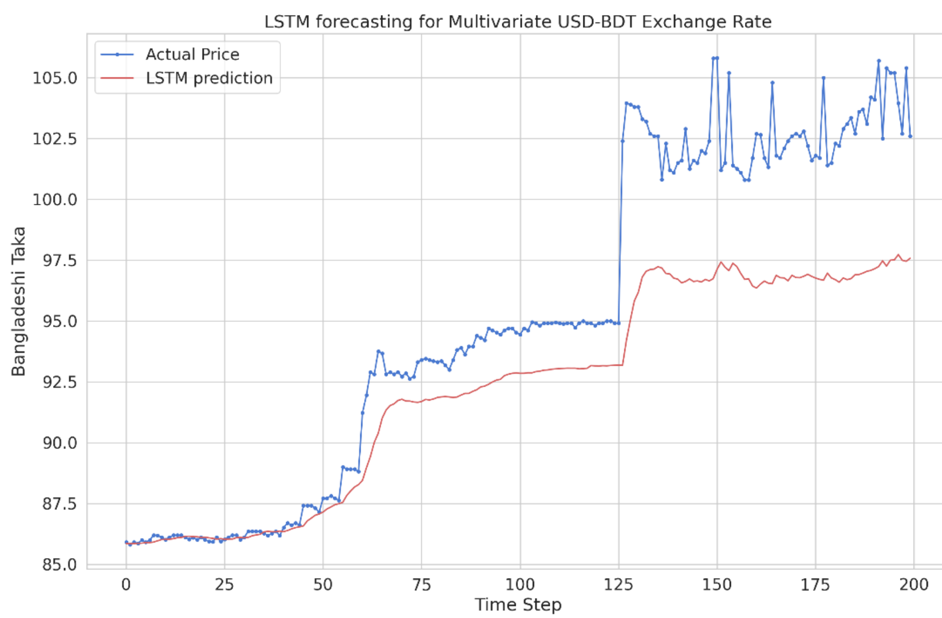


*Figure 4.18 Forecasting of LSTM Trained on Multivariate Dataset*

However, the predicted exchange rate of decision tree trained on univariate dataset is shown in Figure 4.2.1. The RMSE, MAPE and $R^2$ value of this forecasting is 3.8859, 2.86 % & 0.6616 respectively.

# CHAPTER 5

# CONCLUSION AND FUTURE WORK

## 5.1 Conclusion and Future Work

The implemented models in this study were analyzed, and it was found that BRR had the best performance on both univariate and multivariate datasets. However, BRR demonstrated superior performance on the univariate dataset. Based on the results, it can be concluded that the Bayesian Ridge Regression (BRR) model is suitable for short-term forecasting using only past days as features.

In the future, this study can be expanded by training models with datasets that include more relevant microeconomic factors, and microeconomic factors those are measured on a daily basis. The deployment of additional deep learning models will lead to a more comprehensive comprehension of the problem at hand. Additionally, sentiment analysis can be employed to gain a better understanding of economic trends and improve the performance of the models for forecasting exchange rates.

## References

[1] J. M. Rizzo, “The economic determinants of the choice of an exchange rate regime: a probit analysis,” *Econ Lett*, vol. 59, no. 3, pp. 283–287, Jun. 1998, doi: 10.1016/S0165-1765(98)00056-1.

[2] M. Auboin and M. Ruta, “The relationship between exchange rates and international trade: a literature review,” *World Trade Review*, vol. 12, no. 3, pp. 577–605, 2013, doi: 10.1017/S1474745613000025.

[3] C. M. Reinhart and K. S. Rogoff, “The Modern History of Exchange Rate Arrangements: A Reinterpretation,” *Q J Econ*, vol. 119, no. 1, pp. 1–48, Feb. 2004, doi: 10.1162/003355304772839515.

[4] K. Mohammed, K. Uddin, G. M. Azmal, A. Quaosar, and D. C. Nandi, “FACTORS AFFECTING THE FLUCTUATION IN EXCHANGE RATE OF THE BANGLADESH: A CO-INTEGRATION APPROACH”.

[5] A. Biswas, I. A. Uday, K. M. Rahat, Mst. S. Akter, and M. R. C. Mahdy, “Forecasting the United State Dollar(USD)/Bangladeshi Taka (BDT) exchange rate with deep learning models: Inclusion of macroeconomic factors influencing the currency exchange rates,” *PLoS One*, vol. 18, no. 2, p. e0279602, Feb. 2023, doi: 10.1371/journal.pone.0279602.

[6] F. Ahmed and J. A. Keya, “Time Series Analysis for Predicting the Exchange Rate of USD to BDT,” 2016, doi: 10.5281/zenodo.3369751.

[7] Md. S. Mia, Md. S. Rahman, and S. Das, “Forecasting the BDT/USD Exchange Rate: An Accuracy Comparison of Artificial Neural Network Models and Different Time Series Models,” *Journal of Statistics Applications & Probability Letters*, vol. 4, no. 3, pp. 131–138, Sep. 2017, doi: 10.18576/jsapl/040304.

[8] R. Ramasamy and S. K. Abar, “Influence of Macroeconomic Variables on Exchange Rates,” *Journal of Economics, Business and Management*, vol. 3, no. 2, pp. 276–281, 2015, doi: 10.7763/JOEBM.2015.V3.194.

[9] A. Nicholas Refenes, A. Zapranis, and G. Francis, “Stock performance modeling using neural networks: A comparative study with regression models,” *Neural Networks*, vol. 7, no. 2, pp. 375–388, Jan. 1994, doi: 10.1016/0893-6080(94)90030-2.

[10] M. Rehman, G. M. Khan, and S. A. Mahmud, “Foreign Currency Exchange Rates Prediction Using CGP and Recurrent Neural Network,” *IERI Procedia*, vol. 10, pp. 239–244, 2014, doi: 10.1016/j.ieri.2014.09.083.

[11] M. S. Islam and E. Hossain, “Foreign exchange currency rate prediction using a GRU-LSTM hybrid network,” *Soft Computing Letters*, vol. 3, p. 100009, Dec. 2021, doi: 10.1016/j.socl.2020.100009.

[12] T. N. Pandey, A. K. Jagadev, S. Dehuri, and S.-B. Cho, “A novel committee machine and reviews of neural network and statistical models for currency exchange rate prediction: An experimental analysis,” *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 9, pp. 987–999, Nov. 2020, doi: 10.1016/j.jksuci.2018.02.010.

[13] M. Rout, B. Majhi, R. Majhi, and G. Panda, “Forecasting of currency exchange rates using an adaptive ARMA model with differential evolution based training,” *Journal of King Saud University - Computer and Information Sciences*, vol. 26, no. 1, pp. 7–18, Jan. 2014, doi: 10.1016/j.jksuci.2013.01.002.

[14] M. M. Panda, S. N. Panda, and P. K. Pattnaik, “Multi currency exchange rate prediction using convolutional neural network,” *Mater Today Proc*, Jan. 2021, doi: 10.1016/J.MATPR.2020.11.317.

[15] J. Brownlee, “How to Avoid Data Leakage When Performing Data Preparation - MachineLearningMastery.com,” Jun. 22, 2020. https://machinelearningmastery.com/data-preparation-without-data-leakage/ (accessed May 07, 2023).