Machine Learning Assignment

PROJECT REPORT

TEAM ID : 8

PROJECT TITLE:

Using machine learning models to Predict S&P 500 Price level and spread direction

Name	SRN
CHARAN M REDDY	PES2UG23CS146
DILEEP	PES2UG23CS177

Problem Statement

Financial markets are volatile and unpredictable, making it challenging for investors and traders to make short-term decisions about the S&P 500 index. Traditional analysis methods often fail to capture complex patterns in historical price movements and technical indicators.

We Have Developed data-driven predictive models that can analyze historical market data, identify patterns, and forecast both the price levels and directional movement (spread) of the S&P 500 index . The models aims to achieve reliable predictions that can serve as a decision-support tool for traders and investors in navigating S&P 500 index movements.

Objective / Aim

We Developed machine learning models to:

Predict the S&P 500 closing price for the next trading day using regression techniques .

Classify the spread direction using classification algorithms.

Engineer meaningful features from historical price data including technical indicators (Moving Averages, RSI, Bollinger Bands) .

Evaluate model performance using appropriate metrics (RMSE, MAE for regression; Accuracy, Precision, Recall for classification) .

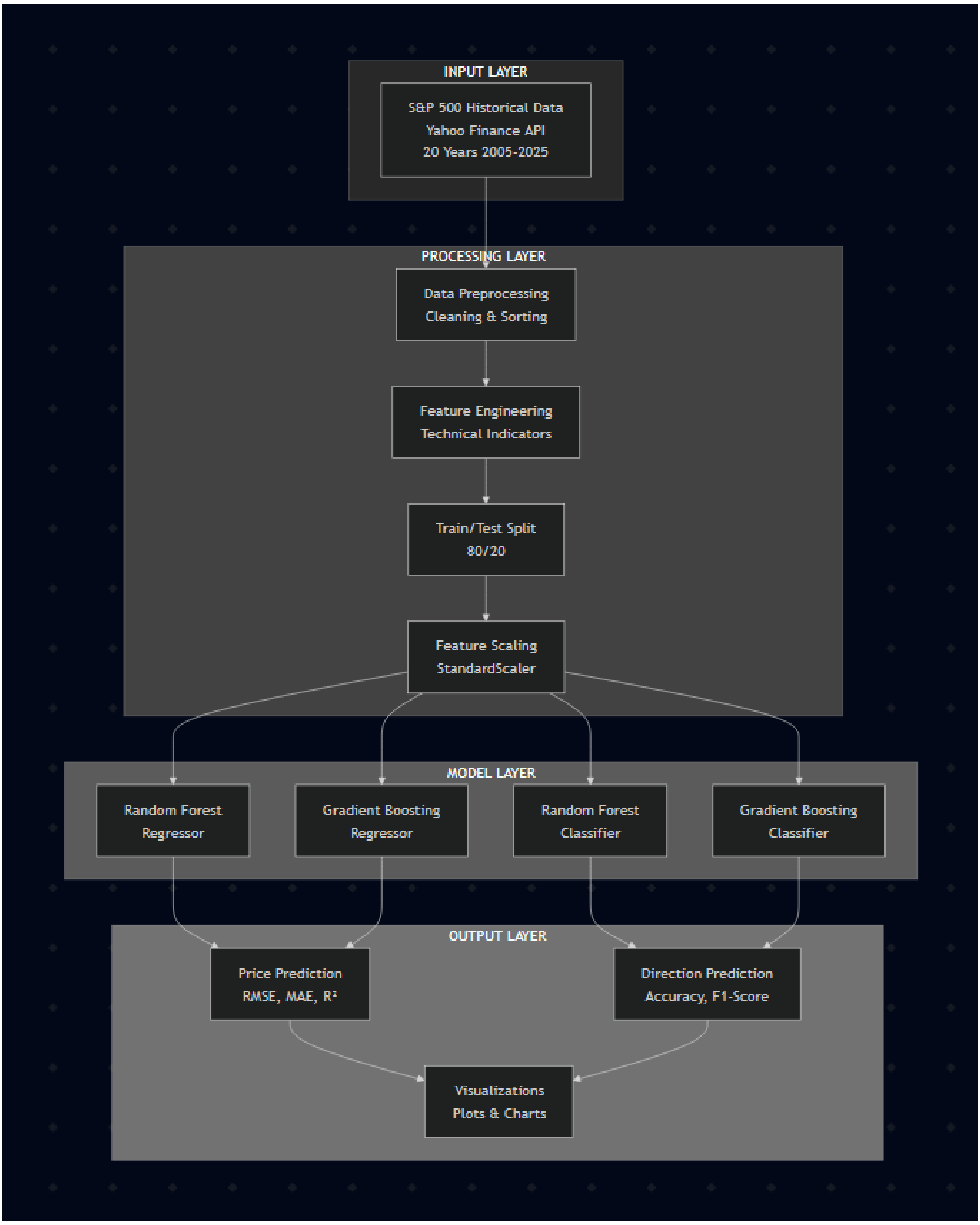
Provide actionable insights for short-term financial decision-making by comparing multiple ML algorithms (Random Forest, Gradient Boosting, Neural Networks) .

Dataset Details

- **Source:** Yahoo Finance (through python library 'yfinance')
- **Size:** 5030 samples , 5 features before feature engineering , 21 after that .
- **Key Features:** Low , SMA_50 , Bb_upper , SMA_200 , High
- **Target Variable:** Next_Close (for day of prediction)

Architecture Diagram

You can use Mermaid or Eraser.io to create a detailed Architecture diagram



Methodology

- Downloaded 20 years of S&P 500 historical data from Yahoo Finance using yfinance library
- Cleaned the dataset by removing multi-level headers and organizing columns properly
- Created technical indicators including Moving Averages (SMA 20, 50, 200), RSI, Bollinger Bands, MACD, and ATR
- Added lagged features like previous day's closing price and daily returns to capture momentum
- Defined two target variables: next day's closing price for regression and spread direction (up/down) for classification
- Removed rows with missing values created during indicator calculation
- Split data into 80% training and 20% testing using time-series split to maintain chronological order
- Applied Standard Scaling to normalize all features
- Trained four models: Random Forest Regressor, Gradient Boosting Regressor, Random Forest Classifier, and Gradient Boosting Classifier
- Evaluated models using appropriate metrics and compared their performance

Results & Evaluation

Regression Task (Price Prediction):

Both Random Forest and Gradient Boosting Regressors showed similar performance with moderate prediction accuracy.

Random Forest achieved an RMSE of 760.13 and MAE of 502.55, while Gradient Boosting recorded RMSE of 760.51 and MAE

of 512.48. The R^2 scores for both models were around 0.15, indicating they captured about 15% of price variance.

The relatively high error values occurred because the test period (2024-2025) included a strong bull market rally

where prices exceeded the training data range. Models struggled to predict prices beyond what they learned during

training, which is a common limitation in financial forecasting.

Evaluation Metrics Used:

RMSE (Root Mean Squared Error) - measures average prediction error magnitude

MAE (Mean Absolute Error) - average absolute difference between predicted and actual prices

R^2 Score - proportion of variance in prices explained by the model

Classification Task (Spread Direction):

The classification models showed limited success in predicting market direction. Random Forest Classifier achieved

47.72% accuracy with precision of 0.50 and recall of 0.33. Gradient Boosting Classifier performed slightly worse at

45.34% accuracy. The F1-scores hovered around 0.40 for both models, indicating difficulty in reliably predicting whether

the market would move up or down.

The confusion matrices revealed that both models struggled more with identifying upward movements (lower recall for

positive class) compared to downward movements. This suggests the models were conservative in predicting bullish trends.

Evaluation Metrics Used:

Accuracy - percentage of correct directional predictions

Precision - reliability of upward movement predictions

Recall - ability to catch actual upward movements

F1-Score - harmonic mean of precision and recall

Confusion Matrix - detailed breakdown showing true positives, false positives, true negatives, and false negatives

Feature Importance Analysis:

The top five most influential features were Low price, SMA_50 (50-day moving average), BB_upper

(Bollinger Band upper bound), SMA_200 (200-day moving average), and High price. This shows that price levels and

medium-to-long-term moving averages were the strongest predictors for the models.

Conclusion

The regression models achieved moderate success with RMSE around 760 points, but the low R^2 scores (0.15) revealed that most price movements were driven by factors beyond historical patterns and technical indicators.

The classification task proved even more challenging, with accuracy barely above random guessing (around 45-48%), showing that predicting market direction is extremely difficult even with feature engineering.

Random Forest and Gradient Boosting performed similarly across both tasks, with neither showing superiority. The feature importance analysis confirmed that price levels and moving averages were the most valuable predictors, validating the relevance of technical analysis in trading strategies.

Through this project, I gained hands-on experience with the complete machine learning workflow for financial data. I learned how to collect and clean time-series data, engineer technical indicators from raw price information, and properly split data chronologically to avoid bias.

The project taught me that while machine learning can identify historical patterns, financial markets are influenced by countless external factors like economic news, geopolitical events, and investor sentiment that cannot be captured through price history alone.

Most importantly, I understood that stock prediction models should be viewed as decision-support tools rather than primary tools for decision. The mediocre performance shows why professional traders combine multiple strategies and risk management techniques rather than relying solely on predictive models.