University of North Texas

Introduction to Big Data and Data Science – CSCE 5300

Group Number - 09

Team Members:

1. *Koushik Reddy Sudireddy – 11885863*
2. *Merla Ganesh Reddy – 11797294*
3. *Ruthvika Muchala - 11906895*

Table of contents

Abstract

The goal of this project is to predict the next-day stock price movement (UP/DOWN) using machine learning. A large historical price dataset was cleaned, processed, and enriched with technical indicators, totalling 851,264 rows across 501 stocks. Multiple ML models, including Logistic Regression, Random Forest, and XGBoost, were trained. Their performance is evaluated with regards to accuracy, precision, recall, F1-score, and ROC-AUC. The walk-forward validation approach is implemented for realistic temporal evaluation, while financial backtesting is conducted in a bid to simulate real-world trading. The results achieved show modest predictive power, with models at best achieving about 53% accuracy.

Introduction

Stock price prediction remains a challenging problem due to noise, volatility, and multiple external influences. Machine learning offers tools to analyse historical behavior and identify predictive patterns. This project aims at developing a robust time-series prediction pipeline using technical indicators and ML models and evaluating the practical utility of predictions via backtesting.

Problem Statement & Objectives

Problem Statement

To predict whether the next day’s stock closing price of stocks will be higher or lower than today’s, using historical price data and technical indicators.

Objectives

- Preprocess large-scale multi-stock historical datasets.
- Perform detailed exploratory data analysis.
- Engineer the relevant financial and technical features.
- Train ML models to classify next-day movement.
- Evaluate performance using standard classification metrics.
- Perform walk-forward validation and backtesting.

Data Description

Dataset source: Kaggle:

- prices.csv : Open, close, high, low, volume (851k rows).
- securities.csv : Company metadata including GICS sector.
- Time Period: 2010–2016
- Symbols: 501
- It parses dates, sorts data, merges it with metadata, and removes entries where values are missing.

Methodology

1. Data Loading & Cleaning

Historical stock price data was imported, checked for inconsistencies, and cleaned by handling missing values, parsing dates, and sorting records chronologically to prepare the dataset for analysis.

2. Merging Sector Information

Company metadata, including sector classifications, was merged with the price dataset to enrich the data and enable sector-wise analysis and filtering during modeling.

3. Exploratory Data Analysis (EDA)

An initial analysis was performed to understand sector volumes, price behaviors, and correlations between variables, helping identify patterns and guide feature engineering decisions.

4. Feature Engineering (Technical Indicators, Lags)

A variety of predictive features were generated, including lagged returns, moving averages, RSI, MACD, Bollinger Bands, volatility metrics, and volume-based indicators to capture market trends and momentum.

5. Train/Validation/Test Split Using Chronological Order

The dataset was split into training, validation, and testing sets based strictly on time to prevent data leakage and mimic real-world forecasting scenarios.

6. Model Training

Multiple machine learning models—including Logistic Regression, Random Forest, and XGBoost—were trained using the engineered features to learn patterns associated with next-day stock movement.

7. Performance Evaluation

Model predictions were assessed using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to determine their predictive effectiveness and compare them with baseline strategies.

8. Walk-Forward Validation

A rolling, time-based validation method was applied to evaluate how models perform in sequential time periods, simulating realistic model deployment in changing market environments.

9. Backtesting Using Random Forest

The best-performing model's predictions were used to simulate trades over historical data, generating portfolio returns and financial metrics to understand real-world profitability.

Feature Engineering

The following features were created:

1. **Return-based Features**
   - Return
   - Next-day return
   - Lagged returns (1, 2, 4, 8 days)
2. **Moving Averages**
   - MA(5), MA(10), MA difference
3. **Volatility Indicators**
   - Rolling volatility (10 days)
   - Average True Range (ATR 14)
4. **Momentum Indicators**
   - RSI (14)
   - MACD & Signal line
5. **Bollinger Bands**

- High band, low band, band width

6. **Volume Features**

- Volume change
- Relative volume

7. **Date Features**

- Day of week
- Month
- Quarter

Model Development

Models trained on BAC stock after global feature generation:

1. Logistic Regression (balanced class weights)
2. Random Forest Classifier
3. XGBoost Classifier

Baseline comparisons:

- Previous day return
- Moving average crossover

Evaluation Metrics

- Accuracy
- Precision
- Recall
- F1-score
- ROC-AUC

Results & Observations

Baseline Accuracy

- Previous day return: **0.4887**
- MA crossover: **0.4923**

Machine Learning Results

Model	Accuracy	Precision	Recall	F1	ROC-AUC
Logistic	0.494	0.507	0.395	0.444	0.506
Random Forest	**0.535**	**0.562**	0.407	0.472	0.547
XGBoost	0.526	0.541	**0.486**	**0.512**	**0.550**

Random Forest highest accuracy; XGBoost best balance of metrics.

Walk-Forward Validation

Walk-forward simulates real-time prediction.
Average results across 5 folds:

- Accuracy: **0.49–0.50**
- F1 score: up to **0.515 (XGBoost)**
- Shows model stability but limited predictive edge.

Backtesting Results

Using Random Forest predictions:

- **Sharpe Ratio:** 0.560
- **Max Drawdown:** -0.317
- **Final Strategy Return:** 1.146
- **Buy & Hold:** 1.249

Model does **not** outperform buy-and-hold.

Literature Survey

Stock market prediction has always presented a challenge because of its volatile and non-stationary behavior associated with financial time series data. In initial stages of stock market prediction, techniques like ARIMA, GARCH, and linear regression analysis were mostly adopted for prediction based on their statistical techniques. Though these techniques helped to analyze market trends and volatility accurately, they lack capabilities to deal with market non-linear relationships associated with stock markets.

Then machine learning techniques started to gain popularity since they showed capabilities to utilize non-linear patterns beyond linear relationships. Support Vector Machine (SVM) algorithms were among the first to demonstrate their efficiency in being better than traditional methods (Huang et al., 2005). The development of ensemble learning techniques such as Random Forest provided better robustness to outliers and capabilities to generalize while learning from technical charts (Chen & Hao, 2017). The development of XGBoost further transformed predictive modeling on structured data into making highly accurate predictions (Chen & Guestrin, 2016).

Models of deep learning and specifically LSTM networks have also been tested for their capability to analyze long dependencies extracted from sequence data. It was shown by Fischer & Krauss (2018) that LSTM-based architectures have demonstrated their effectiveness and could potentially surpass traditional approaches while being tested for several stocks over large time horizons. A number of works have also discussed limitations associated with high probabilities of overfitting, lack of data, or random behavior associated with short-term directions for stock prices to move.

Technical analysis techniques such as moving averages, RSI, or MACD have remained integral to both theoretical and applied prognostication systems to-date, though it is universally acknowledged in theoretical studies that any one technique is never reliable to any significant extent. Combining several techniques through machine learning algorithms is but a step towards improvement beyond random outcomes, as expected from numerous contemporary prognostication studies conducted for finance-related prediction tasks.

On the whole, it appears from the literature that while stock prediction is still a challenge, machine learning techniques based on engineered features seem to hold promise for making progress on this task.

Limitations

- Stock movements near to random, thus poor ML predictability.
- Only technical indicators were used, no sentiment/news/macro data.
- Single-stock modeling limits generalization.
- No hyperparameter search (default + light tuning).
- No transaction cost modelling

Proposal Requirements that couldn’t be done

1. LSTM Model

The LSTM model was not implemented due to the additional preprocessing steps required to convert the dataset into sequence format, which was time-consuming and exceeded the project timeline.

2. SHAP (Feature Interpretation)

SHAP analysis was planned to interpret model decisions, but it was not completed because the computation time was very high on the full dataset, and running SHAP on boosted models requires additional optimization and GPU support.

3. Sentiment Analysis from News

News sentiment could not be added as the dataset did not include historical news headlines, and integrating an external API or dataset required extra preprocessing and alignment that was not feasible within the deadline.

Future Enhancements

- Include LSTM/GRU or Transformers for sequence modeling.
- Add sentiment analysis from news/Twitter.
- Include macroeconomic indicators.
- Portfolio-level as opposed to single stock prediction.
- Advanced backtesting: slippage, fees.
- Cross-validation with the expanded walk-forward method.

Conclusion

This study examined the effectiveness of machine learning algorithms to make predications regarding short-term stock price movements by employing technical indicators extracted from past stock prices. After pursuing extensive activities involving data cleaning and analysis, several machine learning models were developed and tested. It was found that the Random Forest and XGBoost models performed comparatively better than simple baseline methods such as yesterday's return and moving average crossover strategies, but not very effectively (around 53% accuracy), thus confirming past studies validating the high randomness and efficiency of stock markets and making it very tough to predict short-term stock prices.

The project also conducted backtesting and found that no trader strategy informed by their model performed better than simply buying and holding. This further supports how intricate analysis of financial markets is to accomplish using just technical variables. Nevertheless, this project succeeded at showcasing how all workflows associated with making predictive analysis for finance from beginning to end should be achieved. This makes for easy expansion of this project into things such as LSTM and sentiment analysis addition to more complex financial values.

Members Contribution

Member 1: Ruthvika Muchala — Data & Feature Engineering + Baselines

Contributions:

- Collected and preprocessed NYSE dataset.
- Cleaned and merged price + fundamentals files.
- Generated global features including:
    - lagged returns
    - moving averages
    - volatility windows
    - RSI, MACD, Bollinger Bands
    - volume change indicators
- Built baseline models:
    - Previous day return
    - Moving average crossover
- Performed exploratory data analysis and visualizations.

Member 2 : Koushik Sudireddy — Machine Learning Models + Evaluation

Contributions:

- Implemented ML models:
    - Logistic Regression (balanced)
    - Random Forest
    - XGBoost
- Tuned hyperparameters manually where required.
- Evaluated models using accuracy, precision, recall, F1-score, ROC-AUC.
- Compared ML results with baselines and interpreted findings.
- Generated confusion matrices, ROC curves, and performance summaries.

Member 3 : Merla Ganesh Reddy — Backtesting, Financial Metrics & Report

Contributions:

- Implemented backtesting framework using model predictions.
- Computed financial metrics:
  - Cumulative returns
  - Sharpe ratio
  - Maximum drawdown
- Validated model performance under realistic trading constraints.
- Prepared final report, formatting, documentation, and created presentation slides.
- Integrated visualizations and final interpretations into the report.

Github link : https://github.com/koushikreddy17/Stock-Price-Movement-Prediction-using-NYSE-Dataset.git