Using Statistical Model and Machine Learning Algorithms to Predict Stock Prices of Some Corporations in Vietnam

1st Tran Thi Tuyet Linh   
IS403.N21.HTCL  
University of Information TechnologyHo Chi Minh city  
[20521540@gm.uit.edu.vn](mailto:20521540@gm.uit.edu.vn)

2nd To Thai Duy   
IS403.N21.HTCL  
University of Information TechnologyHo Chi Minh city  
[20521247@gm.uit.edu.vn](mailto:20521247@gm.uit.edu.vn)

3rd Nguyen Viet Hoang   
IS403.N21.HTCL  
University of Information TechnologyHo Chi Minh city  
[20521348@gm.uit.edu.vn](mailto:20521348@gm.uit.edu.vn)

*Abstract* — Stock trading has always been a significant part of the financial world. With the potential of the stock market and the rapid development of machine learning, stock prices prediction has been a hot issue in recent years. Various methods have been proposed to predict stock prices. However, each has unique characteristics that cause differences in performance. In this study, we use Statistical Model and Machine Learning Algorithms such as Linear Regression, ARIMA, SARIMAX, Random Forest, DLM, RNN, LSTM, GRU and XGBoost to predict the stock prices of some corporations in Vietnam based on past prices. The comparison results will be based on three evaluation parameters: RMSE, MAE, and MAPE.

Keywords — Stock prices, Linear Regression, ARIMA, SARIMAX, Random Forest, DLM, RNN, LSTM, GRU, XGBOOST

# Introduction

The prediction of stock prices has always been a challenging task in the financial market. Accurate forecasting of stock prices is important for investors seeking to make decisions. In recent years, the utilization of machine learning algorithms in stock price prediction has garnered significant attention. Machine learning algorithms possess the ability to automatically learn from historical data and identify the best predictive models based on their variations. This enables the generation of potential forecasts for stock prices based on current and past information.

Based on this premise, this research aims to construct a stock price prediction model using various machine learning algorithms. Historical data of stock prices, financial indicators, and market factors will be employed to train the models. Subsequently, the models will be evaluated and tested on new data to determine their accuracy and performance.

The machine learning algorithms considered in this study include Linear Regression, ARIMA, SARIMAX, Random Forest, DLM, RNN, LSTM, GRU and XGBOOST. Each algorithm offers unique capabilities and approaches to capture the underlying patterns and dynamics in stock price movements.

The study aims to explore the effectiveness of these algorithms in forecasting stock market trends and provide valuable insights for investors and financial analysts. By using historical data and advanced machine learning techniques, the research aims to enhance the accuracy and reliability of stock price predictions, facilitating informed decision-making in the dynamic world of financial markets.

# Related Word

A stock market is where companies issue their shares to expand their business, and investors can buy or sell each other's shares at certain prices. Investors around the world can buy and sell stocks, thus making a profit by selling at a higher price than they bought. The challenge is fluctuating stock price movements can change within minutes or seconds. Therefore, the theory of predicting stock prices emerges.

In Dias Satria’s research he concludes that the ARIMA Box-Jenkins modeling is unsuitable for predicting stock prices. This is due to the data’s nonlinear characteristics, which causes the assumption of white noise in the estimation of the ARIMA Box-Jenkins parameter to be violated. Also, he compared three models: RNN, LSTM, GRU, within GRU presented the best performance in the case of predicting the stock prices based on RMSE value [1].

In another article, Xiwen Jin and Chaoran Yi have concluded that the LSTM and the GRU perform relatively better results and the Random Forest is the worst. The R2 score for different models they have analyzed: LSTM 0.84, GRU 0.86, Random Forest Regressor 0.51, XGBoost Regressor 0.69, Linear Regression 0.73 and LGBM Regressor 0.72. We can see that XGBoost, and Random Forest didn’t have good performance compared to LSTM and GRU models [2].

In another article, Jinlong Ruan, Wei Wu and Jiebo Luo compared LSTM with SARIMAX model. The difference between average accuracies of LSTM and SARIMAX models is minor from either a holistic or a regional perspective. Nevertheless, as the amount of training data increases the class of SARIMAX models falls into a competitive disadvantage: its running time grows at a geometric rate. In comparison, the running time of LSTM models is relatively constant, which reinforces the argument that LSTM outperforms SARIMAX in handling enormous data [3].

# Data

## Data sources

### GAS

A screenshot of a spreadsheet

Description automatically generated with medium confidence

Figure . Data of GAS

The data is about the stock price of Petrovietnam Gas JSC (GAS) from 5/6/2018 to 5/6/2023.

Source data: Investing

### HPG

A picture containing text, screenshot, number, font

Description automatically generated

Figure . Data of HPG

The data is about the stock price of Hoa Phat Group JSC (HPG) from 5/6/2018 to 5/6/2023.

Source data: Investing

### VPB

A picture containing text, screenshot, number, font

Description automatically generated

Figure . Data of VPB

The data is about the stock price of Vietnam Prosperity Joint Stock Commercial Bank (VPB) from 5/6/2018 to 5/6/2023

Source data: Investing

In this research, we will predict the close price for the next 30 days (6/6/2023 – 6/7/2023).

## Descriptive Statistics

### GAS

|  |  |
| --- | --- |
| **Value** | **Result** |
| Mean | 94291.68825 |
| Standard Error | 432.7113026 |
| Median | 97200 |
| Mode | 105000 |
| Standard Deviation | 15304.77306 |
| Sample Variance | 234236078.4 |
| Kurtosis | -0.573062611 |
| Skewness | -0.48998054 |
| Range | 76542 |
| Minimum | 53900 |
| Maximum | 130442 |
| Sum | 117958902 |
| Count | 1251 |

Table . Descriptive of GAS

### HPG

|  |  |
| --- | --- |
| **Value** | **Result** |
| Mean | 20058.99241 |
| Standard Error | 281.2951062 |
| Median | 16900 |
| Mode | 20800 |
| Standard Deviation | 9949.261174 |
| Sample Variance | 98987797.91 |
| Kurtosis | -0.671410924 |
| Skewness | 0.788744918 |
| Range | 36484 |
| Minimum | 7411.8 |
| Maximum | 43895.8 |
| Sum | 25093799.5 |
| Count | 1251 |

Table . Descriptive of HPG

### VPB

|  |  |
| --- | --- |
| **Value** | **Result** |
| Mean | 14247.12126 |
| Standard Error | 188.6386895 |
| Median | 10926.1 |
| Mode | 7926 |
| Standard Deviation | 6672.052049 |
| Sample Variance | 44516278.55 |
| Kurtosis | -1.450534918 |
| Skewness | 0.392879807 |
| Range | 20648.4 |
| Minimum | 6277.8 |
| Maximum | 26926.2 |
| Sum | 17823148.7 |
| Count | 1251 |

Table . Descriptive of VPB

## Visualization

### GAS

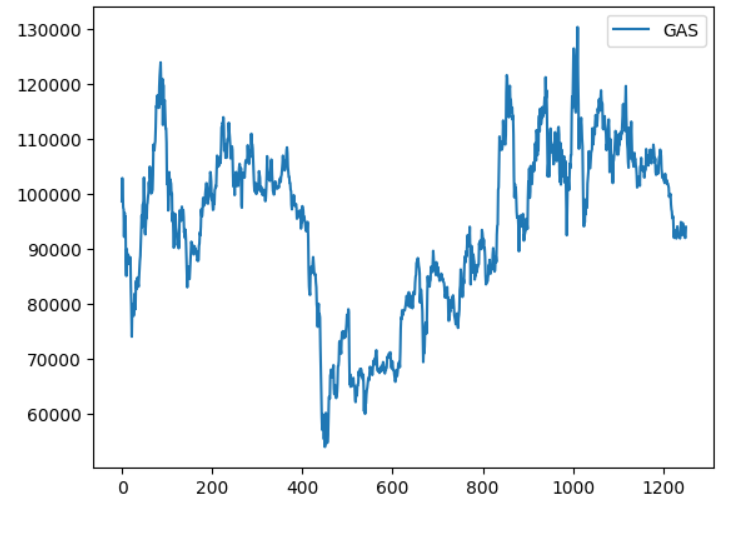


Figure 4. Visualization of GAS close price

### HPG

A picture containing screenshot, plot, line, text

Description automatically generated

Figure 5. Visualization of HPG close price

### VPB

A picture containing text, screenshot, plot, line

Description automatically generated

Figure 6. Visualization of VPB close price

# Methods

## Linear Regression

*Simple Linear Regression* estimates the relationship between a scalar response *y* and a single explanatory variable *x* (also called dependent variable *y* and independent variable 𝑥), given a set of data that includes observations for both of these variables for a particular sample. [4]

The basic model for multiple linear regression is:

(1)

Where: *Y* is dependent variable; *X1, … Xm* is the distinct independent or predictor variables; *β0* is the y-intercept (value of y when all other parameters are set to 0); *β1, . . . βm*is the regression coefficient of the independent variable; ε is model error

## ARIMA

ARIMA is the abbreviation for “Autoregressive integrated moving average”. The ARIMA model is popularly used to forecast univariate time series data. ARIMA model can handle a time series if it is stationary and has no data missing [5]. This method is used in multiple studies for forecasting.

ARIMA models are expressed in the form of ARIMA (p,d,q) [6]. All p, d, and q are non-negative numbers.

### AR is Auto Regression, and p is the number of autoregressive terms [2]. The equation for AR model is:

### (2)

in which:

is the current value; is the constant term; p is the number of orders; is the auto-regression coefficient and is the error

### MA is the Moving Average, and q is the number of terms in the moving average [5]. The equation for MA model is:

(3)

Where: is the current value; is the constant term; p is the number of orders; is the moving average coefficient and is the error

### Last, the I part is Integrated, and d is the number of differences (order) required to make it a stationary sequence.

After combining them, we will have the ARIMA(p, d, q) express as follow:

(4)

The first step to apply ARIMA model is identification of the time series. An Augmented Dicky–Fuller (adf) unit-root test shows if the dataset is stationary or not. If unit root exists then the time series is considered to be non-stationary[6]. If the time series is found to be stationary then we can use the ARMA model to estimate and forecast. But if it is not stationary then in order to apply ARIMA it has to be converted into stationary by differencing. After identification, ARIMA models are estimated for the specific stationary time series.

## SARIMAX

SARIMAX stands for Seasonal Autoregressive Integrated Moving Average with Exogenous Variables. The SARIMAX model is an advancement of the seasonal ARIMA model with external variables, called SARIMAX (p, d, q) (P,D,Q)s (X), where X is the vector of external variables; p, d and q are the order of AR autocorrelation, the degree of difference, and the order of the moving average part, respectively, extended by P, D and Q to handle seasonality, which is referred to as the seasonal part of the model, and s is the number of periods per season [7]. The external variables can be modeled by multi linear regression equation is expressed as equation (1).

## Random Forest

Random forests as described here were introduced by ***Breiman*** (2001), although many of the ideas had cropped up earlier in the literature in different forms. It has many applications in solving classification and regression problems and needs to optimize a few parameters. There are two parameters in the RF model that generally affect the performance of the model: the number of trees *ntree* and number of candidate variables randomly sampled at each split *ntry*. [8]

1*. For b = 1 to B*:

(a) Draw a bootstrap sample of size N from the training data.

(b) Grow a random-forest tree to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size is reached.

* 1. Select m variables at random from the *p* variables.
  2. Pick the best variable/split-point among the *m*.
  3. Split the node into two daughter nodes.

2. Output the ensemble of trees.

To make a prediction at a new point x:

(5)

## DLM

Dynamic linear models (DLM) offer a very generic framework to analyze time series data. The models can be seen as general regression models where the coefficients can vary in time. In addition, they allow for a state space representation and a formulation as hierarchical statistical models, which in turn is the key for efficient estimation by Kalman formulas and by Markov chain Monte Carlo (MCMC) methods. A dynamic linear model can handle non-stationary processes, missing values and non-uniform sampling as well as observations with varying accuracies [9].

Dynamic Linear Models are a special case of general state-space models where the state and the observation equations are linear, and the distributions follow a normal law. Generalized DLMs relax the assumption of normality by allowing the distribution to be any of the exponential family of functions (which includes the Bernoulli, binomial and Poisson distributions, useful for count data).

There are two constitutive operations for dynamic linear models: filtering and smoothing. In a few words, filtering is the operation consisting in estimating the state values at time t, using only observations up to (and including) t-1. On the contrary, smoothing is the operation which aims at estimating the state values using the whole set of observations [10].

## RNN

## RNN is one of the components of an artificial neural network with feedback connections closed by loop. RNNs are called “repetitive” because they perform the same task for each sequence element by leveraging previously captured information sequentially to predict future unknown sequential data. The general structure of RNN loops and the looping modules are shown in Figure 4.[1]

A diagram of a flowchart

Description automatically generated with low confidence

Figure . RNN Architecture

In RNN network systems, different time step lengths have the same one weight and can be cyclically linked across time. Due to the sharing of weights, the temporal parameters in the RNN network system are greatly reduced. [11]

One of the drawbacks is that the RNN network only store some of the data steps in the previous sequence. Thus, RNNs are not suitable for storing longer sequences. Another type of recurrent network, LSTM, was introduced to overcome this shortcoming [11].

## LSTM

LSTM uses one of the most common forms of RNN. This time recurrent neural network is meant to avoid long-term dependence problems and is suitable for processing and predicting time series. The LSTM model filters information through the gate structure to maintain and update the state of memory cells. Its door structure includes input, forgotten, and output gates. Each memory cell has three sigmoid layers and one tanh layer. [12]

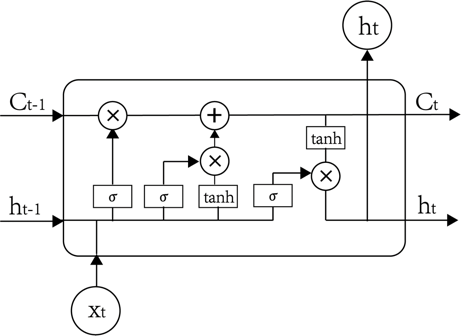


Figure . Structure of LSTM memory cells.

Sigmoid function:   *(6)*

Tanh activation function  *(7)*

Forget gate: *(8)*

Input gate: *(9)*

(Where the calculation methods of *it* and *Ĉt* are shown in Eqs. 10 and 11)

*(10)*

*(11)*

Output gate [13]: *(12)*

*(13)*

## GRU

The GRU model is a modified version of the LSTM model, it not only merges the forget gate and the input gate into an update gate but also drops the cell state, achieving reduction of number of parameters. GRU aims to solve the **vanishing gradient problem**. A GRU unit is composed of reset gate and update gate, due to the simpler architecture, it is contributing to train faster and search optimal solution easily [14].

A picture containing text, diagram, screenshot, line

Description automatically generated

Figure . GRU Architecture

In the first step, reset gate is calculated using both the hidden state from the previous time step and the input data at the current time step, it be reserved by applying a sigmoid function σ, as expressed in the equation [14]:

(14)

Where: is input data at the current time step.

is the hidden state from the previous time step.

and are the weighting vectors respectively.

Next, decided the information which will be kept from the previous time steps together with the new inputs. This equation expressed in equation:

(15)

Second, the update gate is computed using the previous hidden state and current input data using the same formula, like the reset gate. But each weight multiplied with the input and hidden state is independent and unique to each gate, which means the final vectors for the update gate are different from the reset gate, as expressed in equation:

(16)

Next, summed with the output, which is from the update gate multiplied by the candidate hidden state, as expressed in equation:

(17)

## XGBoost

XGBoost, developed by Tianqi Chen, is known for it’s scalability, speed, and performance. It implements gradient boosting with several enhancements to improve its effectiveness and efficiency. [15]

Objective: (18)

: training loss measures well model fit on training data.

: Regularization, measures complexity of trees.

To learn tree ensembles, we need additional training (boosting), start from constant prediction, add a new function each time at a result we will have:

(19)

: model at training round t

: new function at round t

The prediction at round t will be [16]:

(2)

# RESULT

Evaluation forecasting model:

**Mean Absolute Error (MAE)**: Average of the absolute error between the actual value and the predicted value

(20)

**Root Mean Squared Error (RMSE):** The square root of the mean square error is calculated using the following formula:

(21)

**Mean Absolute Percentage Error (MAPE):** The measure of prediction accuracy of a forecasting method, defined by the formula:

(22)

In this evaluation, nine distinct models were assessed for the time series examination on 7:2:1, 6:3:1, 8:1:1 training, validating, and testing. RMSE, MAPE and MAE scores were used to assess model execution.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GAS dataset** | | | | |
| **Model** | **Proportion** | **MAPE** | **RMSE** | **MAE** |
| **LINEAR** | *7:2:1* | *27.85%* | *31587.77* | *30502.06* |
| *6:3:1* | *37.77%* | *42824.24* | *40311.92* |
| *8:1:1* | *19.16%* | *22535.56* | *21559.57* |
| **ARIMA** | *7:2:1* | *8.96%* | *11507.11* | *9982.91* |
| *6:3:1* | *19.47%* | *23765.07* | *21166.91* |
| *8:1:1* | *6.31%* | *8384.78* | *6750.34* |
| **RNN** | *7:2:1* | *7.47%* | *10057.47* | *7914.38* |
| *6:3:1* | *11.14%* | *13714.76* | *10849.46* |
| *8:1:1* | *5.05%* | *7276.63* | *5546.81* |
| **LSTM** | *7:2:1* | *7.39%* | *9942.21* | *7818.73* |
| *6:3:1* | *11.81%* | *14520.65* | *11486.71* |
| *8:1:1* | *5.05%* | *7292.85* | *5576.32* |
| **GRU** | *7:2:1* | *7.52%* | *10179.54* | *8008.34* |
| *6:3:1* | *10.96%* | *13542.26* | *10809.97* |
| *8:1:1* | *5.04%* | *7258.21* | *5538.75* |
| **SARIMAX** | *7:2:1* | *1.71%* | *2805.41* | *1872.42* |
| *6:3:1* | *1.63%* | *2614.13* | *1734.98* |
| *8:1:1* | *1.62%* | *2754.71* | *1839.19* |
| **RF** | *7:2:1* | *6.31%* | *8577.42* | *6789.20* |
| *6:3:1* | *9.82%* | *12086.52* | *9695.78* |
| *8:1:1* | *5.32%* | *7564.55* | *5888.84* |
| **DLM** | *7:2:1* | *2.39%* | *3362.73* | *2605.39* |
| *6:3:1* | *7.19%* | *9051.91* | *7849.78* |
| *8:1:1* | *5.61%* | *6720.05* | *6080.53* |
| **XGBOOST** | *7:2:1* | *6.30%* | *8572.36* | *6774.99* |
| *6:3:1* | *9.16%* | *11877.31* | *9575.56* |
| *8:1:1* | *8.84%* | *7348.30* | *5657.84* |

Table . Evaluation of GAS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **HPG dataset** | | | | |
| **Model** | **Proportion** | **MAPE** | **RMSE** | **MAE** |
| **LINEAR** | *7:2:1* | *49.72%* | *12762.53* | *9970.27* |
| *6:3:1* | *32.69%* | *8485.49* | *7626.16* |
| *8:1:1* | *89.31%* | *17465.75* | *16864.04* |
| **ARIMA** | *7:2:1* | *46.92%* | *11854.09* | *9581.04* |
| *6:3:1* | *248.69%* | *69419.31* | *59232.21* |
| *8:1:1* | *25.97%* | *5683.12* | *4418.22* |
| **RNN** | *7:2:1* | *29.23%* | *9769.91* | *7952.87* |
| *6:3:1* | *34.99%* | *11878.86* | *9521.47* |
| *8:1:1* | *31.35%* | *7696.02* | *6878.81* |
| **LSTM** | *7:2:1* | *29.05%* | *9582.45* | *7803.23* |
| *6:3:1* | *34.84%* | *11900.47* | *9535.61* |
| *8:1:1* | *27.84%* | *6858.74* | *6066.57* |
| **GRU** | *7:2:1* | *29.17%* | *9749.41* | *7937.89* |
| *6:3:1* | *34.51%* | *11828.86* | *9484.67* |
| *8:1:1* | *29.10%* | *7167.21* | *6378.28* |
| **SARIMAX** | *7:2:1* | *1.36%* | *536.43* | *344.81* |
| *6:3:1* | *1.29%* | *564.15* | *376.69* |
| *8:1:1* | *1.63%* | *474.39* | *321.51* |
| **RF** | *7:2:1* | *30.15%* | *9148.60* | *7449.74* |
| *6:3:1* | *34.90%* | *10753.57* | *8661.18* |
| *8:1:1* | *22.78%* | *5595.20* | *4710.56* |
| **DLM** | *7:2:1* | *29.17%* | *7337.69* | *6003.24* |
| *6:3:1* | *34.98%* | *9759.23* | *8151.12* |
| *8:1:1* | *15.84%* | *3398.39* | *2741.53* |
| **XGBOOST** | *7:2:1* | *29.53%* | *8817.70* | *7180.44* |
| *6:3:1* | *28.15%* | *100024.99* | *7532.29* |
| *8:1:1* | *19.48%* | *4538.69* | *3865.27* |

Table . Evaluation of HPG

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **VPB dataset** | | | | |
| **Model** | **Proportion** | **MAPE** | **RMSE** | **MAE** |
| **LINEAR** | *7:2:1* | *19.41%* | *4287.41* | *3812.87* |
| *6:3:1* | *32.69%* | *8485.49* | *7626.16* |
| *8:1:1* | *33.06%* | *6264.67* | *5786.23* |
| **ARIMA** | *7:2:1* | *358.77%* | *78651.86* | *68277.79* |
| *6:3:1* | *141.47%* | *32841.37* | *28355.88* |
| *8:1:1* | *11.79%* | *2591.41* | *1978.59* |
| **RNN** | *7:2:1* | *15.71%* | *4076.06* | *3336.71* |
| *6:3:1* | *17.96%* | *4400.72* | *3556.53* |
| *8:1:1* | *13.49%* | *3051.32* | *2600.78* |
| **LSTM** | *7:2:1* | *15.05%* | *3869.98* | *3182.67* |
| *6:3:1* | *18.49%* | *4632.42* | *3762.75* |
| *8:1:1* | *11.69%* | *2610.57* | *2236.11* |
| **GRU** | *7:2:1* | *15.93%* | *4182.56* | *3418.89* |
| *6:3:1* | *16.61%* | *4384.66* | *3511.98* |
| *8:1:1* | *13.75%* | *3111.87* | *2655.57* |
| **SARIMAX** | *7:2:1* | *1.57%* | *470.94* | *319.06* |
| *6:3:1* | *1.49%* | *459.64* | *321.82* |
| *8:1:1* | *1.82%* | *452.42* | *322.93* |
| **RF** | *7:2:1* | *14.30%* | *3276.38* | *2769.42* |
| *6:3:1* | *15.63%* | *3997.27* | *2998.68* |
| *8:1:1* | *9.27%* | *2036.61* | *1581.02* |
| **DLM** | *7:2:1* | *18.05%* | *3898.94* | *3420.63* |
| *6:3:1* | *20.41%* | *4686.82* | *4071.94* |
| *8:1:1* | *7.81%* | *1705.85* | *1314.61* |
| **XGBOOST** | *7:2:1* | *13.61%* | *3160.35* | *2693.37* |
| *6:3:1* | *14.86%* | *3864.79* | *2808.90* |
| *8:1:1* | *9.86%* | *2183.07* | *1671.03* |

Table . Evaluation of VPB

***Predicting Visualization:***

***For GAS dataset:***

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the Random Forest model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the XGBoost model with rate of 7-2-1

A picture containing text, screenshot, font, plot

Description automatically generated

Figure . Predictive results of the RNN model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the GRU model with rate of 8-1-1

A picture containing text, screenshot, plot, font

Description automatically generated

Figure . Predictive results of the LSTM model with rate of 8-1-1

***For HPG dataset:***

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the Random Forest model with rate of 8-1-1

A picture containing screenshot, text, plot, line

Description automatically generated

Figure . Predictive results of the XGBoost model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the RNN model with rate of 7-2-1

A picture containing text, screenshot, diagram, plot

Description automatically generated

Figure . Predictive results of the GRU model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the LSTM model with rate of 7-2-1

***For VPB dataset:***

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the Random Forest model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the XGBoost model with rate of 8-1-1

A picture containing text, screenshot, plot, line

Description automatically generated

Figure . Predictive results of the RNN model with rate of 8-1-1

A picture containing text, screenshot, plot, diagram

Description automatically generated

Figure . Predictive results of the GRU model with rate of 8-1-1

A picture containing text, screenshot, plot, font

Description automatically generated

Figure . Predictive results of the LSTM model with rate of 8-1-1

# Conclusion

In this study, we have employed various learning algorithms and machine learning techniques, accompanied by three different types of train-test splits, to train our models and assess their accuracy. The results reveal that ARIMA, Linear Regression models exhibit relatively high errors because the dataset is not suitable for these models. On the other hand, GRU, SARIMAX and Random Forest outperforms other models in terms of MAPE and RSME measures. However, the result of SARIMAX model is excellent because of the exogenous. Finally, the XGBoost model emerges as the most effective among the mentioned models, displaying the lowest MAPE and RSME values in predictive performance.

Moving forward, our future research endeavors will be dedicated to the development of increasingly refined models, with the overarching objective of attaining heightened precision and accuracy in the prediction of stock prices.

##### Acknowledgment

Our team want to express our deepest gratitude to ***Assoc. Prof. Dr. Nguyen Dinh Thuan, Mr. Nguyen Minh Nhut, Ms. Nguyen Thi Viet Huong*** for their unwavering care and support throughout this course. Your presence has meant the world to our team, and we are truly grateful for your assistance and the sharing of your valuable experience. Your guidance will undoubtedly help us achieve great things with our project.

##### References

[1] D. Satria, “PREDICTING BANKING STOCK PRICES USING RNN, LSTM, AND GRU APPROACH,” *Appl. Comput. Sci.*, vol. 19, pp. 82–94, Mar. 2023, doi: 10.35784/acs-2023-06.

[2] X. Jin and C. Yi, “The Comparison of Stock Price Prediction Based on Linear Regression Model and Machine Learning Scenarios,” presented at the 2022 International Conference on Bigdata Blockchain and Economy Management (ICBBEM 2022), Atlantis Press, Dec. 2022, pp. 837–842. doi: 10.2991/978-94-6463-030-5\_82.

[3] J. Ruan, W. Wu, and J. Luo, “Stock Price Prediction Under Anomalous Circumstances,” in *2020 IEEE International Conference on Big Data (Big Data)*, Atlanta, GA, USA: IEEE, Dec. 2020, pp. 4787–4794. doi: 10.1109/BigData50022.2020.9378030.

[4] M. Tranmer, J. Murphy, M. Elliot, and M. Pampaka, “Multiple Linear Regression (2nd Edition)”.

[5] V. Ş. Ediger and S. Akar, “ARIMA forecasting of primary energy demand by fuel in Turkey,” *Energy Policy*, vol. 35, no. 3, pp. 1701–1708, Mar. 2007, doi: 10.1016/j.enpol.2006.05.009.

[6] B. Dey, B. Roy, S. Datta, and T. S. Ustun, “Forecasting ethanol demand in India to meet future blending targets: A comparison of ARIMA and various regression models,” *Energy Rep.*, vol. 9, pp. 411–418, Mar. 2023, doi: 10.1016/j.egyr.2022.11.038.

[7] S. Kumar, A. Gupta, K. Arora, and K. Vatta, “Effect of Rainfall in Predicting Tomato Prices in India: An Application of SARIMAX and NARX Model,” vol. 32, pp. 159–164, Dec. 2022.

[8] “randomforest.pdf.” Accessed: Jun. 17, 2023. [Online]. Available: https://www.math.mcgill.ca/yyang/resources/doc/randomforest.pdf?fbclid=IwAR1hi5MSrcuWRBq2zcpM79yZzyWgPZKmgoA5U7RVhQ9\_dnre3p6BHlZa1WQ

[9] M. Laine, “Introduction to Dynamic Linear Models for Time Series Analysis,” 2020, pp. 139–156. doi: 10.1007/978-3-030-21718-1\_4.

[10] “dlmodeler.pdf.” Accessed: Jun. 17, 2023. [Online]. Available: http://cran.nexr.com/web/packages/dlmodeler/dlmodeler.pdf

[11] L. Lin, “Stock Prediction and Analysis based on RNN Neural Network,” *SHS Web Conf.*, vol. 151, p. 01002, Dec. 2022, doi: 10.1051/shsconf/202215101002.

[12] J. Qiu, B. Wang, and C. Zhou, “Forecasting stock prices with long-short term memory neural network based on attention mechanism,” *PLOS ONE*, vol. 15, p. e0227222, Jan. 2020, doi: 10.1371/journal.pone.0227222.

[13] G. Ding and L. Qin, “Study on the prediction of stock price based on the associated network model of LSTM,” *Int. J. Mach. Learn. Cybern.*, vol. 11, Jun. 2020, doi: 10.1007/s13042-019-01041-1.

[14] C.-J. Chen, F.-I. Chou, and J.-H. Chou, “Temperature Prediction for Reheating Furnace by Gated Recurrent Unit Approach,” *IEEE Access*, vol. 10, pp. 33362–33369, 2022, doi: 10.1109/ACCESS.2022.3162424.

[15] T. Chen, “XGBoost: A Scalable Tree Boosting System”.

[16] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.