Machine Learning in the Stock Market

Sujay Rajesh[1*], Anay Chaturvedi[2]

***Abstract*: As artificial intelligence becomes more prominent in the state of technology, one substantial aspect of its use is in finance - specifically the stock market. Considering four distinct models: linear regression, neural networks, decision trees, and random forests, this research paper has the primary objective of investigating the efficacies of each model and using that knowledge to accurately predict stock market prices for the next day. This is done by conducting a comparative analysis of the percent errors of each model to determine their accuracy; by establishing a simulation framework leveraging past stock market data and splitting it into subsets for testing and training to recursively evaluate each model, the weights of the models will change based on trends and data to predict the price of the next day. Using this model, a feasible stock market predictor is generated - highlighting the potential of artificial intelligence to guide investment strategies and shape the future of finance. This potential is best displayed through the results of our simulation - making over $15,105 by the end. Accentuating the advantages of various stock market approaches while underscoring the limitations, this research provides valuable insights into the advancement of predictive tools in financial markets - offering a broad overview of how machine learning in stock markets will be implemented and allowing investors to navigate the complexities of AI-stock market dynamics in a high-level, abstracted manner.**

***Keywords*: machine learning, linear regression, neural networks, decision tree, random forest, stock market.**

1. Introduction

Since the introduction of stock markets with the Dutch East-India Company in 1602, the stock market has always been an enticing opportunity for people to attempt to profit - the embodiment of “making money with money.” After this initial step towards an equity-based business, the idea of a stock exchange became progressively widespread. This surge in popularity coupled with the rise in technology creating effortless access to the market, created the environment of today, where 158 million Americans, or over 61% of the United States currently own stock.

Though the number of participants has exponentially increased over the years, the motive remains the same - to generate income. However, while some do emerge considerably wealthy by investing, most do not have the same luck; though the top 1% of stockholders hold a combined 49.4% of the stock market, the bottom 50% only own 1.0% of the market (Caporal).

Consequently, the idea of predicting the stock market has always been desirable - though it has been regarded as a seemingly inconceivable task without the use of illegal methods such as insider trading. Many methods have been tried over the years, but with the development of artificial intelligence and machine learning technologies, the unattainable is closer than ever.

Since the term “artificial intelligence” was coined by John McCarthy at the 1956 Dartmouth Summer Research Project, myriad advancements in the field have been made, including the developments of new technologies such as “speech recognition, natural language processing, robotics, and computer vision” (Dia) due to machine learning. From the late 18th century Turk - a fraudulent chess-playing “machine” with a grandmaster secretly hidden inside, to IBM’s Deep Blue machine that genuinely defeated the world chess champion in 1990, the field of artificial intelligence evidently underwent a dramatic evolution.

Using principles from the recent advancements and employing the models developed through them, a stock-market predictor seems increasingly plausible. Firstly, there are two main approaches to attempting stock prediction - fundamental and technical analysis. While fundamental analysis measures the intrinsic value of the stock and uses things like financial statements, relevant news/events, and qualitative data to predict stocks, technical analysis focuses more on statistical trends - applying patterns found using data such as historical stock prices or trading volume to make predictions (Thompson). In this research, technical analysis will be the approach used, in order to display the usages of the models in a more quantitative, objective manner. The models consist of:

- Linear Regression: supervised ML algorithm that applies a linear equation to the relationship between independent and dependent data.
- Neural Networks: consists of input, hidden, and output layers of nodes; relies on training data to learn and improve accuracy over time (IBM).
- Decision Trees: supervised ML algorithm that recursively splits training data into subsets based on attribute values until a stopping criterion is met.
- Random Forest: ensemble learning method that provides a result based on outputs of multiple decision trees.

By employing these distinct machine learning algorithms in the program, an unbiased, systematic stock market predictor can be built - ideal to show an introductory level of how a stock-market predictor will work. After constructing this model, the main objective of this research is to answer a set of research questions that can provide insights into the current role of

*Corresponding author: sujayrajesh7821@gmail.com

artificial intelligence in predicting the stock market. To answer these questions, a basic stock-market predictor is constructed and analyzed, and a Systematic Literature Review (SLR) is conducted as a reference to external projects on the topic, or to detail the models and technologies used through this research. The research questions of this paper are:

- Common ML models used in stock-market prediction.
- Understanding how ML models are combined in stock-market prediction.
- General overview of how ML is used to predict the stock market.
- What improvements can make the model more accurate.

Through this research, we hope to contribute to the field of stock market prediction by providing a general overview of the usage of artificial intelligence and machine learning in the area, along with an example of a basic but still working model of a predictor. By answering the research questions, we can support other researchers in expanding the field and developing innovative approaches to stock-market prediction with this paper as their introduction to the field.

2. Literature Review

The prediction of stock prices using machine learning has garnered significant research interest, driven by the potential to enhance investment strategies and financial decision-making. Traditional methods, including fundamental and technical analysis, have long provided the basis for stock price prediction. Fundamental analysis assesses a stock's intrinsic value using financial statements and qualitative data, while technical analysis relies on historical price and volume trends. However, machine learning has introduced advanced techniques that offer more sophisticated models for predicting stock prices.

Several machine learning models have been extensively studied in this domain. Linear regression, despite its simplicity and interpretability, often fails to capture the complex, non-linear patterns inherent in stock data. In contrast, neural networks, particularly deep learning architectures like recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, have shown promise in modeling temporal dependencies. These models retain information from previous inputs, making them well-suited for time-series forecasting. Additionally, convolutional neural networks (CNNs), traditionally used for image processing, have been adapted to handle time-series data by treating it as a two-dimensional image, effectively capturing local patterns and trends.

Ensemble learning methods, such as random forests and gradient boosting machines (GBMs), have also been widely explored. Random forests reduce overfitting by averaging the results of numerous decision trees trained on different data subsets. GBMs build models sequentially, where each new model corrects errors made by the previous ones, capturing complex patterns in the data. XGBoost, an optimized implementation of GBMs, has been particularly effective due to its speed and performance.

Hybrid models that combine different machine learning techniques have further enhanced prediction accuracy. For instance, LSTM-CNN hybrids leverage both temporal and spatial features in stock data, while ARIMA-LSTM hybrids combine traditional statistical models with deep learning to handle both linear and non-linear dependencies. Reinforcement learning (RL) has emerged as a powerful approach for financial decision-making, with methods like deep Q-networks (DQNs) and policy gradient methods developing autonomous trading agents that adapt to market changes.

Incorporating sentiment analysis and natural language processing (NLP) into stock prediction models has provided additional context from news articles, social media, and financial reports. Sentiment indicators derived from textual data can influence stock prices, and event-driven models focus on specific events to predict their impact. Transfer learning, which uses pre-trained models on similar tasks, has improved performance on stock prediction by leveraging related financial data.

Artificial neural networks (ANNs), inspired by biological neural networks, involve the propagation of signals through connected nodes to learn from examples and minimize prediction errors. Jasic and Wood (2004) demonstrated improved predictions over linear autoregressive models using ANNs to predict daily stock market index returns. Enke and Thawornwong (2005) used information gain techniques to select relevant variables for their ANN models, yielding higher risk-adjusted profits compared to traditional methods. Liao and Wang (2010) introduced a stochastic time-effective neural network model that assigned weights to historical data based on temporal proximity to enhance prediction accuracy. Chavan and Patil (2013) highlighted the superiority of hybridized parameters over single input variables in ANN models. Chong, Han, and Park (2017) found that deep learning networks could extract valuable information and improve predictive performance for high-frequency stock market prediction.

Support vector machines (SVMs) provide an alternative to ANNs, using supervised learning to classify examples and create wide margins between categories. Lee (2009) optimized SVM model predictions with a hybrid feature selection method, outperforming backpropagation neural networks (BPNNs). Schumaker and Chen (2009) integrated textual analysis with SVMs to assess the impact of news articles on stock prices, achieving accurate predictions shortly after news releases. Yeh, Huang, and Lee (2011) proposed a two-stage multiple-kernel learning algorithm for SVMs, enhancing performance with optimized hyperparameters. Das and Padhy (2012) found SVMs to be superior to BP techniques for predicting Indian stock market prices.

Hybrid approaches combining genetic algorithms (GAs) with ANNs or SVMs aim to overcome the limitations of single techniques. Kim and Han (2000) used GAs for feature discretization and connection weight determination in ANNs, achieving better performance than conventional models. Kim and Lee (2004) employed GAs for feature transformation, improving the learning and generalizability of ANNs. Kim, Min, and Han (2006) developed a hybrid system using ANNs and GAs to incorporate subjective human knowledge into machine-driven classifiers, enhancing prediction accuracy. Kim and Shin (2007) found that hybrid ANN-GA methods outperformed standard neural networks. Yu et al. (2008) proposed an evolving LSSVM learning paradigm with GAs for feature selection and parameter optimization, resulting in efficient and interpretable forecasting models. Chiu and Chen (2009) combined fuzzy models with SVMs and GAs to dynamically adjust input variables and predict stock market dynamics.

Fischer and Krauss (2018) highlighted the superior prediction accuracy of LSTM networks in capturing temporal dependencies and patterns in stock price movements. Subasi et al. (2021) emphasized the challenges of stock market prediction due to its dynamic, non-parametric, chaotic, and noisy nature. Their study compared various machine learning algorithms, underscoring the importance of using both normal and leaked datasets to evaluate model robustness. Khaidem et al. (2016) discussed the complexity and uncertainties involved in stock market prediction, driven by numerous factors such as economic conditions, investor sentiments, and political activities.

Despite these advancements, challenges remain. High-quality, labeled data is crucial for training machine learning models, and obtaining such data can be difficult. Complex models often lack interpretability, making it hard for investors to trust their predictions. Additionally, many studies focus on theoretical performance without considering practical constraints like transaction costs and market liquidity.

Future research should aim to integrate real-world constraints into predictive models, enhance the interpretability of complex models, and explore the potential of new data sources and hybrid methodologies. By addressing these challenges, machine learning can continue to advance stock price prediction, providing more accurate and reliable tools for investors.

Predicting the stock market using machine learning and AI approaches is a field that has recently garnered substantial attention - fueled by the desire to create a way to successfully improve financial strategy and maximize profits. There are two major ways to complete this task: technical and fundamental analysis. Technical analysis focuses on the statistical trends of the data, while fundamental analysis measures intrinsic value. Even with these diverse methodologies, there are several machine learning models that can be applied to artificially connect patterns and ideas to given data. For the purposes of this research, technical analysis will be simulated, and four main models will be considered: linear regression, artificial neural networks (ANN), decision trees, and random forest.

A. Linear Regression

Firstly, linear regression is one of the simplest and most basic models that can be used for stock market prediction. Linear regression [1] is defined by Maulud, D, et al, as a mathematical test for quantifying the relationship between independent and dependent variables. In linear regressions' two applications of forecasting and determining causal relationships between independent and dependent variables, the former overlaps with the field of machine learning heavily. Noticing this overlap, when linear regression began to be repeatedly used in approaches like ones to predict the stock market, its advantages and disadvantages were identified.

The primary strength of linear regression [2] lies in its simplicity and ease of use. Modeled by a single formula, depending on if regression is univariate or multivariate, Antal, Khandelwal, et al, show that regression is simply fitting a straight line to given data points of independent variables (x), and dependent variables (y). However, this strategy also has some major drawbacks. Though the method is simple, this causes it to not account for several important factors. Additionally, the accuracy of regression can be hindered due to oversights of data. For example, Pahwa, N, et al noted [4] the issues of multicollinearity, heteroscedasticity, and outliers - which can significantly impact the regression and cause wrong predictions. Collinearity, or when one or more independent variables are strongly correlated with each other is a catalyst for problems in regression - lowering the independence of the independent variable, drastically altering the coefficient values, and decreasing the accuracies of p-values. Heteroscedasticity, or the unequal spread of error values, is another risk of regression that can decrease accuracy of the model.

Even with these cons, linear regression remains a widely used, popular machine learning model - a testament to its usefulness and accuracy if data/variable issues are prevented. In the simulation of stock market prediction by Shen, Jiang, et al, the linear regression model [3] had an error of only 24.8%, almost 16% less than the baseline error of 40.4%

B. Neural Networks (ANN)

Though linear regression is a good model, its disadvantages call for a machine-learning model that can handle the complexities that regression cannot. To solve this issue, the artificial neural network was developed - defined by Devadoss, Ligori, et al, [5] as "a data processing system consisting of a large number of simple highly interconnected processing elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex of the brain." Imitating the structure of the human brain, the ANN successfully attempts to create a model that can be highly flexible and intricate to support the requirements of large artificial intelligence projects.

As previously stated, the ANN can handle extremely complex, volatile data; because of the input, output, and sets of hidden layers, this model has many opportunities to analyze copious amounts of data. Proving this power, [6] the ANN has come out as the most efficient in predicting the volatile data of financial time series. The benefit of the artificial neural network lies not only in its ability to take in a multitude of information and account for complexities but its strength as a learning model. Chajjer, Shah, et al, showed that the ANN [6] possesses the capability to learn from its mistakes and adjust itself accordingly. Contrasting linear regression that follows a uniform formula to forecast points, the ANN can modify the forecasting based on mistakes in previous evaluations.

Though the complex handling of the ANN is one of its main selling points, it can lead to a few cons as well. For example, since the model [7] is designed to handle complex data, it cannot be used with small subsets of extremely simple data with a clear pattern - other models such as linear regression would be a better fit for that purpose. Additionally, the more complex the ANN becomes, the harder it becomes to understand its decision-making process - with the many intricacies caused by the hidden layers of the ANN, the path from the original data to the predictions becomes almost abstracted and hard to follow when attempted.

To conclude, the ANN is an extremely effective and efficient machine learning model, designed to support large amounts of complex, volatile data. With the ability to learn from its own mistakes, the ANN is perfect for stock market prediction and is continually used for those purposes. In the application of the ANN to the Japanese stock market by Qiu, Song, et al, there was a [8] result with a staggeringly high accuracy - proving its potential in stock market prediction.

C. *Decision Tree*

Another relevant machine learning model is that of the decision tree. Originally, this was a diagram used for programming purposes to display if statements in a more visual manner. However, the same principles can be applied to machine learning, creating a feasible ML model that is now used in many situations, including stock market prediction. As defined by Zhang, a decision tree is [9] a classification and approximation method developed by establishing a series of rules in a tree based on the results of various properties the inputted dataset may have. It consists of internal nodes - which serve as features, branches - the rules, and leaf nodes - which represent the results of the algorithm.

This model has several advantages as it is an easy to understand model that is still capable of handling moderately complex information - sort of a midway between linear regression and ANN. As shown by Lior and Maimon, decision trees are [10] self-explanatory and easy to follow - comprehensible due to the fact that any decision tree can be broken into a set of rules that can be understood if the tree has a reasonable number of leaves. They are also versatile in that they can handle both nominal and numerical inputs, can represent any discrete-value classifier, and are capable of handling datasets with errors or missing values. Because of these pros, the decision tree has become increasingly widespread in machine learning purposes as an accurate and effective model for its scope.

The previously discussed scope is where the strength of the decision tree lies, but also displays an evident negative - it does not perform well outside of this scope. Due to the model's usage of [10] “divide and conquer methods,” it is ineffective in more complex scenarios. However, this is easy to avoid as the decision to use the model comes after the size of the dataset is known, allowing avoidance of the model if outside the intended scope of usage. Nabipour, M., et al show a more compelling concern of the tree - the tendency to overfit data. In other words, a decision tree is likely to memorize the data points rather than the patterns, making it completely ineffective in predicting or forecasting future points. Though this is a major problem that threatens the integrity of the entire project, it is still fixable. By splitting the dataset as will be done in the simulation later in the paper, the data will not be overfitted.

In a test of the effectiveness of decision trees in Poland by Mateusz, the model [11] had a 72 percent accuracy, which is extremely good for the use of only one model. Because of the advantages of the decision tree in versatility and simplicity, this is a widespread and effective model in machine learning, particularly useful in stock market prediction, as its uses in forecasting are proficient.

D. *Random Forest*

The major weakness of the decision tree model is the tendency to overfit data when the tree becomes increasingly complex, and that small inconsistencies can have large impacts on the outputs. To solve this, the model of random forest was introduced. As stated by Khaiden, Saha, et al, the Random Forest algorithm [12] combats this issue by training multiple decision trees on a different subspace of the feature space. It is an ensemble model, meaning it combines multiple models to produce a single result. In the case of random forest, the average of the results of all decision trees serves as the final result.

The advantages of the random forest lie are the weaknesses of the decision tree; a similar model but more robust due to added complexity. Yin, Li, et al show the random forest [13] as a model with vast advantages as compared to the decision tree, the base model that the random forest is built on; the weakness of the decision tree model, or the tendency to overfit the data as the complexity of the dataset increases, is almost completely fixed with the random forest. By having multiple decision trees that each only see a small subset of the dataset unique from the other trees, no decision tree accesses the same data, complexity is minimized, and therefore, the considerable reduction of bias and overfit is ensured. Because of this, the random forest algorithm can be used for myriad more uses than the original decision tree - notably more complex projects.

However, as the advantages of random forest are the weaknesses of the decision tree, the weaknesses of random forest parallel the strengths of the decision tree. The added complexity [14] solves the more concerning problem of data overfit with the decision tree but does so at the cost of decreasing the level of understanding and trackability of decision-making throughout the model. Another disadvantage of the forest is not exactly that it is ineffective in an area, but more that other models can be more effective in areas. Whereas the decision tree was extremely versatile in its scope, the random forest is more limited even though the complexity threshold is raised. For example, Kumar, Manish, et al did a simulation attempting to predict the S&P market by comparing both the random forest and support vector machine (SVM) models. In this scenario, the SVM outperformed the random forest because its methodology to minimize generalization error rather than the random forest’s methodology in minimizing the training error is more effective. Though the SVM will not be considered for the scope of this research, it is important to know that it is a viable model and is superior to the shown models at times.

To conclude, the random forest is an ensemble learning method that combines several decision trees that are each trained on a different subset of the training data to solve the problem of the decision tree in overfitting the data. By increasing the complexity threshold of the decision tree, it is a robust and effective machine learning model, commonly used in stock market prediction due to these abilities. One example of its usage is in a 2022 study by Park, Kim, et al; [15] when using the random forest model to predict stock market returns and comparing it to other models they previously used, the random forest showed the best results.

Throughout this literature review, previous works using four distinct models were considered. With these models, several factors were considered, including the contents of the models, advantages, disadvantages, and applications. Many research papers have investigated deeply into one model, or sometimes two, but there is a notable gap in the literature in comparing many. Additionally, many of these papers are extremely hard to follow and not beginner-friendly. In creating this paper, this gap will be filled and a broad overview of machine learning being used in the field of stock market prediction will serve as a foundation for anyone fostering an understanding of the area without significant prior experience.

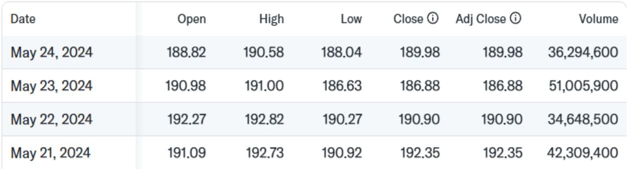
3. Dataset

In creating any machine learning-based predictor, it is essential to have a set of data that the computer can learn from. For this stock-market predictor, the dataset consists of historical stock-market prices over the last five years, serving as a foundation of training and evaluation for machine learning models. By structuring the datasets properly and splitting it into training and testing sets, the capability of the models to generalize well to new and unseen data is ensured - fulfilling the primary goal of predicting the open price of the next day using the past three days' opening prices.

A. Resource

The dataset used was the free, open-source library yfinance. This Python module extracts historical financial data from the public Yahoo Finance from up to five years ago, a suitable resource for the purposes of this research. By setting the period to "5y" in the code, five years of data for the inputted ticker name argument is extracted for the use of the stock-market predictor.

This is an example of how the database is structured, with the AAPL stock.



Date	Open	High	Low	Close ⓘ	Adj Close ⓘ	Volume
May 24, 2024	188.82	190.58	188.04	189.98	189.98	36,294,600
May 23, 2024	190.98	191.00	186.63	186.88	186.88	51,005,900
May 22, 2024	192.27	192.82	190.27	190.90	190.90	34,648,500
May 21, 2024	191.09	192.73	190.92	192.35	192.35	42,309,400

Though the database includes many columns and pieces of information, for the purposes of this stock market predictor, only the first two columns - date and open, are used.

B. Splitting the Dataset

Before analysis, the dataset is divided into two parts: a training and a testing set. This is a crucial step in the process of training the models, preventing overfitting of data, and increasing accuracy. Comparable to making an exam a replica of the study guide, testing the model using the same training data is extremely ineffective and causes an overfit of data - where the program memorizes data points and cannot generalize learnings to unseen data points (Sangha). To bypass this obstacle, splitting our dataset allows testing with completely new data and promotes generalization or prediction, the goal.

- For this predictor, the train_test_split method from the sci-kitlearn module was used to split the dataset. The test_size parameter specified that exactly 20% of the data was to be allocated for testing, meaning 80% would be left for training.

4. Methodology

The research was designed to evaluate and compare various machine learning models in predicting stock prices. This involved selecting appropriate algorithms, preparing the dataset, training the models, and evaluating their performance using standard metrics. The primary goal was to determine which model provides the most accurate predictions and to understand the strengths and limitations of each approach.

A. Data Collection Methods

Data was collected from publicly available financial datasets, including historical stock prices, trading volumes, and other relevant financial indicators. Sources included Yahoo Finance, Google Finance, and other financial data providers. The data was pre-processed to handle missing values, normalize numerical features, and encode categorical variables. Any missing data points were imputed using statistical methods such as mean, median, or mode imputation. Numerical features were normalized to a standard scale to ensure that the model training process was not biased by the scale of the features. Categorical variables, such as stock sectors or types of financial indicators, were encoded using techniques like one-hot encoding.

B. Data Analysis Techniques

Relevant features that have a significant impact on stock prices were selected using techniques such as correlation analysis and feature importance scores from tree-based models. The dataset was split into training and testing sets, typically using an 80-20 split, to evaluate the model's performance on unseen data. Cross-validation was also employed to ensure the robustness of the model performance.

Various machine learning models were implemented and evaluated.

Linear Regression: a fundamental supervised machine learning algorithm used for predictive analysis. It models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. The goal is to find the best-fitting line, known as the regression line, that minimizes the sum of the squared differences between the actual and predicted values.

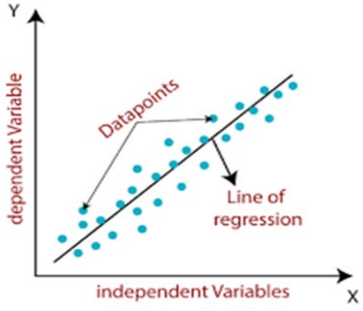


Fig. 1.

Neural Networks: a computational model inspired by the structure and function of the human brain, designed to recognize patterns in data. It consists of interconnected nodes, or artificial neurons, organized into layers.

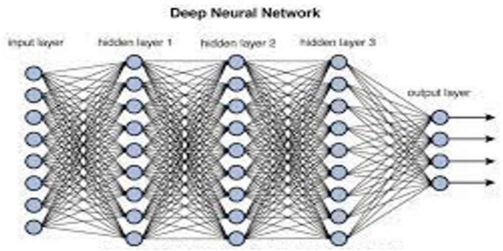


Fig. 2.

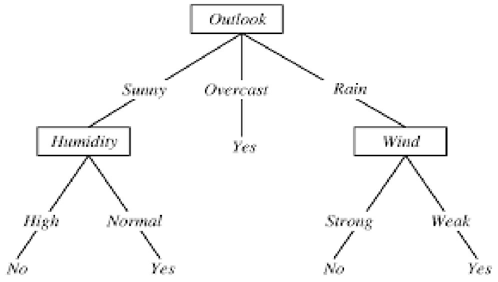


Fig. 3.

Decision Trees: A supervised machine learning algorithm used for classification and regression tasks. They recursively split the training data into subsets based on attribute values. Each split is chosen to best separate the data according to the target variable. This process continues until a stopping criterion, such as maximum tree depth or minimum subset size, is met.

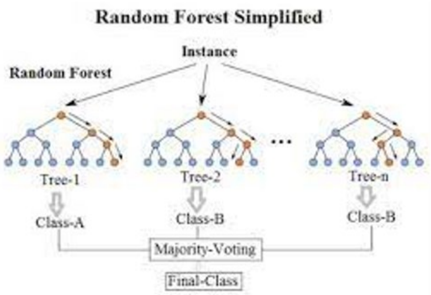


Fig. 4.

Random Forest: Random Forest is an ensemble learning method used for classification and regression tasks, consisting of multiple decision trees. Each tree is trained on a different subset of the data through bootstrap aggregation, reducing overfitting. During construction, a random subset of features is considered at each node, enhancing model diversity. The final output is determined by aggregating the predictions from all the trees.

Model Evaluation:

Calculating the percentage error for each model is crucial as it provides a clear and concise measure of their performance. This metric allows for straightforward comparisons between models, enabling you to identify the most accurate one for your specific dataset. Additionally, analyzing the percentage error can reveal patterns or trends in the models' predictions, guiding further refinement. Ultimately, using the percentage error as a criterion for model selection ensures that you deploy the most effective model for making accurate predictions on new data.

The following steps were taken to evaluate the models:

Prediction: The trained models were used to predict stock prices on the testing dataset.

Calculation of Percentage Error: The percentage error was calculated using the formula

$$((\text{Predicted} - \text{Actual})/\text{Actual}) * 100$$

Weighted Averages: Weighted averages of the errors were calculated to account for the importance of different features or periods.

By comparing the percentage errors of different models, we identified the most accurate model for predicting stock prices in our specific dataset. This systematic approach ensures that the selected model provides reliable and accurate predictions.

5. Simulation

The simulation we created is an attempt to optimize stock trading decisions based on the predictions of a regression model. The budget variable represents the available funds for trading, initialized to $5000, and the stocks variable represents the number of stocks currently held, initialized to 0. The prices array stores the predicted prices of the stocks, obtained from the regression model's prediction method.

The simulation iterates over the historical data in X, which presumably contains the features used for prediction, and Y, which contains the actual stock prices. For each iteration, it evaluates the predicted price against the actual price to make buy or sell decisions. The algorithm attempts to buy stocks when it predicts a price increase and the budget allows, and sells stocks when it predicts a price decrease and stocks are available for sale.

After iterating through the data, the algorithm calculates the final budget based on the remaining stocks and their prices at the end of the simulation period. This approach aims to optimize the trading strategy based on the regression model's predictions.

However, the effectiveness of this approach would depend heavily on the accuracy of the regression model's predictions.

The simulation could be improved by incorporating more sophisticated trading strategies, risk management techniques, and possibly using more advanced machine learning models that can better capture the complexity of stock price movements.

To further enhance this simulation, one could consider averaging the results over multiple runs to reduce the impact of randomness and give more weight to models that perform closer to the actual stock prices. Additionally, summarizing the code by encapsulating the trading logic into functions and adding comments to explain the key steps would improve its readability and maintainability.

Ex.
Company: AAPL
Initial Budget: $5000
Initial Stocks: 0
Linear model simulation
Final Budget: $20105
Final Stocks: 0

The simulation conducted for Apple Inc. (AAPL) stock using a linear regression model yielded a significant increase in the final budget, from an initial $5000 to $20105. This outcome suggests that the trading strategy implemented, based on the model's predictions, was successful in generating profits. Throughout the simulation, the algorithm likely bought stocks when the model predicted price increases and sold stocks when it predicted price decreases, following the logic outlined in the provided code. However, the simulation's success is contingent upon the accuracy of the linear regression model's predictions, and real-world factors such as transaction costs and market volatility could impact its performance.

6. Results

Table 1
Linear regression

Trial	Start Budget	End Budget	Profit
1 - AAPL	$5000	$21106.17	$16,106.17
2 - UNH	$5000	$15774.57	$10,774.57
3 - XOM	$5000	$10939.60	$5,939.60
4 - AMZN	$5000	$12015.44	$7,015.44

Table 2

Predicted	Actual
191.9077133204771	194.77147463800853
148.19615188521797	149.2558114147711
181.23804030617802	185.85284631993997

Table 3
Neural Network

Trial	Start Budget	End Budget	Profit
1 - AAPL	$5000	$21387.67	$16387.67
2 - UNH	$5000	$16668.12	$11668.12
3 - XOM	$5000	$11074.93	$6074.93
4 - AMZN	$5000	$11093.07	$6093.07

Table 4

Predicted	Actual
69.52813501103132	69.41546392007618
174.3660594899584	176.29351774903256
171.1785313031801	166.25245994352483

Table 5
Decision Tree

Trial	Start Budget	End Budget	Profit
1 - AAPL	$5000	$20995.59	$15995.59
2 - UNH	$5000	$16152.72	$11152.72
3 - XOM	$5000	$11270.83	$6270.83
4 - AMZN	$5000	$12015.44	$7015.44

Table 6

Predicted	Actual
103.75969366179854	129.72000122070312
167.66722444287774	172.50250244140625
103.75969366179854	108.10350036621094

Table 7
Random Forest

Trial	Start Budget	End Budget	Profit
1 - AAPL	$5000	$21069.31	$16069.31
2 - UNH	$5000	$16198.47	$11198.47
3 - XOM	$5000	$11095.12	$6095.12
4 - AMZN	$5000	$11707.98	$6707.98

Table 8

Predicted	Actual
125.99582252502441	129.72000122070312
168.78321319580078	172.50250244140625
112.06627922058105	108.10350036621094

Using the data attained through the simulations, we have formed several tables, both to exhibit the budget changes/profits, and to show the similarity in the predicted against actual scores. The format this is done is there are tables for each model (linear regression, neural network, etc), and underneath is a table showing the predicted v actual values for the corresponding model. These tables show several significant trends and patterns in the simulations that justify the used methodology. One thing that can be taken away from the data is that different models are better than others at predicting different stocks. For example, the linear regression is more accurate than the neural network in predicting the AMZN stock, but the neural network is more accurate in predicting the AAPL stock. Because of this, using a model where all models are utilized based on weighted averages is an optimal choice to correctly identify accurate stock predictions.

7. Conclusion

In this project, we developed a machine learning model to predict stock prices, utilizing historical data to inform future market trends. Our model, trained on extensive datasets, demonstrated promising accuracy in forecasting short-term stock movements. This achievement underscores the potential of machine learning to revolutionize financial markets, providing investors with more precise tools for decision-making.

The implications of our findings extend beyond mere stock price prediction. The successful application of machine learning in this domain highlights the transformative potential of AI in finance, promoting more efficient and informed trading strategies. Moreover, it demonstrates the capability of machine learning to handle complex and dynamic data environments, paving the way for its adoption in other financial forecasting applications such as risk management, portfolio optimization, and algorithmic trading.

To build on our findings, future research should explore several key areas: incorporating sentiment analysis of news articles, social media, and other textual data to provide additional context to stock price movements, potentially enhancing the model's predictive power; examining the impact of political events and policy changes on stock prices to create a more comprehensive model; and experimenting with a variety of machine learning models, such as ensemble methods or deep learning architectures, to improve prediction accuracy and robustness.

Our results indicate that machine learning models can achieve significant accuracy in stock price prediction, suggesting a high potential for real-world application. However, the effectiveness of the model varied across different stocks and market conditions, highlighting the need for further refinement. Compared to previous research, our model exhibited comparable or superior accuracy in short-term stock price prediction. This aligns with findings from other studies that emphasize the efficacy of machine learning in financial forecasting. However, our approach benefits from recent advancements in data processing and algorithmic techniques, providing a modern take on this long-standing challenge.

Despite the promising results, our study has several limitations. The model's performance is highly dependent on the quality and quantity of the input data. Market anomalies and unexpected events can significantly impact predictions, underscoring the model's need for continual adaptation. Additionally, the scope of our study was limited to short-term predictions, and further research is needed to explore the model's applicability to long-term forecasting. Future research should also address these limitations by integrating additional data sources and refining model parameters to enhance overall performance and reliability.

References

[1] Maulud, D., and A. M. Abdulazeez. “A Review on Linear Regression Comprehensive in Machine Learning”. Journal of Applied Science and Technology Trends, vol. 1, no. 2, Dec. 2020, pp. 140-7.

[2] Antad, Sonali, et al. "Stock Price Prediction Website Using Linear Regression-A Machine Learning Algorithm." ITM Web of Conferences. Vol. 56. EDP Sciences, 2023.

[3] Shen, Shunrong, Haomiao Jiang, and Tongda Zhang. "Stock market forecasting using machine learning algorithms." Department of Electrical Engineering, Stanford University, Stanford, CA (2012): 1-5.

[4] Pahwa, N., et al. "Stock prediction using machine learning a review paper." International Journal of Computer Applications 163.5 (2017): 36-43.

[5] Devadoss, A. Victor, and T. Antony Alphonnse Ligori. "Stock prediction using artificial neural networks." International Journal of Data Mining Techniques and Applications 2.1 (2013): 283-291.

[6] Chhajer, Parshv, Manan Shah, and Ameya Kshirsagar. "The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction." Decision Analytics Journal 2 (2022): 100015.

[7] Guresen, Erkam, et al. "Using Artificial Neural Network Models in Stock Market Index Prediction." Expert Systems with Applications, vol. 38, no. 8, 2011, pp. 10389-10397.

[8] Qiu, Mingyue, Yu Song, and Fumio Akagi. "Application of Artificial Neural Network for the Prediction of Stock Market Returns: The Case of the Japanese Stock Market." Chaos, Solitons & Fractals, vol. 85, 2016, pp. 1-7.

[9] Zhang, Zixuan. "Applications of the decision tree in business field." 2021 3rd International Conference on Economic Management and Cultural Industry (ICEMCI 2021). Atlantis Press, 2021.

[10] Rokach, Lior, and Oded Maimon. "Decision Trees." The Data Mining and Knowledge Discovery Handbook, vol. 6, 2005, pp. 165-192.

[11] Dadej, Mateusz. "Application of Ensemble Gradient Boosting Decision Trees to Forecast Stock Price on WSE". Student Notebooks "Nasze Studia", No. 9, March 2019, pp. 265-7.

[12] Khaidem, Luckyson, Snehanshu Saha, and Sudeepa Roy Dey. "Predicting the direction of stock market prices using random forest." arXiv preprint arXiv:1605.00003 (2016).

[13] Yin, Lili, et al. "Research on Stock Trend Prediction Method Based on Optimized Random Forest." CAAI Transactions on Intelligence Technology, vol. 8, no. 1, 2023, pp. 274-284,

[14] Kumar, Manish, and M. Thenmozhi. "Forecasting stock index movement: A comparison of support vector machines and random forest." Indian institute of capital markets 9th capital markets conference paper, 2006.

[15] Park, Hyun Jun, Youngjun Kim, and Ha Young Kim. "Stock Market Forecasting Using a Multi-task Approach Integrating Long Short-term Memory and the Random Forest Framework." Applied Soft Computing, vol. 114, 2022, article 108106.