Stock Market Prediction

Nexus Info_ Saloni Nimgaonkar

Introduction

Predicting stock market values has become an essential task for analysts, financial institutions, and investors alike in today's dynamic and linked financial world. It is possible to make a big difference between large profits and large losses by having the capacity to predict market trends and movements. The area of stock market prediction has improved significantly in recent years thanks to the development of sophisticated computing tools and the accessibility of large volumes of financial data.

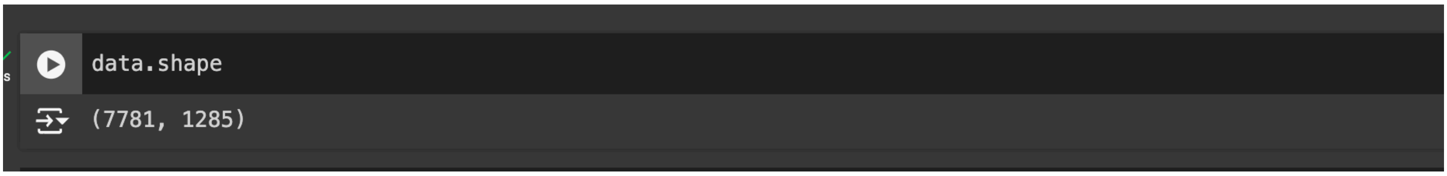
With the goal of forecasting stock market values, this study provides a thorough examination of a dataset. Among the many aspects included in the dataset are trade volume, historical price data, and a variety of technical indicators that are based on market performance. We aim to build strong prediction models that can accurately anticipate the future prices of stocks or other financial instruments by utilising machine learning techniques and statistical models.

Understanding the dataset

1. **Date**: The date of the trading day.
2. **Open**: The price at which a security first trades upon the opening of an exchange.
3. **High**: The highest price at which a security traded during the trading day.
4. **Low**: The lowest price at which a security traded during the trading day.
5. **Close**: The final price at which a security is traded on a given trading day.
6. **Adj Close**: The closing price of a security that has been adjusted to reflect all relevant information, such as dividends and stock splits.
7. **Volume**: The number of shares or contracts traded during a given period.
8. **Ticker**: The symbol representing a particular security on a stock exchange.
9. **RSI (Relative Strength Index) adjclose 15**: RSI is a momentum oscillator that measures the speed and change of price movements. The formula for RSI is: RSI = 100 - (100 / (1 + RS)), where RS (Relative Strength) = Average gain over N periods / Average loss over N periods. In this case, "adjclose 15" refers to the adjusted close price over the last 15 periods.
10. **RSI volume 15**: Similar to RSI adjclose 15, but calculated based on volume instead of price.

11. **High-15**: The highest price observed over the last 15 periods.
12. **K-15 (Fast Stochastic Oscillator)**: K = (Current Close - Lowest Low) / (Highest High - Lowest Low) * 100. It measures the position of the most recent closing price relative to the range over the last N periods.
13. **D-15 (Slow Stochastic Oscillator)**: D = 100 * (H3 / L3), where H3 is the 3-day sum of (C - L) and L3 is the 3-day sum of (H - L). It is a moving average of K over the last N periods.
14. **Stochastic-K-15**: The current value of the fast stochastic oscillator (K-15).
15. **Stochastic-D-15**: The current value of the slow stochastic oscillator (D-15).
16. **Stochastic-KD-15**: The interaction between the fast and slow stochastic oscillators.
17. **Volumenrelativo**: The ratio of the current volume to the average volume over a certain period.
18. **Diff**: The difference between two values, often used to indicate changes or divergences.
19. **INCREMENTO**: Indicates whether the price increased or decreased compared to the previous period.
20. **TARGET**: The target variable, which could represent various things depending on the context, such as a classification label (e.g., buy, sell, hold) or a regression target (e.g., predicted price change).

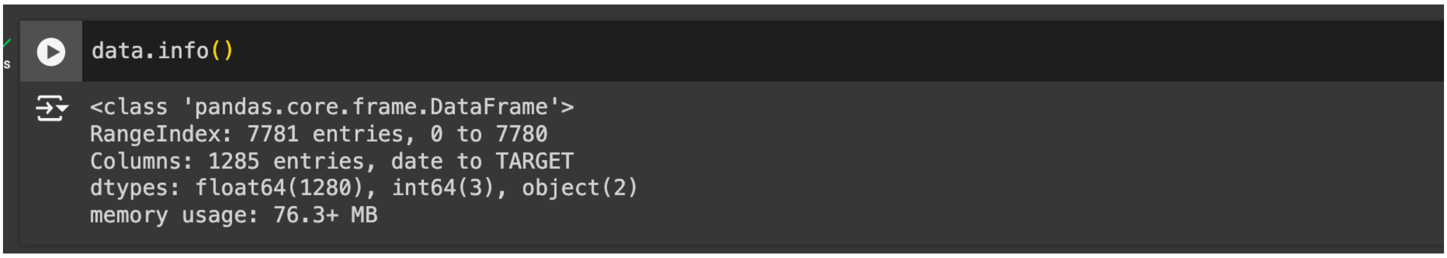
Exploratory Data Analysis



```
data.shape
```

```
(7781, 1285)
```

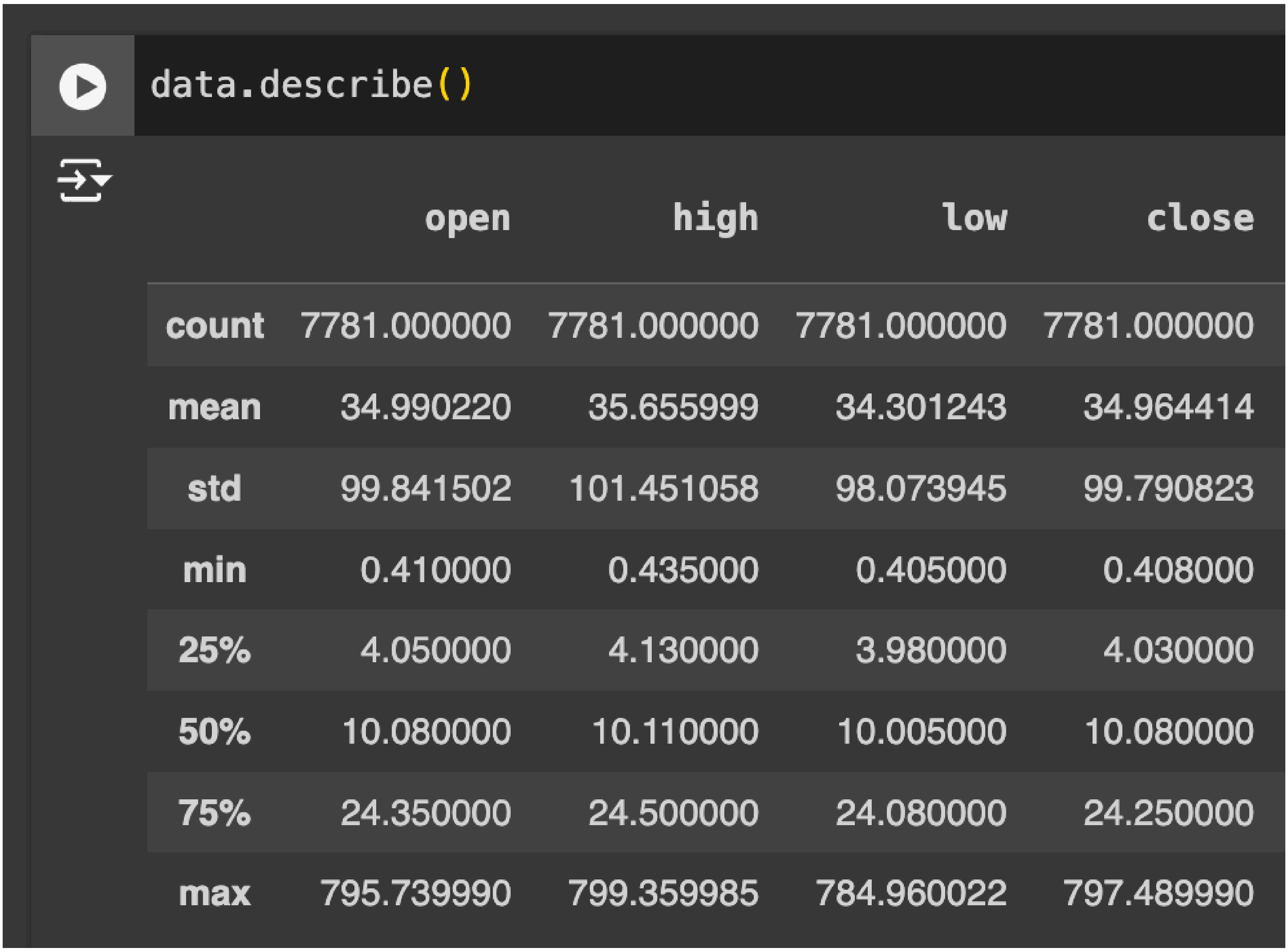
The shape of (7781, 1285) indicates that your dataset has 7781 rows and 1285 columns.



```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7781 entries, 0 to 7780
Columns: 1285 entries, date to TARGET
dtypes: float64(1280), int64(3), object(2)
memory usage: 76.3+ MB
```

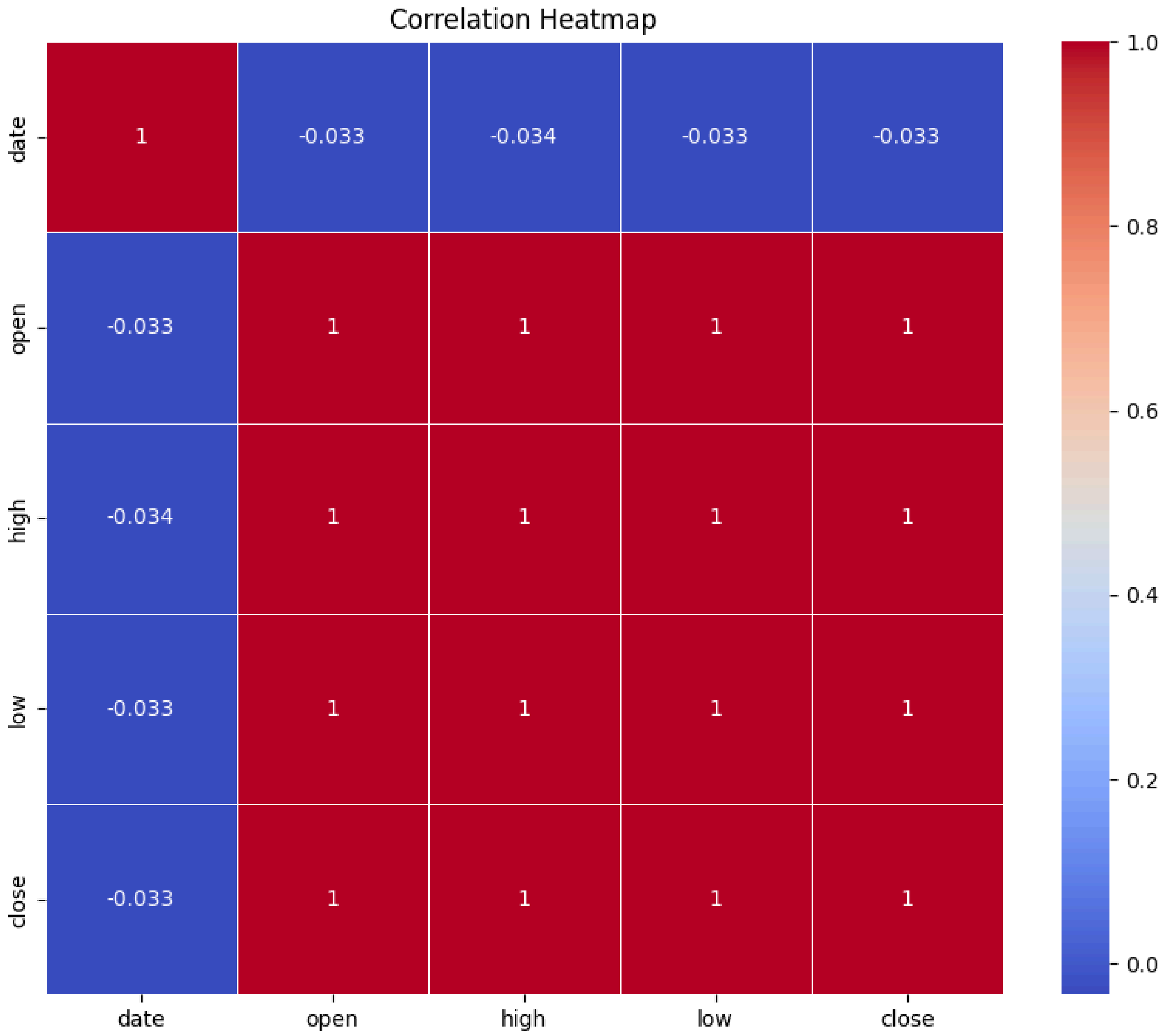
- 1280 columns with float64 data type (likely containing numerical data).
- 3 columns with int64 data type (likely containing integer data).
- 2 columns with object data type (likely containing strings or mixed data).



```
data.describe()
```

	open	high	low	close
count	7781.000000	7781.000000	7781.000000	7781.000000
mean	34.990220	35.655999	34.301243	34.964414
std	99.841502	101.451058	98.073945	99.790823
min	0.410000	0.435000	0.405000	0.408000
25%	4.050000	4.130000	3.980000	4.030000
50%	10.080000	10.110000	10.005000	10.080000
75%	24.350000	24.500000	24.080000	24.250000
max	795.739990	799.359985	784.960022	797.489990

- **count**: Number of non-null values in each column.
- **mean**: Average value of each column.
- **std**: Standard deviation, which measures the dispersion or spread of the values.
- **min**: Minimum value in each column.
- **25%**: First quartile, also known as the 25th percentile.
- **50%**: Median, also known as the 50th percentile or the second quartile.
- **75%**: Third quartile, also known as the 75th percentile.
- **max**: Maximum value in each column.



Correlation between the features considered for stock market prediction

Stock Market Prediction

For effective and computationally affordable stock market prediction, Random forest is used. Random Forest is a popular choice for stock market prediction for several reasons:

1. Ensemble Method: Random Forest is an ensemble learning method that combines the predictions of multiple decision trees. This ensemble approach tends to produce more robust and accurate predictions compared to individual models, making it well-suited for complex and noisy datasets like those found in stock market data.

2. Non-linearity: Stock market data often exhibits non-linear relationships between input features and the target variable (stock prices or market movements). Decision trees, which

form the basis of Random Forest, are capable of capturing non-linear patterns in the data, allowing the model to effectively learn from the complex interactions between different factors influencing stock market movements.

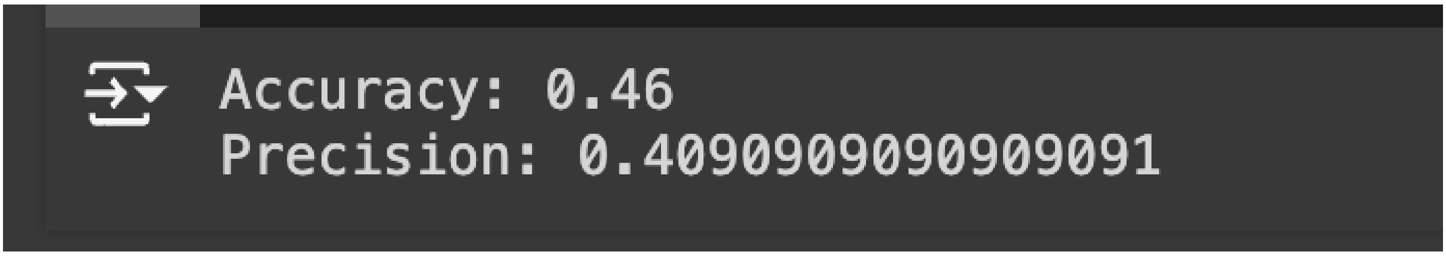
3. Feature Importance: Random Forest provides a built-in feature importance measure, which can help identify the most relevant features for predicting stock market values. This is crucial for understanding which factors drive market dynamics and can aid in feature selection and interpretation.

4. Robustness to Overfitting: Random Forest is less prone to overfitting compared to individual decision trees, thanks to techniques like bootstrap sampling and random feature selection. This helps prevent the model from memorizing noise in the training data and improves its generalization performance on unseen data.

5. Handling High Dimensionality: Stock market datasets often contain a large number of features, including technical indicators, market sentiment data, and economic indicators. Random Forest can handle high-dimensional data efficiently and is capable of capturing the interactions between numerous input variables without requiring extensive feature engineering.

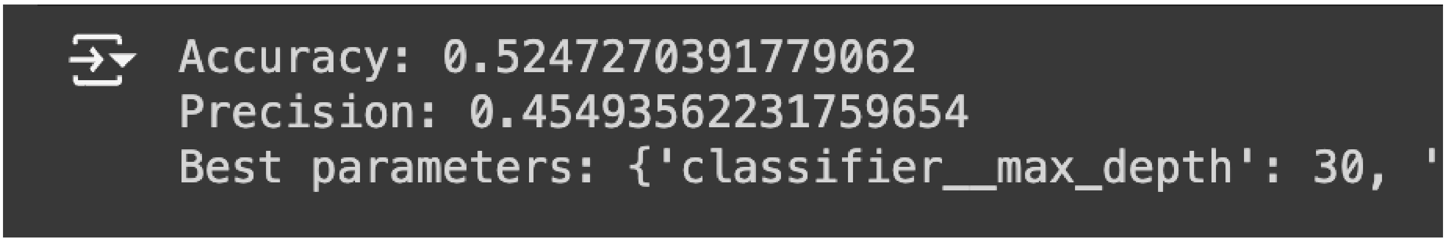
By incorporating a diverse set of features, the Random Forest model can leverage multiple sources of information to make more accurate predictions of future stock market values.

Initial Precision and Accuracy



```
Accuracy: 0.46
Precision: 0.4090909090909091
```

After Hyper parameter Tuning using GridSearchCV



```
Accuracy: 0.5247270391779062
Precision: 0.45493562231759654
Best parameters: {'classifier__max_depth': 30, '
```