A STUDY ON DATA MINING APPROACHES FOR BANKING SHARE USING SUPERVISED MACHINE LEARNING APPROACHES

M. Vijayakanth[1], V. Veeramanikandan[2]

Article History: **Received:** 16.01.2023 **Revised:** 01.03.2023 **Accepted:** 14.04.2023

Abstract

NIFTY, National Stock Exchange Fifty abbreviation, holds significant prominence within the Indian stock market. It is a pivotal benchmark index mirroring the performance of the top 50 most substantial and highly liquid corporations enlisted on India's National Stock Exchange (NSE). Data mining aims to convert unprocessed data into valuable and practical insights, assisting enterprises, researchers, and analysts in making knowledgeable choices and anticipating forthcoming trends. Supervised machine learning is a branch of artificial intelligence and machine learning where an algorithm learns from a labeled dataset. This approach provides the algorithm with input-output pairs, where the inputs are the features, and the outputs are the corresponding target values. This research considers overall banking share with five parameters processed using various supervised machine learning approaches and its accuracy parameters.

Keywords: Data mining, Share market, Machine learning, Accuracy parameters, and Decision tree

[1]Research Scholar, Dept. of Computer and Information Science, Annamalai University, Annamalainagar – 608 002, Tamil Nadu, India
[2]Assistant Professor, Dept. of Computer Science, Thiru Kolanjiappar Govt. Arts College, Vridhachalam-606 001, Tamil Nadu, India

Email: [1]vijayakanth82@gmail.com, [2]klmvmani@gmail.com

DOI: 10.31838/ecb/2023.12.1.570

1. Introduction

The main objective of supervised learning is to train the algorithm to generalize its learning from the provided labeled examples, so that it can accurately predict the correct output for new, unseen inputs. This enables the algorithm to make predictions or classifications on new data points that it has not encountered during training. Common applications of supervised learning include image classification, sentiment analysis, spam email detection, and stock price prediction. The algorithm learns to map the inputs to the correct outputs by identifying patterns and relationships in the labeled data.

Functioning as an extensively utilized gauge of the comprehensive performance of the Indian stock market, the NIFTY offers valuable perspectives to traders and investors regarding the performance of the leading 50 companies spanning diverse sectors and the overall market conditions. It operates as a point of reference for various financial instruments, encompassing index funds, exchange-traded funds (ETFs), and derivative contracts such as NIFTY futures and options. As a result, investors regularly depend on the NIFTY to assess market patterns, formulate well-informed investment decisions, and craft proficient trading tactics.

The stock market, also referred to as the share market, is a domain where data mining techniques hold considerable potential. Here are a few ways in which data mining is employed within the share market: Predicting Stock Prices: Data mining algorithms have the capacity to analyze historical stock price data alongside various economic indicators, facilitating the prediction of forthcoming stock prices. Techniques like time series analysis and regression help uncover patterns and trends that contribute to accurate price forecasts.

Analyzing Investor Behavior and Market Trends: Data mining contributes to the analysis of investor actions, market trends, and the factors influencing market shifts. This insight serves as a valuable resource for shaping strategic investment choices. In essence, data mining in the share market involves extracting insightful information from extensive and intricate datasets, enabling well-informed investment decisions, risk management, and adaptation to dynamic market conditions.

Review of the Literature

The conventional approach to forecasting stock prices relies on historical data analysis. However, accurate prediction holds immense significance for investors aiming to maximize returns. This research paper is dedicated to exploring various machine learning models for market prediction. Within this study, supervised machine learning algorithms, including Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), and Neural Network (NN), have been employed to predict stock market behavior. These algorithms have been tested using data from the Bombay Stock Exchange (BSE), with accuracy serving as the primary metric to determine the most effective algorithm for improved predictions [1]. The authors discuss that the study proposes a new hybridization of seasonal trend decomposition methods based on Loess (STDL) and Optimal Kernel Extreme Learning Machines (OKELM) for the short- and medium-term prediction of the daily close price of the CRUDE OIL index. ELM parameter tuning is done by using the Gray Wolf Optimization Algorithm (GWO) to improve the predictive performance of the ELM further. The validation of the proposed work is done using two measures of performance, namely MASE and SMAPE [2] and [3]. explains the systematics of machine learning-based stock market prediction approaches based on the use of a generic framework [4]. The hybrid stock prediction model results are calculated using the mean absolute error (MAE) and the RMSE metric. The performance of the hybrid stock prediction model is better than the single prediction model, namely DNN and ANN, with a 5% to 7% improvement in the RMSE score. Indian stock price data are considered for the work [5]. The Random Forests classifier is one of the best classification techniques available capable of accurately classifying large amounts of data. Random Forests is a collection learning method for classification and regression that builds multiple decision trees at training time and outputs the class, which is the mode of the classes output from single trees [6]. A single decision tree is easy to design but typically suffers from high variance, making them uncompetitive in terms of accuracy. One way to overcome this limitation is to generate many variants of a single decision tree, each time choosing a different subset of the same training set in the context of randomization-based ensemble methods [7]. Random Trees (RT) belong to a class of machine learning algorithms that perform ensemble classification. The term ensemble implies a method that makes predictions by averaging over the predictions of multiple independent base models. There are three main decisions to make when creating a random tree. These are (1) the method of splitting the leaves, (2) the type of predictor to use in each leaf, and (3) the method of introducing randomness into the trees [8]. The REPTree is a C4.5 based fast decision tree algorithm that constructs multiple trees and finally chooses the optimal one as representative. Each

decision/regression tree is constructed using information gain/variance and pruned using reduced error pruning with backfitting. Missing values are treated as in C4.5 by breaking the corresponding instances into pieces [9]. Machine Learning strategy that will be taught using publicly released stock data to build information, then using that information to make a valid prediction.For accuracy and prediction of stock Classification and Regression Algorithms are used with Kaggle dataset a machine learning technique comes under supervised learning that are Random Forest, Decision Tree, and Logistic Regression to predict stock prices for the given company previous year data, employing prices with daily trading prices. Python is the coding language used to anticipate the stock market using machine learning. Result come across that Regression model has more accuracy and can predict more accurate stock price [10]. Data from a few manufacturing companies over the last five years (2015–2019), including financial and non-financial data, as well as stock price variance over each year. We used KNIME's machine learning to find the best models for predicting the variance of the stock price in a given year based on the parameters provided. We expect the algorithms to be able to predict which company will return a capital gain for the investor [11]. Stock trend forecasting system with a focus on reducing the amount of sparseness in the data collected using machine learning. We conduct an outlier detection of the data available for reducing dimensionality and implement a K-Nearest Neighbour Algorithm to classify stock trends. The experimental results show the performance and effectiveness of the proposed trend forecasting system compared to the existing systems. The proposed system's model (i.e., KNN classifier) gives better results of low error (MSE = 0.00005, MAE = 0.005 and Logcosh = 0.004) on KSE dataset as compared to previous works [12]. The different techniques that are used in the prediction of share prices from traditional machine learning and deep learning methods to neural networks and graph-based approaches. It draws a detailed analysis of the techniques employed in predicting the stock prices as well as explores the challenges entailed along with the future scope of work in the domain [13]. Supervised ML techniques can be used in forecasting recession and stock market crash (more than 20% drawdown). After learning from strictly past monthly data, ML algorithms detected the Covid-19 recession by December 2019, six months before the official NBER announcement. Moreover, ML algorithms foresaw the March 2020 S&P500 crash two months before it happened. The current labor market and housing are harbingers of a future U.S. recession (in 3 months). Financial factors have a bigger role to play in stock market crashes than economic factors. The labor market appears as a top-two feature in predicting both recessions and crashes [14].

Methodologies and Backgrounds

Gaussian Processes

A Gaussian process is a stochastic process, such that every finite collection of those random variables has a multivariate normal distribution, i.e. every finite linear combination of them is normally distributed. The distribution of a Gaussian process is the joint distribution of all those random variables, and as such, it is a distribution over functions with a continuous domain, e.g. time or space. Gaussian Process (GP) is a powerful supervised machine learning method that is largely used in regression settings. This method is desirable in practice since:

- it performs quite well in small data regime;
- it is highly interpretable;
- it automatically estimates the prediction uncertainty.

This last point is what sets GP apart from many other machine learning techniques: for a GP model, its prediction f(x) at a location x is not a deterministic value, but rather a random variable following a normal distribution,

$$f(x) \sim N(\mu(x), \sigma^2(x)) \quad \ldots (1)$$

Here, $\mu(x)$ denotes the prediction mean and $\sigma^2(x)$ is the prediction variance, which serves as an indicator for the prediction uncertainty.

Linear Regression

Linear regression is a statistical technique employed to comprehend and forecast the connection between two variables by discovering the optimal straight line that most effectively aligns with the data points. It aids in ascertaining how alterations in one variable correspond to changes in another, proving valuable for predictions and trend recognition. The core idea of linear regression is to find the best-fitting straight line (also called the "regression line") through a scatterplot of data points. This line represents a linear equation of the form:

$$y = m_x + b \quad \ldots (2)$$

Where:

- y is the dependent variable (the one you want to predict or explain).
- x is the independent variable (the one you're using to make predictions or explanations).
- m is the slope of the line, representing how much
- y changes for a unit change in x.
- b is the y-intercept, indicating the value of y when x is 0.

SMOreg

The effectiveness of the Sequential Minimal Optimization algorithm (SMO) has been demonstrated in training support vector machines (SVMs) for classifying tasks involving sparse datasets. SMO's ability to operate without a quadratic programming solver sets it apart from many other SVM algorithms. Here are the steps for implementing SMOreg:

Data Collection: Gather the dataset containing input features and corresponding target values for regression.
Kernel Selection: Choose an appropriate kernel function for your regression problem. Common choices include linear, polynomial, and radial basis function (RBF) kernels.
Parameter Selection: Determine the hyperparameters of the SVM, such as the regularization parameter (C) and the kernel-specific parameters.
Initialize Parameters: Initialize the Lagrange multipliers (alphas), bias (b), and error cache for all data points. Alphas represent the importance of each data point in the regression.
Main Loop: Iterate through the training dataset multiple times using the SMO algorithm:
Pair Selection: Select two alphas (alpha_i and alpha_j) based on a heuristic that aims to optimize the convergence speed. These alphas correspond to the data points involved in the optimization step.
Update Alphas: Update the selected alphas using the SMO optimization rules while considering the box constraints and the equality constraint imposed by the sum of alpha products equaling zero.
Update Error Cache: Update the error cache for all data points to reflect the changes in alphas.
Check Convergence: After each iteration, check for convergence by monitoring the change in alphas. If the change is smaller than a predefined tolerance, or if a maximum number of iterations is reached, the algorithm terminates.
Calculate Bias: Once the alphas are optimized, calculate the bias (b) using the support vectors.
Predict: To make predictions on new data points, calculate the regression output using the optimized alphas, the kernel function, and the bias.
Evaluate: Evaluate the performance of the trained SMOreg model using regression metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), or R-squared.
Fine-tuning and Validation: If necessary, perform cross-validation or other validation techniques to fine-tune hyperparameters and ensure the model's generalization to unseen data.

Random Tree

In machine learning, a Random Tree is a specific type of decision tree variant that introduces randomness during construction. Random Trees are similar to traditional decision trees but differ in how they select the splitting features and thresholds at each node. The primary goal of introducing randomness is to create a more diverse set of decision trees, which can help reduce overfitting and improve the model's generalization performance. Random Trees are commonly used as building blocks in ensemble methods like Random Forests. The critical characteristics of Random Trees are as follows:

- Random Feature Subset
- Random Threshold Selection
- No Pruning
- Ensemble Methods

Steps involved in Random Tree

Step 1. Data Bootstrapping:
Step 2. Random Subset Selection for Features:
Step 3. Decision Tree Construction:
Step 4. Voting (Classification) or Averaging (Regression):

Random Forest

Random Forest is an ensemble learning method combining multiple decision trees to make more accurate and robust predictions for classification and regression tasks. The steps involved in building a Random Forest are as follows:

Step 1. Data Bootstrapping
Step 2. Random Feature Subset Selection
Step 3. Decision Tree Construction
Step 4. Ensemble of Decision Trees
Step 5. Out-of-Bag (OOB) Evaluation
Step 6. Hyperparameter Tuning (optional)

REP Tree

REP Tree (Repeated Incremental Pruning to Produce an Error Reduction Tree) is a machine learning algorithm for classification and regression tasks. It is an extension of decision trees that incorporates pruning to reduce overfitting and improve the model's generalization performance. Below are the steps involved in building a REP Tree.

Step 1. Recursive Binary Splitting
Step 2. Pruning
Step 3. Repeated Pruning and Error Reduction
Step 4. Model Evaluation

Correlation coefficient or R2 score

Calculating the correlation coefficient between two variables involves several steps. Let's assume we have two datasets: X and Y, each containing n data points. Here are the steps to calculate the correlation coefficient:

Step 1. Compute the means of X and Y
Step 2. Calculate the deviations from the norm for both X and
Step 3. Compute the product of the variations for each data point

Step 4. Sum up the effects of deviations
Step 5. Calculate the square of the variations for both X and Y
Step 6. Sum up the squared deviations for X and Y
Step 7. Compute the correlation coefficient (r). The correlation coefficient (r) is calculated using the formula:

$$r = \Sigma ((X - \bar{x}) (Y - \bar{y})) / \sqrt{(\Sigma (X - \bar{x})^2 * \Sigma(Y - \bar{y})^2)} \quad \text{... (3)}$$

Step 8. Interpret the correlation coefficient. The resulting value of "r" will be between -1 and 1, indicating the strength and direction of the linear relationship between X and Y:

- r ≈ +1: A strong positive correlation (as X increases, Y increases).
- r ≈ -1: A strong negative correlation (as X increases, Y decreases).
- r ≈ 0: Little to no linear correlation (no consistent relationship between X and Y).

Mean Absolute Error

Mean Absolute Error (MAE) is a metric used to measure the average absolute difference between predicted and actual (true) values in a regression problem. It is commonly used to assess the accuracy of a regression model's predictions [14]. The formula to calculate Mean Absolute Error (MAE) is as follows:

$$MAE = \Sigma |(\text{Actual Value} - \text{Predicted Value})| / n \quad \text{... (4)}$$

Where:
Σ represents the summation symbol, which sums up the values for all data points. | | denotes the absolute value, ensuring the differences are positive. Actual Value: Refers to the true value of the target variable (ground truth) for a specific data point. Predicted Value: Refers to the value predicted by the regression model for the same data point, and n: Represents the total number of data points in the dataset.

Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) is a commonly used metric to assess the accuracy of a regression model's predictions. It measures the average magnitude of the errors between the predicted and actual (true) values, considering both the direction and magnitude of the errors. The formula to calculate Root Mean Squared Error (RMSE) is as follows [15]:

$$RMSE = \sqrt{(\Sigma (\text{Actual Value} - \text{Predicted Value})^2 / n)} \quad \text{... (5)}$$

Where:

- Σ represents the summation symbol, which sums up the values for all data points.
- (Actual Value - Predicted Value) ² denotes the squared difference between the actual and predicted values for each data point.
- n is the total number of data points in the dataset.

Relative Absolute Error (RAE)

Relative Absolute Error (RAE), also known as Mean Absolute Percentage Error (MAPE), is a metric used to evaluate the accuracy of predictions in regression tasks. It measures the average percentage difference between the absolute and actual (valid) values, providing a relative measure of the prediction errors [16]. The formula to calculate Relative Absolute Error (RAE) is as follows:

$$RAE = (\Sigma |\text{Actual Value} - \text{Predicted Value}| / \Sigma |\text{Actual Value}|) * (100 / n) \quad \text{... (6)}$$

Where:
Σ represents the summation symbol, which sums up the values for all data points. | | denotes the absolute value, ensuring the differences are positive. n is the total number of data points in the dataset. In this formula, actual value refers to the true value of the target variable (ground truth) for a specific data point. Predicted value refers to the value predicted by the regression model for the same data point.

Root Relative Squared Error (RRSE)

Root Relative Squared Error is not a standard or commonly used metric in statistics or machine learning. Root Mean Squared Error (RMSE) and Relative Absolute Error (RAE) are widely used to evaluate the accuracy of regression models. These are well-known metrics with clear interpretations:

Step 1. Root Mean Squared Error (RMSE):

- Collect the actual (true) values and the corresponding predicted values from a regression model.
- For each data point, calculate the squared difference between the actual and predicted values: (Actual Value - Predicted Value) ².
- Sum up all the squared errors to get the total sum of squared errors.
- Divide the total sum of squared errors by the total number of data points (n) to find the Mean Squared Error (MSE).
- Take the square root of the Mean Squared Error (MSE) to get the Root Mean Squared Error (RMSE).

Step 2. Relative Absolute Error (RAE) or Mean Absolute Percentage Error (MAPE):

- Collect the actual (true) values and the corresponding predicted values from a regression model.
- For each data point, calculate the absolute difference between the actual and predicted values: |Actual Value - Predicted Value|.

- For each data point, calculate the absolute percentage difference between the absolute error and the actual value: |Actual Value - Predicted Value| / |Actual Value|.
- Sum up all the absolute percentage errors to get the total sum of absolute percentage errors.

Numerical Illustrations

Dataset

Datasets are the cornerstone for diverse data-centric activities, including machine learning, statistical analysis, and research. They furnish the essential building blocks for training machine learning models, performing experiments, and extracting insights by revealing patterns, trends, and relationships inherent in the data. Table 1 shows the dataset [15] has five parameters related to the share market likely date, open, high, low, and close.

Table 1: Raw Data Related to Banking Share

Date	Open	High	Low	Close
09-Sep-22	40520.75	40685.95	40280.30	40415.70
08-Sep-22	39763.90	40265.75	39706.40	40208.95
07-Sep-22	39337.75	39572.05	39258.25	39455.90
06-Sep-22	39892.95	40073.75	39564.30	39666.50
05-Sep-22	39412.05	39865.20	39407.40	39805.75
02-Sep-22	39422.30	39595.85	39200.45	39421.00
01-Sep-22	38806.70	39667.65	38803.30	39301.25
30-Aug-22	38516.95	39606.55	38472.70	39536.75
29-Aug-22	38111.60	38397.10	37943.85	38276.70
26-Aug-22	39129.65	39337.10	38846.80	38987.15
25-Aug-22	39190.05	39471.75	38803.70	38950.75
24-Aug-22	38552.70	39120.80	38552.20	39038.50
23-Aug-22	37955.45	38869.90	37950.85	38697.65
22-Aug-22	38693.65	38732.85	38247.25	38297.75
19-Aug-22	39732.65	39759.15	38848.40	38985.95
18-Aug-22	39324.40	39703.70	39291.15	39656.15
17-Aug-22	39351.30	39504.50	39202.80	39461.70
16-Aug-22	39284.10	39444.60	39119.90	39239.65
12-Aug-22	38942.45	39088.90	38739.95	39042.30
11-Aug-22	38712.95	38932.10	38648.90	38879.85
10-Aug-22	38298.85	38402.95	38155.30	38287.85
08-Aug-22	37847.35	38302.35	37681.45	38237.40
05-Aug-22	37868.25	38150.45	37779.90	37920.60
04-Aug-22	38111.05	38231.85	37249.50	37755.55
03-Aug-22	37954.55	38068.60	37692.95	37989.25
02-Aug-22	37767.70	38179.85	37632.30	38024.00
01-Aug-22	37594.15	37939.60	37407.20	37903.20

Table 2: R2 Score or Correlation Coefficient

ML Approaches	Open	High	Low	Close
Gaussian Processes	0.9999	0.9996	0.9996	0.8060
Linear Regression	1.0000	1.0000	1.0000	0.8065
SMOreg	1.0000	1.0000	0.9999	0.8067
Random Tree	0.9995	0.9996	0.9996	0.8064
Random Forest	0.9998	0.9998	0.9998	0.8067
REP Tree	0.9995	0.9996	0.9996	0.8070

Table 3: Mean Absolute Error (MAE)

ML Approaches	Open	High	Low	Close
Gaussian Processes	6989.4418	7027.6709	6941.2421	8199.8235
Linear Regression	63.7497	54.8014	60.1724	5428.6012
SMOreg	65.5571	56.7108	62.1476	3620.3632
Random Tree	233.8301	226.9687	228.7738	5427.9104
Random Forest	148.1727	125.9153	135.2051	5427.9104
REP Tree	233.4790	229.7322	229.2373	5406.3476

Table 4: Root Mean Squared Error (RMSE)

ML Approaches	Open	High	Low	Close
Gaussian Processes	7914.7862	7961.2759	7853.7106	9613.3376
Linear Regression	98.2507	88.0559	98.8331	6350.5496
SMOreg	99.1339	91.6780	103.9512	7268.0291
Random Tree	299.9111	293.3892	293.4593	6688.3359
Random Forest	197.9304	176.3000	187.8957	6688.3359
REP Tree	299.5313	296.4773	294.4187	6677.7544

Table 5: Relative Absolute Error (RAE)

ML Approaches	Open	High	Low	Close
Gaussian Processes	82.0512	82.0249	82.1092	91.1318
Linear Regression	0.7484	0.6396	0.7118	60.3328
SMOreg	0.7696	0.6619	0.7352	40.2362
Random Tree	2.7450	2.6491	2.7062	60.3251
Random Forest	1.7394	1.4696	1.5994	60.3251
REP Tree	2.7409	2.6814	2.7117	60.0854

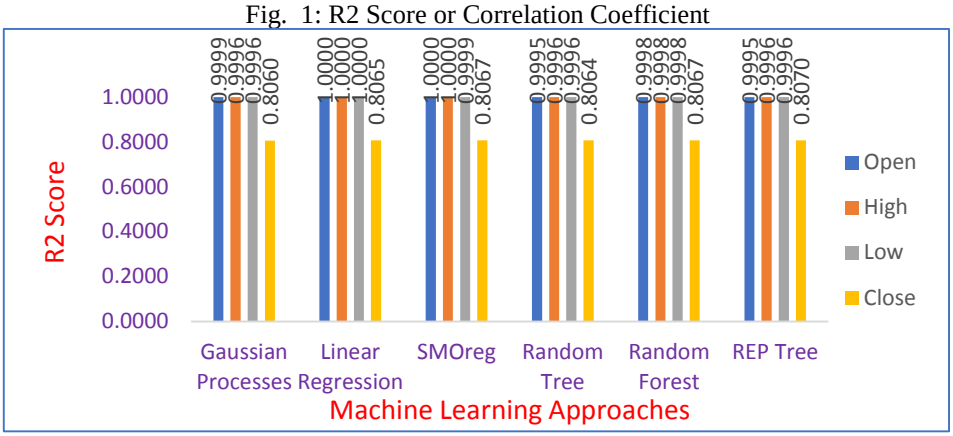
Table 6: Root Relative Squared Error (RRSE)

ML Approaches	Open	High	Low	Close
Gaussian Processes	79.4524	79.4423	79.4349	89.4847
Linear Regression	0.9863	0.8787	0.9996	59.1134
SMOreg	0.9952	0.9148	1.0514	67.6537
Random Tree	3.0106	2.9276	2.9681	62.2576
Random Forest	1.9869	1.7592	1.9004	62.2576
REP Tree	3.0068	2.9584	2.9778	62.1592

Table 7: Time taken to build the models in Seconds

ML Approaches	Open	High	Low	Close
Gaussian Processes	101.0000	121.4300	120.2300	239.5300
Linear Regression	0.0400	0.0100	0.0100	0.0100
SMOreg	16.4000	18.6500	32.5600	65.2000
Random Tree	0.0100	0.0100	0.0100	0.7200
Random Forest	0.5100	0.2400	0.2400	35.1600
REP Tree	0.0400	0.0100	0.0100	0.3200

Fig. 1: R2 Score or Correlation Coefficient



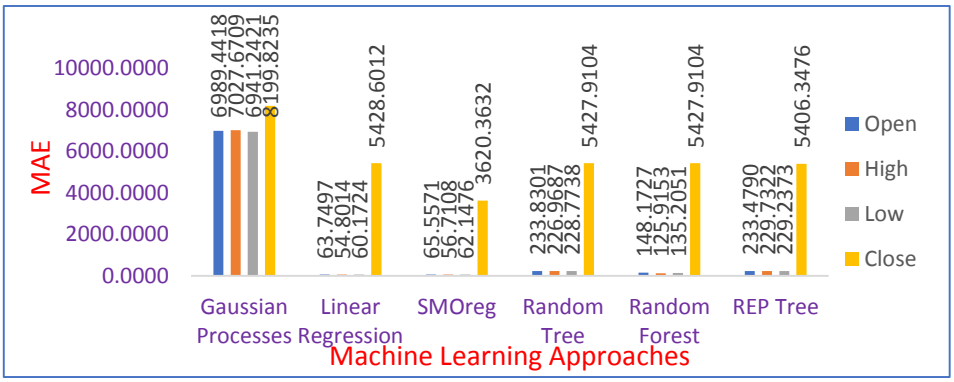


Fig. 2: Mean Absolute Error (MAE)

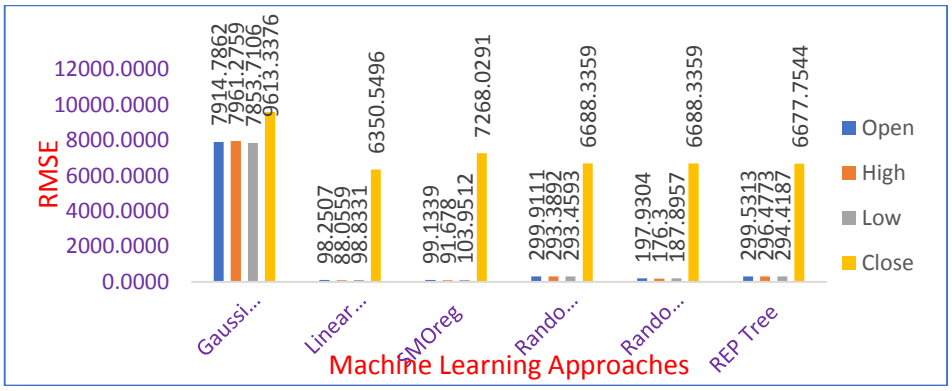


Fig. 3: Root Mean Squared Error (RMSE)

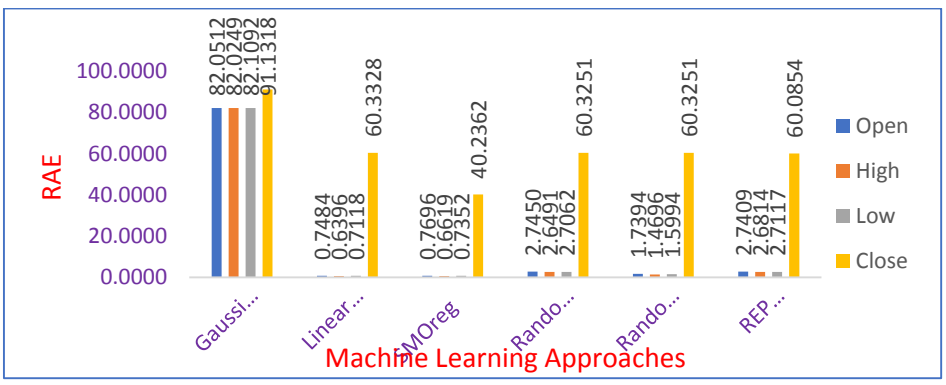


Fig. 4: Relative Absolute Error (RAE)

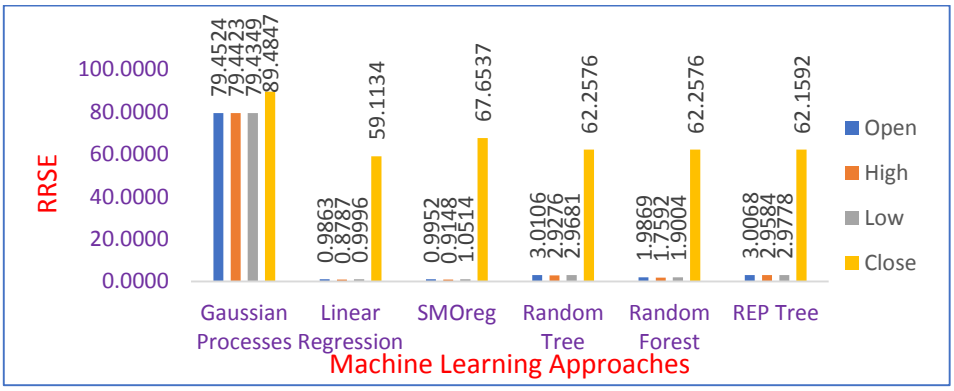


Fig. 5: Root Relative Squared Error (RRSE)

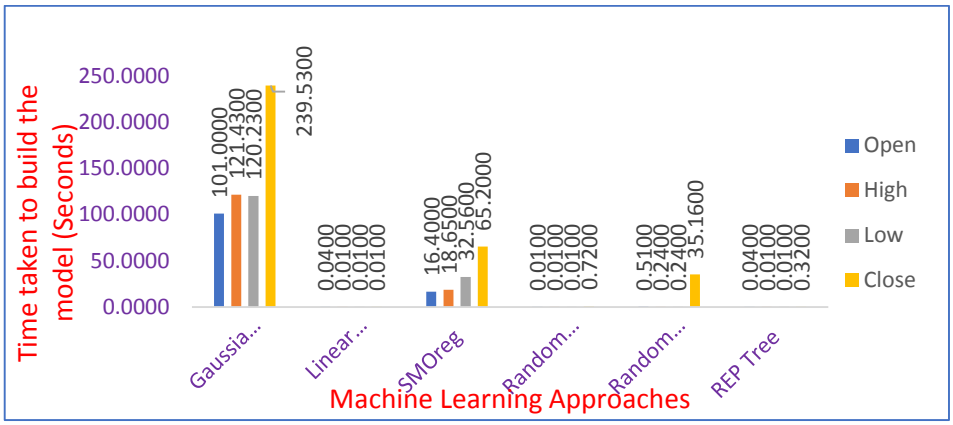


Fig. 6: Time taken to build the models in Seconds

2. Result and Discussion

Table 1 indicates the raw dataset, which includes five parameters: date, open, high, low, and close. The dataset consists of 5645 instances with five parameters. Based on the dataset, it is evident that two different machine learning functions, namely Gaussian processes, and linear regression, and four different tree approaches, namely SMOreg, Random Tree, Random Forest, and REP Tree, are used to find the hidden patterns, and which is the best or influencing parameter to decide future predictions. Related results and numerical illustrations are shown in Table 1 to Table 7 and Figure 1 to Figure 6. They are based on Equation 3, used to find the R2 score or correlation coefficient by comparing four parameters with their relationships. Numerical illustrations suggest that there may be a significant difference from one parameter to another. In this case, using six different ML approaches among these results, all three parameters return a robust positive correlation of nearly 1 when using open, high, and low. Another parameter, namely closed, produces a positive correlation of 0.80. The related numerical illustrations are shown in Table 2 and Figure 1. Further data analysis using Table 3 and Figure 2 revealed improved test scores for using different accuracy parameters over time. The MAE is used to find the model error using Equation 4. In this case, six machine learning algorithms will be used meanwhile, except the Gaussian process, other ML approaches return minimum error compared to other parameters using open, high, and low while using test statistics for using close parameters return high amount of error for using all six ML algorithms. Similarly, the RMSE test statistics return the same results as MAE. RMSE measures the difference between predicted and actual values using Equation 5. The related numerical illustration is shown in Table 4 and Figure 3. Relative Absolute Error (RAE) measures accuracy using Equation 6 to compare the difference between predicted and actual values. Similarly, the RRSE also measures the error and returns the results in percentage using Equation 7. In these cases, the RAE and RRSE return very minimum errors of 0 to 3 for using Linear Regression, SMOreg, Random Tree, Random Forest, and REP Tree. In another case, using the Gaussian process return maximum error for using open, high, low, and close. The RAE-related numerical illustrations are shown in Table 5 and Figure 4. RRSE approaches and related numerical examples are shown in Table 6 and Figure 5. Based on Table 7 and Figure 6 indicate the time taken to build the model in all the ML algorithms. In this case, the Gaussian process takes more time to build the model, and the remaining

five ML approaches consider the minimum amount of time to build the model. Using open, high, and low parameters takes less time to complete the job except for close parameters.

Conclusion and further research

It is essential to consider this study's limitations in the banking share field. The sample size of each group was relatively small, which could impact the generalizability of the results. The banking sector data analysis and predictions, the findings presented in this study contribute to our understanding that all the parameters return strong positive correlations with very satisfactory accuracy except close parameters. Future studies can build upon these, finding future predictions for close parameter with accuracy using other ML approaches.

3. Reference

1. Rath, S., Gupta, B. K., & Nayak, A. K. (2022). Stock Market Prediction Using Supervised Machine Learning Algorithm. In Advances in Distributed Computing and Machine Learning: Proceedings of ICADCML 2021 (pp. 374-381). Springer Singapore.
2. Veeramanikandan, V., and Jeyakarthic. M., (2020). Hybridization Of Stdl with Optimal Kernel Extreme Learning Machine (Okelm) Based Short Term Crude Oil Price Forecasting In Commodity Futures Market. International Journal of Scientific & Technology Research, 9(2), 4029- 4036.
3. Jeyakarthic. M., and Veeramanikandan, V., (2020). Forecasting of commodity future index a hybrid regression model based on support vector machine and grey wolf optimization algorithm. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 9(2), 2856-2862.
4. Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S., & Kim, H. C. (2021). Stock market prediction using machine learning techniques: a decade survey on methodologies, recent developments, and future directions. Electronics, 10(21), 2717.
5. Manujakshi, B. C., Kabadi, M. G., & Naik, N. (2022). A Hybrid Stock Price Prediction Model Based on PRE and Deep Neural Network. Data, 7(5), 51.
6. Rajesh, P., & Karthikeyan, M. (2017). A comparative study of data mining algorithms for decision tree approaches using WEKA tool. Advances in Natural and Applied Sciences, 11(9), 230-243.
7. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.
8. Denil, M., Matheson, D., & De Freitas, N. (2014, January). Narrowing the gap: Random forests in theory and in practice. In International conference on machine learning (pp. 665-673). PMLR.
9. Koulinas, G., Paraschos, P., & Koulouriotis, D. (2021). A machine learning-based framework for data mining and optimization of a production system. Procedia Manufacturing, 55, 431-438.
10. Yelne, A., & Theng, D. (2021, November). Stock Prediction and analysis Using Supervised Machine Learning Algorithms. In 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA) (pp. 1-6). IEEE.
11. Anggraeni, D., Sugiyanto, K., Zam, M. I. Z., & Patria, H. (2022). Stock price movement prediction using supervised machine learning algorithm: KNIME. Jurnal Akun Nabelo: Jurnal Akuntansi Netral, Akuntabel, Objektif, 4(2), 671-681.
12. Khattak, A., Khan, A., Ullah, H., Asghar, M. U., Arif, A., Kundi, F. M., & Asghar, M. Z. (2022). An efficient supervised machine learning technique for forecasting stock market trends. Information and Knowledge in Internet of Things, 143-162.
13. Soni, P., Tewari, Y., & Krishnan, D. (2022). Machine Learning approaches in stock price prediction: A systematic review. In Journal of Physics: Conference Series (Vol. 2161, No. 1, p. 012065). IOP Publishing.
14. Malladi, R. K. (2022). Application of Supervised Machine Learning Techniques to Forecast the COVID-19 US Recession and Stock Market Crash. Computational Economics, 1-25.
15. https://www.kaggle.com/datasets/debashis74017/stock-market-index-data-india-1990-2022?select=NIFTY+BANK_Data.csv