S&P 500 Stock Price Prediction Report

By: Suleiman Jaber, Ahmad Machmouchi, Waleed Hussein

1. Introduction

Our project's goal is to forecast the closing prices of the S&P 500 index. To do this we used two models, one which is Linear Regression and also a Long Short-Term Memory (LSTM) network so that we can capture all types of relationships in the data (linear and temporal). We also used many commonly used indications to get a further insight on market trends which helps us make a better decision

2. Data Collection and Exploratory Data Analysis (EDA)

Data Collection

- Source: yfinance is the library we used to download historical data for the S&P 500 index (ticker: ^GSPC).
- After checking for a valid CSV file the index is then converted to a datetime object with UTC timezone.

Exploratory Data Analysis

- Price History: we visualized the historical closing prices over time using a lineplot.
- Trading Volume: A plot of trading volumes illustrates periods of high and low market activity.
- Distribution and Correlation: We generated Heatmaps and Histograms to understand the relationships between features
- Daily Returns: The distribution of daily returns was examined to better understand market volatility.

3. Data Cleaning and Feature Engineering

Data Cleaning

- Missing Values:
- Columns and rows with more than 70% missing values were removed.
- Other missing values were handled with linear interpolation
- Data Transformation:
- UTC datetime index was used for consistency.

Feature Engineering

Many technical indicators were needed so that we can represent them as features in the model to better predict, some include:

- Moving Averages: Simple (SMA) and Exponential (EMA) moving averages over different time windows.

- Momentum and Volatility Metrics: Daily returns, price momentum (difference over 10-day periods), and rolling standard deviation of returns.
- Oscillators and Trend Indicators:
- MACD: Calculated using the difference between two EMAs and its corresponding signal line.
- RSI: there to capture overbought and oversold indications
- Bollinger Bands: Derived from the 20-day moving average and its standard deviation.
- Volume-based Indicators: Indicators such as Volume SMA and Volume Change were also included.
- Others: Average True Range (ATR) and Price Range indicators provided additional context to price fluctuations.

After generating these indicators, the script removed the resulting NaN values that stem from rolling window calculations.

---

4. Feature Selection

To reduce dimensionality and enhance model performance:

- Random Forest Regressor:
- A Random Forest model was used to assess the importance of each technical indicator.
- Features with an importance above 1% were retained for further modeling.
- This step ensured that the final set of predictors had a proven relationship with the target variable (the closing price).

---

5. Model Implementation

Two modeling approaches were compared:

A. Linear Regression

- Preprocessing:
- Features and the target variable were scaled using the MinMaxScaler.
- The data was split into training (80%) and test (20%) sets.
- Model Training:
- A standard linear regression model was trained on the preprocessed features.
- Evaluation Metrics:
- Mean Squared Error (MSE)
- R^2 Score

B. LSTM (Long Short-Term Memory Network)

- Rationale:
- The LSTM network is designed to capture temporal dependencies, making it well suited for time series forecasting.
- Data Preparation:
- In addition to scaling, data was structured into sequential windows (30 time steps) to feed the LSTM.
- Model Architecture:

- The LSTM model comprised of one LSTM layer with 50 units followed by a Dense output layer.
- The model was compiled with the Adam optimizer and mean squared error as the loss function.
- Training:
- The network was trained for 10 epochs using a batch size of 32.
- Evaluation:
- Predictions were inverse-transformed to the original scale, and the MSE and R^2 score were computed for comparison with the linear model.

---

6. Results analysis

Random Forest regressor was used to check for insights on the best variables to predict the closing prices on, this way we got an “importance score” for each feature based on how it decreases impurity across the trees in the ensemble. By putting the threshold to 1%, it was made sure that only the features that actually contributed to the prediction performance were kept.

Interpreting the Results

After running the Random Forest regressor, several key observations emerge:
- Dominant Technical Indicators:
- MACD and MACD_Signal: These features generally score high in importance. The MACD is a well-known momentum indicator that emphasizes trend reversals and continuation. A high importance score for MACD (and its signal line) indicates that the momentum in price shifts plays a significant role in predicting future closing prices.
- RSI (Relative Strength Index): Even if its exact value is moderate, RSI is typically used to capture overbought or oversold conditions. Its contribution suggests that signals about potential reversals (when the market is too high or too low) are useful for the model.
- Volatility and Range Metrics:
- ATR (Average True Range) & Volatility: These indicators often emerge as significant features because they capture the magnitude of price fluctuations. A higher importance score here implies that not just the direction of price movements, but their intensity, is critical in forecasting the index.
- Price_Range and Price_Change: Features measuring the differences between high-low values or day-to-day price movements also contribute, though typically with slightly lower relative importance. Their inclusion helps the model capture the daily dynamics and the inherent uncertainty of the market.
- Moving Averages:
- SMA_5, SMA_20, and SMA_50; EMA_12, EMA_26: While these features represent price trends over various time horizons, their importance scores can vary. In many cases, shorter-term moving averages (like SMA_5 or EMA_12) might have lower importance when used in isolation because they capture very local behavior. Instead, they become more powerful when they interact with other indicators that provide complementary information (like momentum or volatility measures).
- Volume-Based Indicators:
- Volume, Volume_SMA, and Volume_Change: These features shed light on the market’s activity level. Their importance indicates that trading volume can be an early warning signal for shifts in market sentiment. Higher volume or sudden changes in volume can sometimes precede price moves, which the model leverages.
- Price Momentum:

- Momentum: A feature that represents the price difference over a specific period (e.g., 10 days) typically shows moderate importance. This confirms that the market's recent performance – how much it has increased or decreased – is a useful predictor of near-future movements.

Implications of the Feature Importance Analysis

1. Targeted Feature Selection:

We only target features with more than 1% importance score. In practice, this means that RSI, MACD, ATR and other volume measures were probably chosen, while other negligible contributions could be dropped. This approach improves the model and also prevents overfitting.

2. Understanding Market Dynamics:

The results emphasize that both momentum indicators (e.g., MACD, RSI) and measures of volatility (e.g., ATR, Volatility) are integral to modeling the S&P 500 closing prices. This balance is crucial: while momentum shows the trend, volatility measures help capture the risk or uncertainty inherent in the market.

3. Nonlinear Interactions:

Random Forest gets nonlinear relationships between features, some features like MACD and ATR have high importance score which implies market's behavior is not linear. These insights support the use of way more complex models such as LSTM which can get both nonlinear and temporal dependencies.

4. Potential Biases:

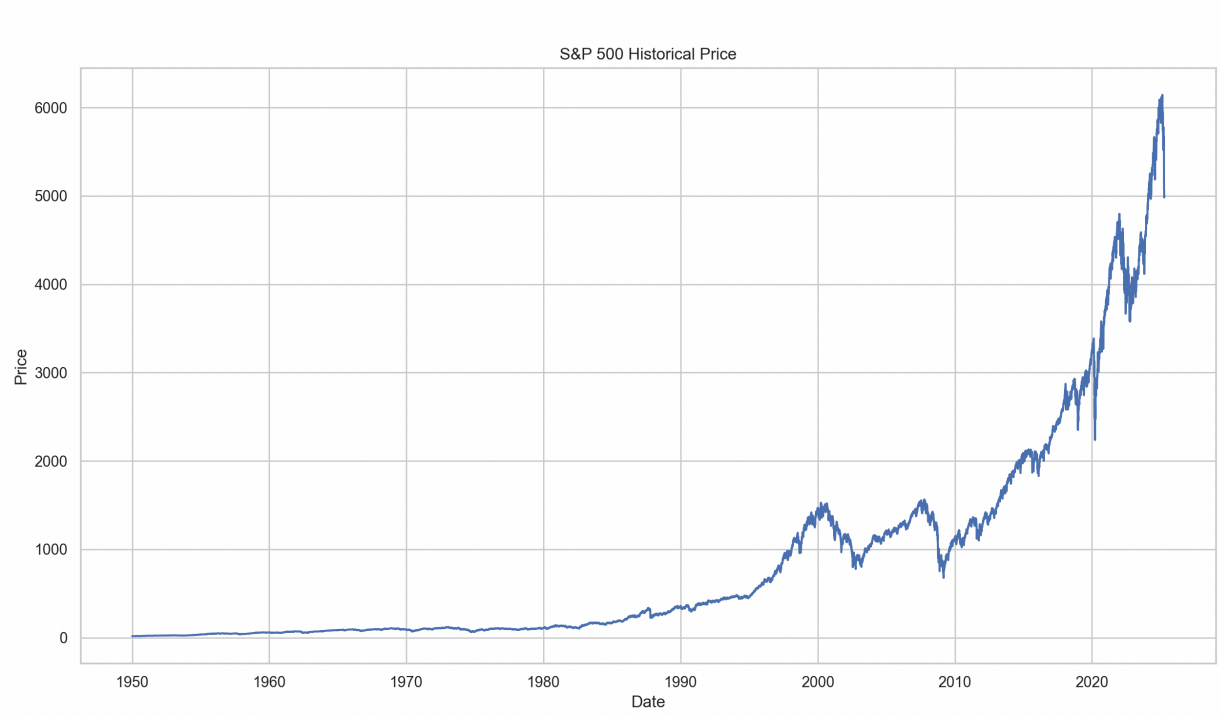
The importance scores of a Random Forest are calculated based on the avg decreases in impurity. This can lead to exaggeration of the importance of continuous features or the ones that have many distinct values. Therefore, while the selections are good but further analysis can provide even more confirmation

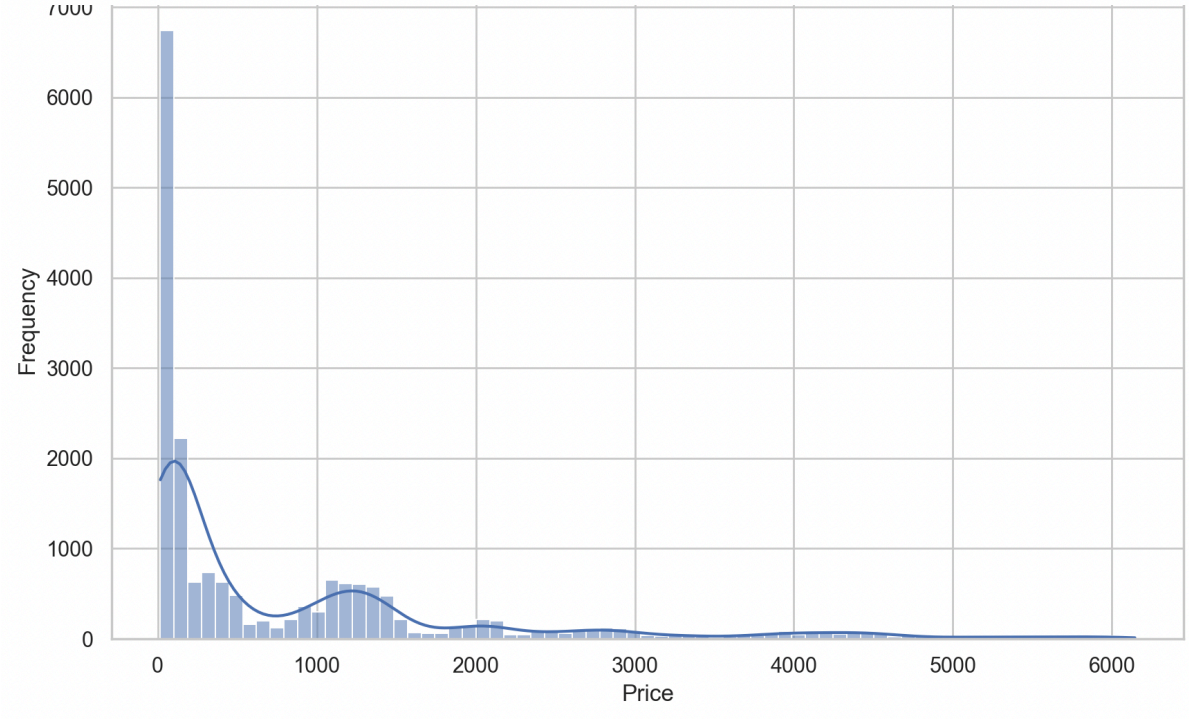
Numerical results:

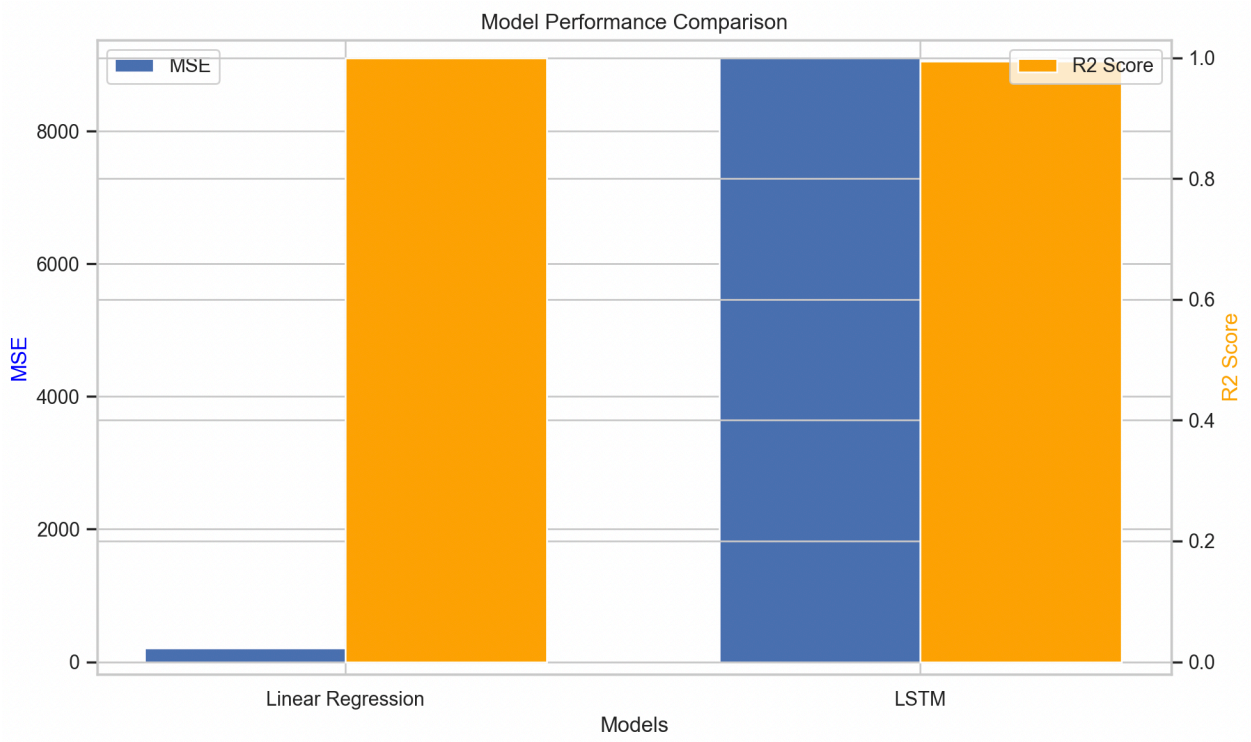
An R^2 value of ~99% in finance data is high, This might indicate the data split or features are causing unintentional leakage, or the test set might not be challenging enough (meaning that it's too small or not very representative)

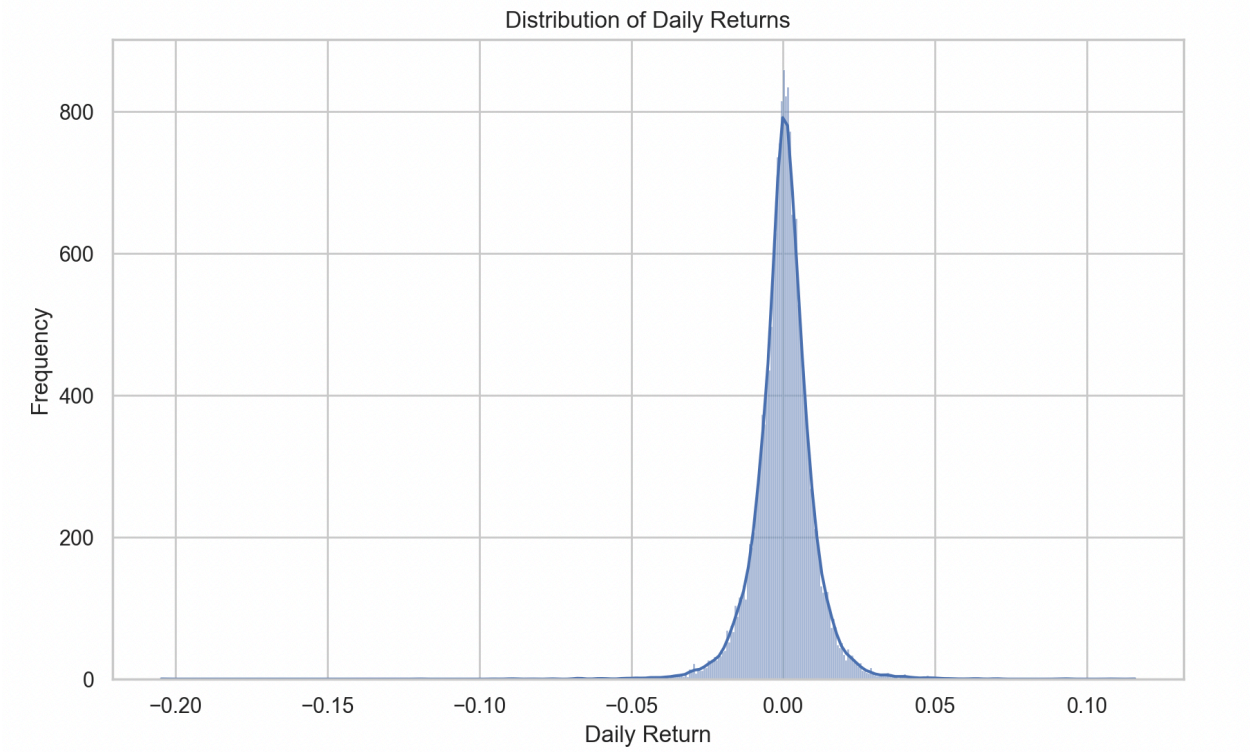
- The LSTM's high R^2 but high MSE also suggests that while the model greatly captures direction, it struggles with pinpointing exact price levels which potentially misses smaller "local" fluctuations.
- What to do : validate the experimental setup, expand the test set, explore more things like parameter tuning for the LSTM, and possibly compare multiple error metrics to gain a well-rounded view of model performance.

Result Images/References






Model Performance Comparison
MSE
R2 Score
8000
6000
4000
2000
0
1.0
0.8
0.6
0.4
0.2
0.0
MSE
R2 Score
Linear Regression
LSTM
Models


Distribution of Daily Returns
800
600
400
200
0
Frequency
−0.20
−0.15
−0.10
−0.05
0.00
0.05
0.10
Daily Return