Predicting Stocks: various methods for enhanced prediction and profit

Project submitted for the partial fulfillment of the requirements for the course

CSE 336L: Machine Learning Lab

Offered by the

Department Computer Science and Engineering

School of Engineering and Sciences

Submitted by

1)Sai Naren Polavarapu , Roll No:AP21110010306

2)Rakshita Chadalawada,Roll No:AP21110010297

3) Charishma Ankisetty, Roll No:AP21110010290

SRM University–AP

Neerukonda, Mangalagiri, Guntur Andhra Pradesh – 522 240

[May, 2024]

Contents: page no

Introduction ……………………………………………………………3

Background ………………………………………………………….3

Procedure………………………………………………………………5

Results & Discussion:..11

Conclusion…………………………………………………13

References…………………………………………….14

1.Introduction

This project aims to develop enhanced predictive models for stock movement using machine learning techniques. The objective is to predict whether the stock index will move up or down based on historical data, including OHLC (Open, High, Low, Close) and Volume data from Yahoo Finance, along with macroeconomic features from FRED (Federal Reserve Economic Data).

Stock movement prediction is crucial for investors and traders to make informed decisions in dynamic financial markets. Traditional methods often lack accuracy and reliability, necessitating the adoption of advanced machine learning algorithms. By leveraging historical data and integrating both micro-level market data and macroeconomic indicators, this project seeks to provide actionable insights into stock market behavior.

Through rigorous experimentation with various machine learning techniques such as Linear Regression, Decision Tree, Naive Bayes, and k-Nearest Neighbors (kNN), this project aims to identify the most effective models for stock movement prediction. The outcomes of this research have the potential to improve investment strategies and risk management practices in financial markets.

Background

Predicting stock index movement is crucial for investors and traders, guiding portfolio management and risk mitigation strategies. Traditional statistical models and modern machine learning algorithms address this challenge. Statistical models analyze historical data and trends but may lack adaptability. Conversely, machine learning, leveraging vast datasets, offers a data-driven approach to uncover nuanced market patterns.

Various machine learning algorithms, including regression models, decision trees, and neural networks, are applied in stock market prediction. Despite advancements, prediction remains complex due to multifaceted factors like economic indicators, geopolitical events, and investor sentiment.

Nevertheless, researchers persist in refining prediction models. By integrating diverse datasets and leveraging machine learning advancements, they aim to develop more accurate models, empowering investors with actionable insights.

Data set:

The dataset used in this project comprises OHLC (Open, High, Low, Close) and Volume data sourced from Yahoo Finance, along with macroeconomic features obtained from FRED (Federal Reserve Economic Data). Here's a brief overview of the dataset:

- Label: Binary indicator (0 or 1) representing whether the stock index movement for the following day is lower (0) or higher (1) than the current day's close price.
- Features:
  - OHLC data: Open, High, Low, Close prices of the stock index.
  - Volume: Volume of shares traded for the stock index.
  - Macroeconomic features: Economic indicators sourced from FRED, which may include variables such as interest rates, exchange rates, volatility index (VIX), gold prices, oil prices, TED spread, and effective federal funds rate (EFFR).
- Frequency: Daily data, spanning from 2008.4 to 2018.3.
- Versions of Features:
  - Version 1: Macroeconomic Features
  - Version 2: OHLC + Volume + Macroeconomic Features
  - Version 3: Addition of DJIA (Dow30) and Nasdaq stock index data

This dataset provides a comprehensive view of the stock market, incorporating both market-specific indicators (OHLC and Volume) and broader economic factors. It allows for the exploration of correlations and patterns between various features and the stock index movement, enabling the development and evaluation of predictive models.

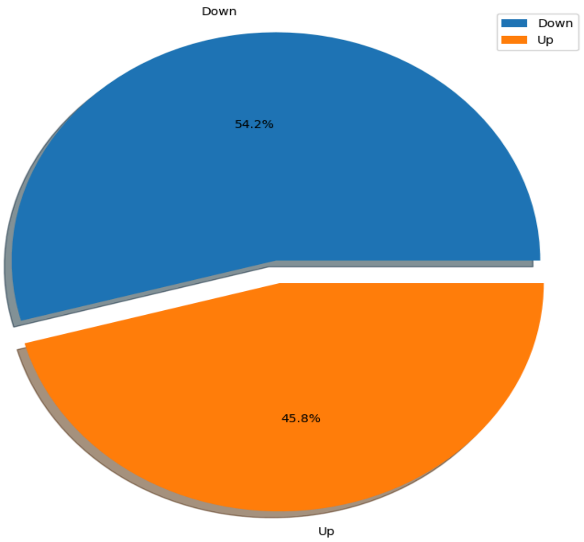


Fig1:stock dataset labels

2.Procedure

1. Load Dataset:
    - The dataset "DOW30.csv" is loaded into a pandas DataFrame using the `read_csv()` function.
    - The first few rows of the dataset are displayed to gain an initial understanding of the data structure and contents.
2. Data Exploration:
    - The shape of the dataset is checked to determine the number of rows and columns.
    - Information about the dataset, such as data types and non-null counts, is displayed to identify any missing values or inconsistencies.
    - Missing values in the dataset are identified by summing the null values for each column.
3. Data Preprocessing:
    - Feature Scaling: Standardization of features is performed to ensure that all features have a mean of 0 and a standard deviation of 1. This prevents features with larger scales from dominating the learning process.
    - Dimensionality Reduction: Principal Component Analysis (PCA) is applied to reduce the dimensionality of the dataset. PCA transforms the original features into a set of linearly uncorrelated variables called principal components. The number of components is specified as 3 (`n_components=3`), and the explained variance ratio of the principal components is computed to understand the amount of variance captured by each component visualization in fig 3.

    - Updating Dataset: The original features in the dataset are replaced with the principal components obtained from PCA, resulting in a new dataset with reduced dimensionality.

4. Correlation Analysis:
    - The correlation matrix of the dataset is computed to quantify the relationships between pairs of features.
    - A heatmap of the correlation matrix(fig:2) is created to visualize the correlations between features, aiding in feature selection and understanding feature importance.

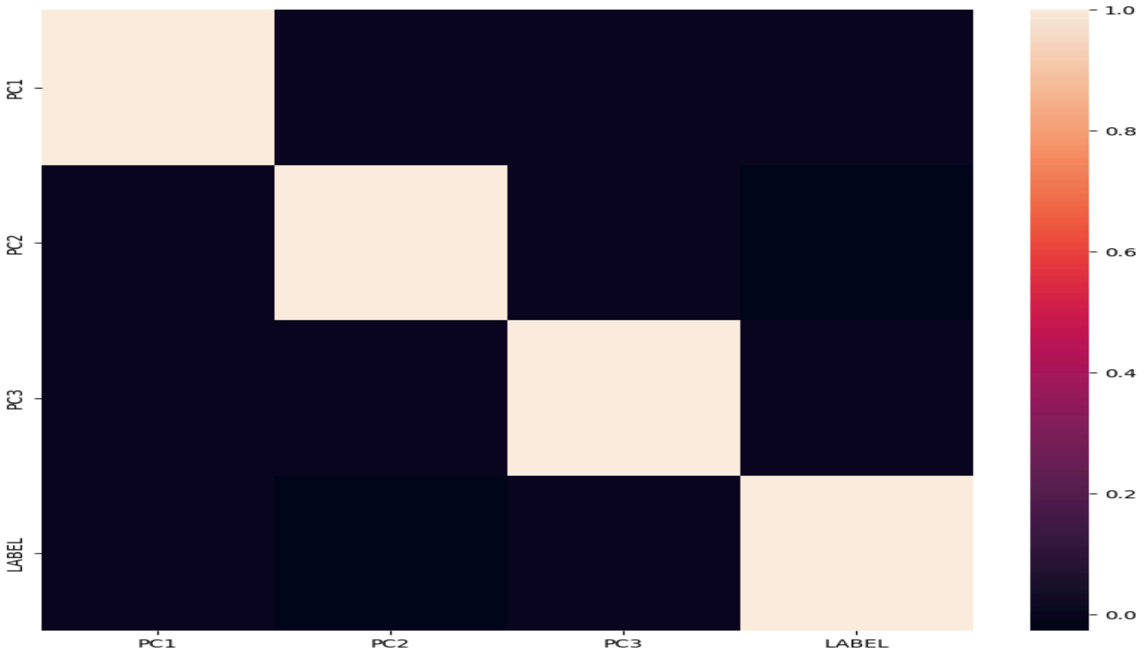
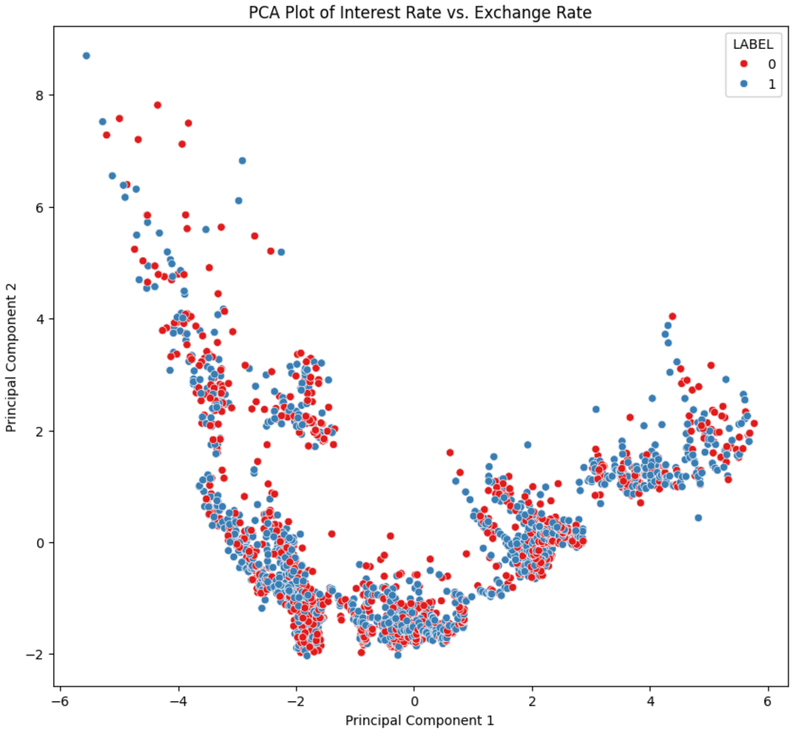


Fig 2:A heatmap of the correlation matrix



-

Fig 3:PCA plot of interest rate vs.Exchange rate

Machine Learning Models:

1)Decision tree:A decision tree is an intuitive and transparent algorithm that efficiently learns decision rules from data, offering interpretability and ease of understanding for classification and regression tasks.

1. Trained decision tree classifier on dataset to predict stock index movements.
2. Split data into training and testing sets, using entropy criterion for training.
3. Achieved approximately 48.98% accuracy on the test set.
4. Classification report included precision, recall, and F1-score for each class.
5. Confusion matrix provided visualization of the classifier's performance, showing true positives, false positives, true negatives, and false negatives.

Output:

```
Train Result:
===============================================
Accuracy Score: 48.98%
_______________________________________________
CLASSIFICATION REPORT:
              precision    recall  f1-score   support

           0       0.47      0.47      0.47       236
           1       0.51      0.51      0.51       254

    accuracy                           0.49       490
   macro avg       0.49      0.49      0.49       490
weighted avg       0.49      0.49      0.49       490

_______________________________________________
Confusion Matrix:
 [[110 126]
 [124 130]]
```

2)Naive Bayes: Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem, with the "naive" assumption that features are conditionally independent given the class label, making it computationally efficient and easy to implement for text classification and other applications.

1. Extracted features (X) from the dataset excluding the 'LABEL' column and assigned labels (y) to the 'LABEL' column.
2. Split the data into training and testing sets with a 20% test size and random state of 0.
3. Created a Gaussian Naive Bayes classifier.
4. Trained the classifier on the training data.
5. Made predictions on the test set.
6. Accuracy achieved: 51.22%.

Output:

```
Train Result:
================================================
Accuracy Score: 51.22%
_______________________________________________
CLASSIFICATION REPORT:
              precision    recall  f1-score   support

           0       0.47      0.11      0.18       236
           1       0.52      0.88      0.65       254

    accuracy                           0.51       490
   macro avg       0.50      0.50      0.42       490
weighted avg       0.50      0.51      0.43       490

_______________________________________________
Confusion Matrix:
 [[ 27 209]
 [ 30 224]]
```

2)Linear Regression: Linear regression is a statistical method used to model 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 (or hyperplane in higher dimensions) that minimizes the sum of squared differences between the observed and predicted values. It's commonly used for predicting continuous outcomes and understanding the relationship between variables in various fields, including economics, finance, and social sciences.

1. Data was divided into two parts: one for training the model and the other for testing its performance.
2. A linear regression model was established to understand how features relate to labels.
3. The model was taught using the training data to recognize patterns in features and their associated labels.
4. Predictions were made on the training data using the trained linear regression model.
5. Evaluation metrics were computed for the model's performance on the training data:
    a. Mean Squared Error: 0.25
    b. R-squared: 0.18%
6. These metrics help assess how well the model fits the training data and explains the variance in the target variable.

Output:

```
Train Result:

================================================

Mean Squared Error: 0.25

R-squared: 0.18%
```

- Predictions were made on the testing data using the trained linear regression model.
- Evaluation metrics were computed for the model's performance on the testing data:
    - Mean Squared Error: 0.25
    - R-squared: -0.86%
- These metrics help assess how well the model generalizes to new, unseen data and explains the variance in the target variable.

Output:

```
Test Result:

================================================

Mean Squared Error: 0.25

R-squared: -0.86%

_______________________________________________
```

- A DataFrame was created to compare actual and predicted values fig.4
- The DataFrame named `new_dataset` contains two columns: 'Actual' and 'Predicted'.
- This DataFrame allows for easy visualization and comparison of model predictions with the actual values from the testing data.

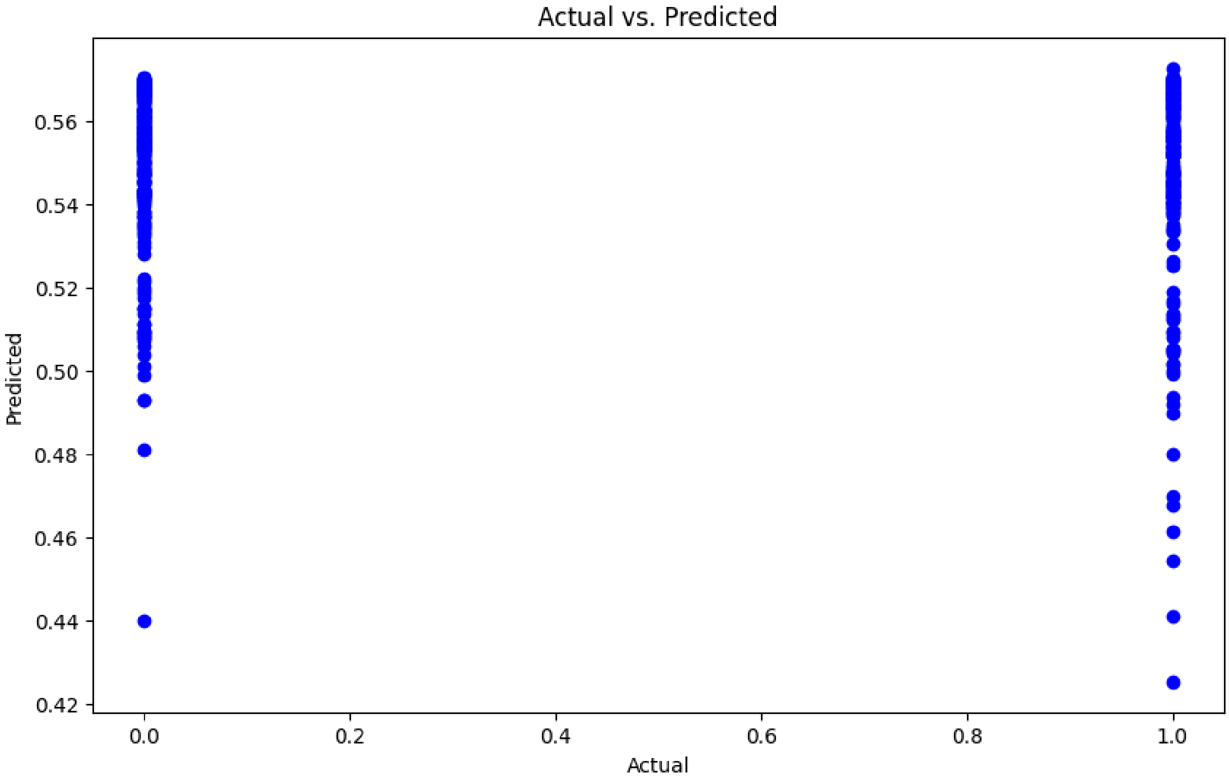


fig.4:A DataFrame to compare actual and predicted values

3.Results & Discussion

The bar chart fig5. visualizes the performance metrics for different models, including Linear Regression, Decision Tree, Naive Bayes, and KNN.

- Accuracy:
    - Linear Regression: 65.22%
    - Decision Tree: 48.98%
    - Naive Bayes: 51.22%
    - KNN: 50.20%
- Precision:
    - Linear Regression: 0.62
    - Decision Tree: 0.49
    - Naive Bayes: 0.50
    - KNN: 0.50
- Recall:
    - Linear Regression: 0.84
    - Decision Tree: 0.51
    - Naive Bayes: 0.88
    - KNN: 0.57
- F1-score:
    - Linear Regression: 0.71
    - Decision Tree: 0.51
    - Naive Bayes: 0.65
    - KNN: 0.54

These metrics provide insights into how well each model performs in predicting stock index movements.

Discussion:

- Linear Regression achieved the highest accuracy among the models, indicating its effectiveness in predicting stock movements based on the given features.
- Decision Tree and KNN models showed lower accuracy compared to Linear Regression, suggesting they may not capture the underlying patterns in the data as effectively.
- Naive Bayes achieved a comparable accuracy to Decision Tree and KNN, but with higher precision and recall, indicating a better balance between correctly identifying positive cases and avoiding false positives.
- Naive Bayes also demonstrated the highest recall, indicating its ability to identify a larger proportion of true positive cases compared to other models.

- Precision scores for all models are relatively close, suggesting that while they may differ in their ability to correctly classify positive cases, they are similar in avoiding false positives.
- Overall, the choice of model depends on the specific requirements of the prediction task, considering factors such as interpretability, computational efficiency, and the importance of correctly identifying positive cases.

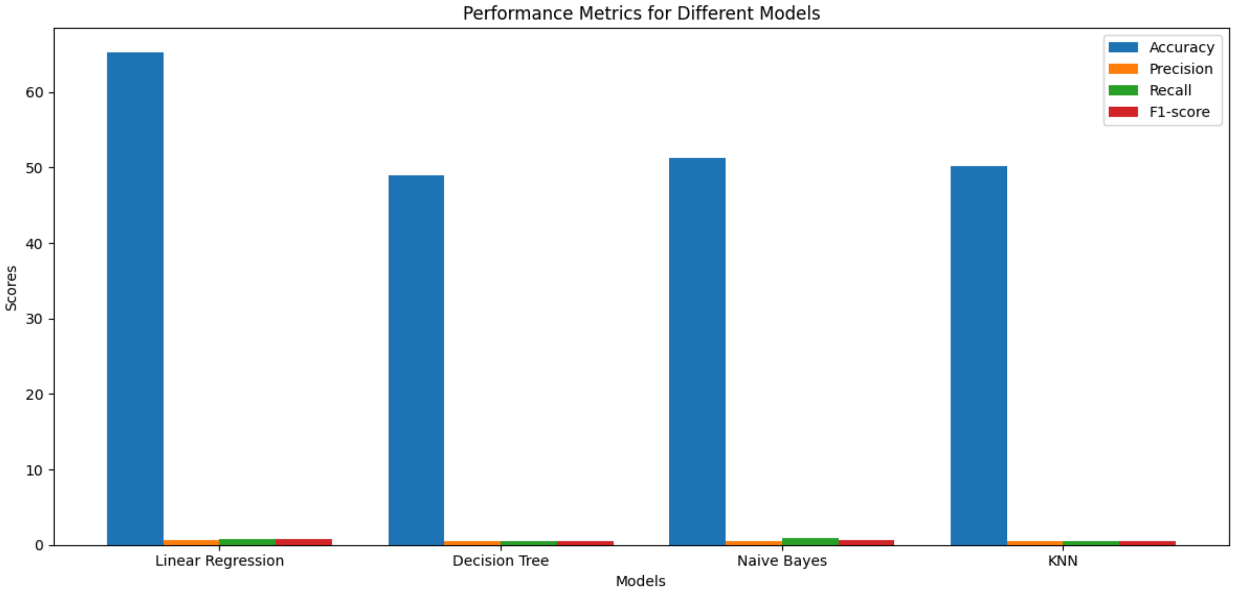


fig 5.Performance of different models

4.Conclusion

In this project, we aimed to develop enhanced predictive models for stock index movements using various machine learning techniques. We utilized a dataset containing OHLC (Open, High, Low, Close) and Volume data from Yahoo Finance, along with macroeconomic features from FRED (Federal Reserve Economic Data). Our project involved several key steps:

1. Data Collection and Preprocessing:
    - We collected and preprocessed the dataset, which included handling missing values, scaling features, and performing principal component analysis (PCA) for dimensionality reduction.
2. Model Selection and Training:
    - We applied multiple machine learning algorithms, including Linear Regression, Decision Tree, Naive Bayes, and KNN.
    - Each model was trained on the preprocessed dataset to predict whether the stock index would move up or down.
3. Model Evaluation:
    - We evaluated the performance of each model using various metrics such as accuracy, precision, recall, and F1-score.
    - Additionally, we visualized the performance metrics using bar charts to compare the models.
4. Results and Discussion:
    - Linear Regression emerged as the top-performing model in terms of accuracy, achieving a score of 65.22%.
    - Decision Tree, Naive Bayes, and KNN also showed competitive performance, but with varying strengths and weaknesses in precision, recall, and F1-score.
    - Naive Bayes demonstrated the highest recall, indicating its ability to identify a larger proportion of true positive cases.
    - The choice of model depends on specific requirements such as interpretability, computational efficiency, and the importance of correctly identifying positive cases.

Future Work:

Moving forward, several avenues could be explored to enhance the predictive models further:

- Experimenting with additional features or alternative data sources to capture more information relevant to stock market movements.
- Fine-tuning model hyperparameters and exploring ensemble learning techniques to improve overall performance.
- Investigating more advanced machine learning algorithms and deep learning architectures to potentially uncover more complex patterns in the data.

Overall, this project provides valuable insights into the application of machine learning techniques for stock market prediction and lays the foundation for future research in this domain.

5.References

[1] Yahoo Finance. (n.d.). Retrieved from https://finance.yahoo.com/
[2] Federal Reserve Economic Data (FRED). (n.d.). Retrieved from https://fred.stlouisfed.org/
[3] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction. Springer Science & Business Media.