Report on ML Project

|  |  |  |
| --- | --- | --- |
| SL NO. | Contents |  |
| 1. | Framing the problem | Defining a problem involves clearly stating and understanding the issue or challenge that needs to be addressed. |
| 2. | Data Gathering | Data gathering is the act of collecting information or facts from different sources |
| 3. | Data Pre­-processing | Data preprocessing involves the preparation and cleaning of raw data before it is used in analysis­ or machine learning. |
| 4. | Exploratory Data Analysis (EDA) | Exploratory Data Analysis (EDA) is the process of visually and statistically analyzing data sets to summarize their main characteristics |
| 5. | Feature Engineering | Feature engineering involves creating new features from existing ones or external data sources to improve model performance and extract meaningful insights. |
| 4. | Model Training/Testing | Training and testing the model against real world data |

**Framing The Problem**

A stock, also known as equity, is a security that represents the ownership of a fraction of the issuing corporation. Units of stock are called shares which entitle the owner to a proportion of the corporation's assets and profits equal to how much stock they own.

Most often, stocks are bought and sold on stock exchanges, such as the Nasdaq or the New York Stock Exchange (NYSE). After a company goes public through an initial public offering (IPO), its stock

available for investors to buy and sell on an exchange. The price of the stock is influenced by supply and demand factors in the market, among other variables.

However, the price of a stock can be influenced by a variety of factors such as market GDP growth, Global Economic Conditions, Market Liquidity etc. Which means that it is difficult to predict the price of a stock after a particular period of time. However, this Is not a simple regression or classification problem as the prices depend on time in general. This means that the model should also be able to relate the price with the time.   
  
We shall hence try to incorporate a sentiment analysis model to find out the emotions or ‘hype’ behind a particular stock which will give better results in prediction of the corresponding bullish and bearish trends.

**Data Gathering**

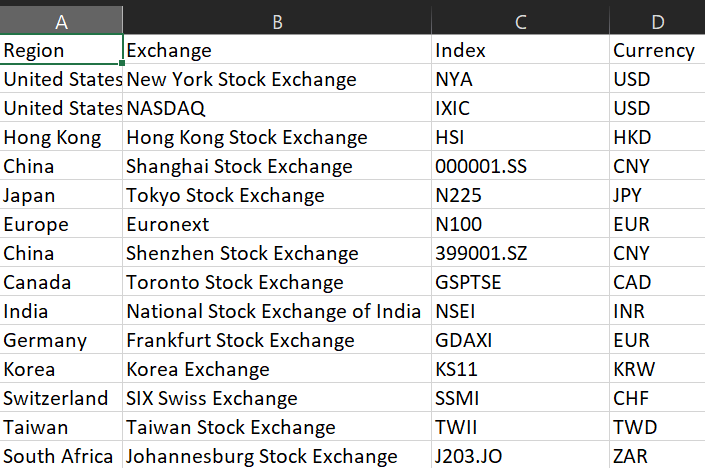
The stock data required for predicting the stock was gathered via kaagle.com. This data contained features such as the High, Low, Prices, datetime etc. The data was later cleaned and sorted using pandas to clear null values and sort the values. The ticker value was used as index for further computations for creating basic graphs and heatmaps for initial evaluation. Later the raw data used was the live data from the stock market

using the yahoo finance API and the pandas data reader module. Specifying the ticker value the API would gather the datetime, High, Open and Low Price of the company in last one day

**Data Preprocessing**

This step is performed before building an ML model using the appropriate analytical technique**.** The following problems were found with the obtained dataset which had to be fixed.

* Dataset obtained had missing values in the Open And High columns which needed to be filled with the mean prices.
* The Dataset also had multiple stocks with different currency units in it which had to be filtered out for individual price prediction.

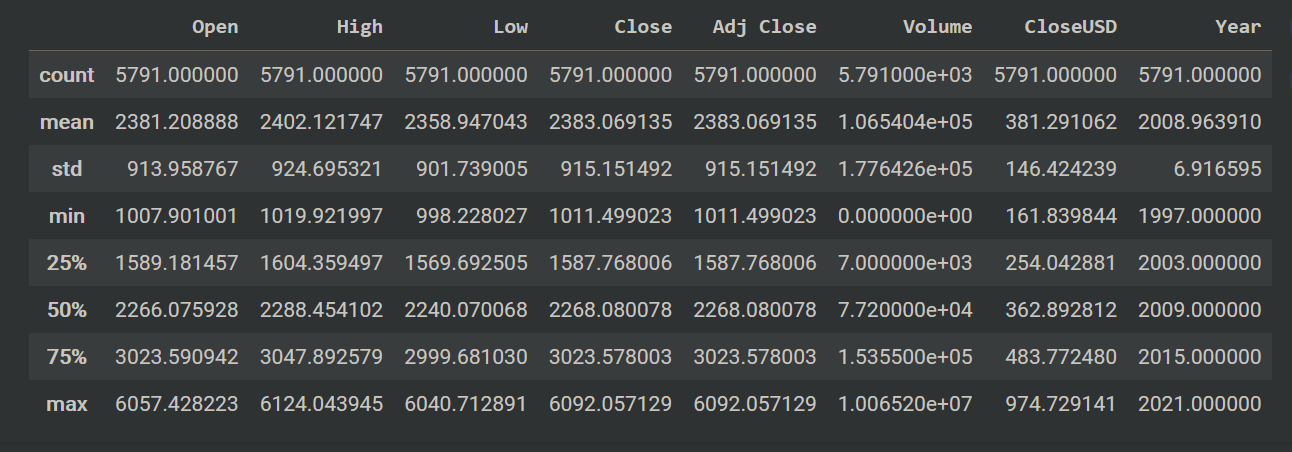


* The Statistics for each index in the data is as follows:

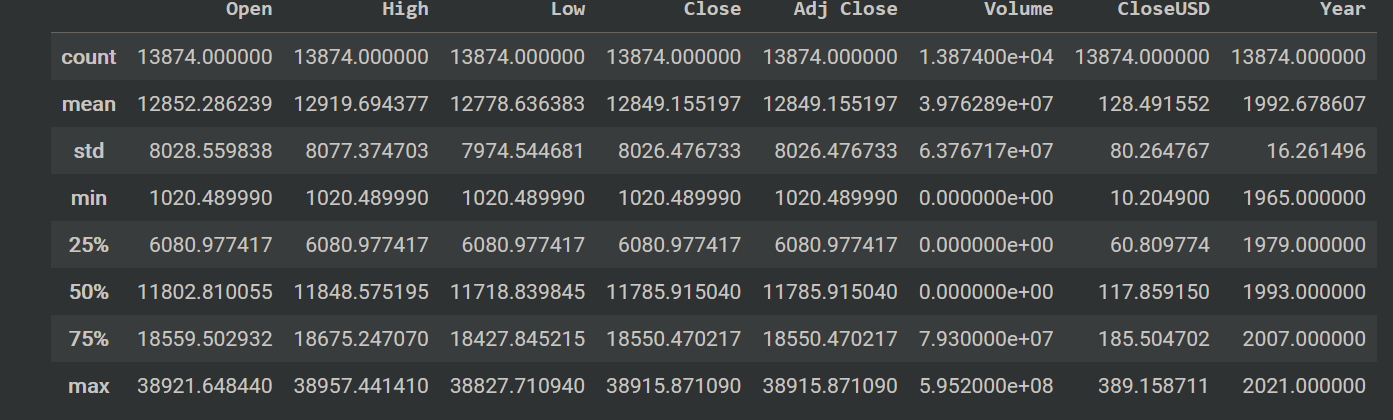
HSI



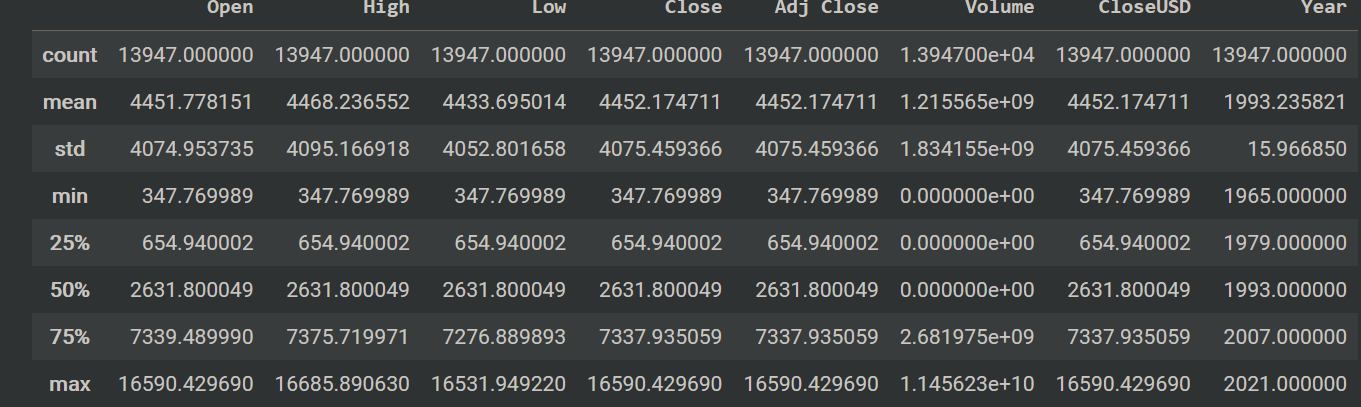
000001.SS



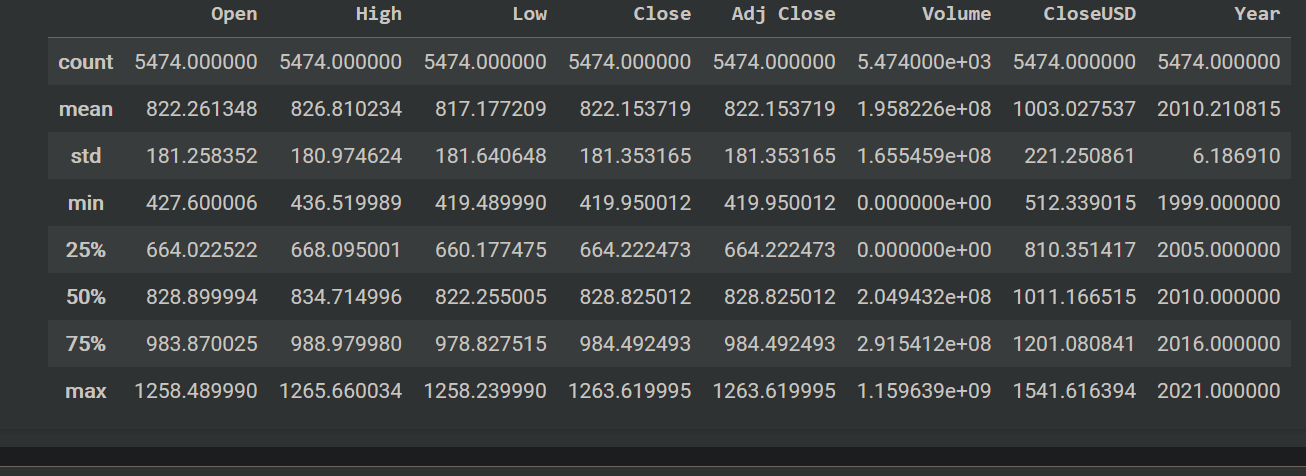
N225



NYA



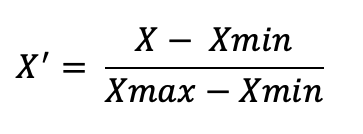
N100



A new Column “year” was added in order to get a general idea on how the price trends move with time.

Before passing the data into the model, it was normalized. Normalization of data is important as it changes all the data to be in the same scale. 1 USD is different from 1 INR in terms of value. However, if the data is not normalized, it may affect the performance of the final model.

The MinMax scaler was used to normalize the dataset. This type of data normalization works by considering the largest value in a column as equal to 1 and the smallest value as equal to 0. The rest of the data in between is transformed into a range between 0 and 1.

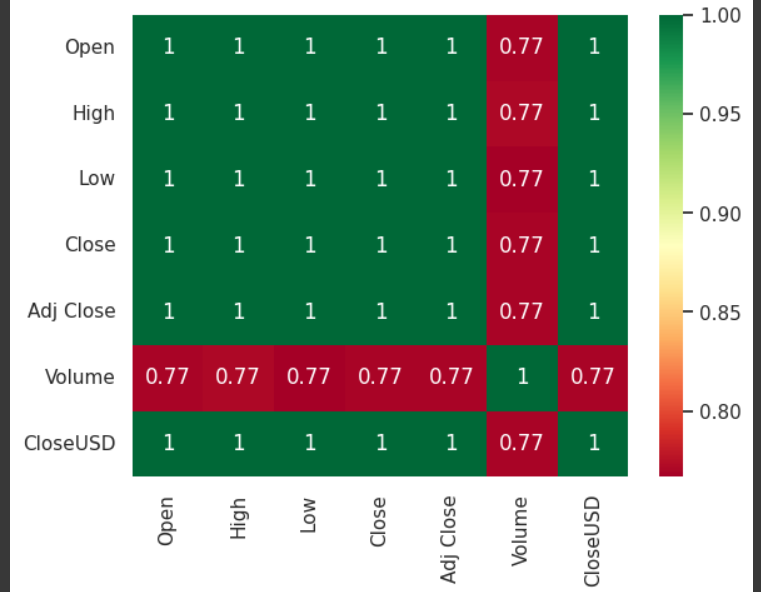


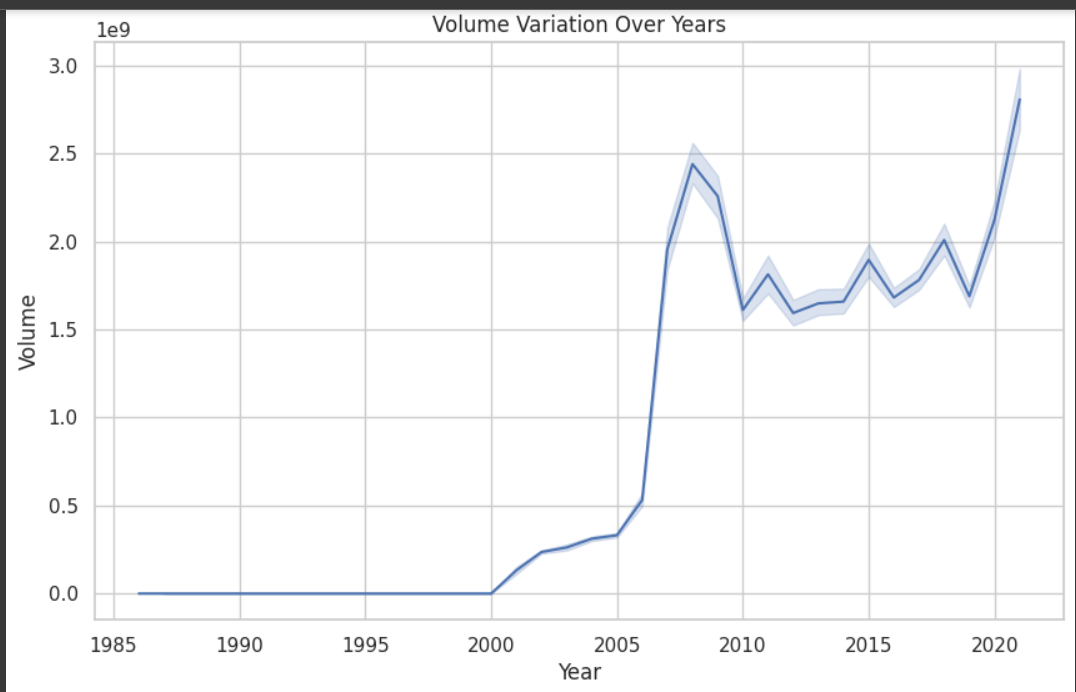
**Exploratory Data Analysis (EDA)**

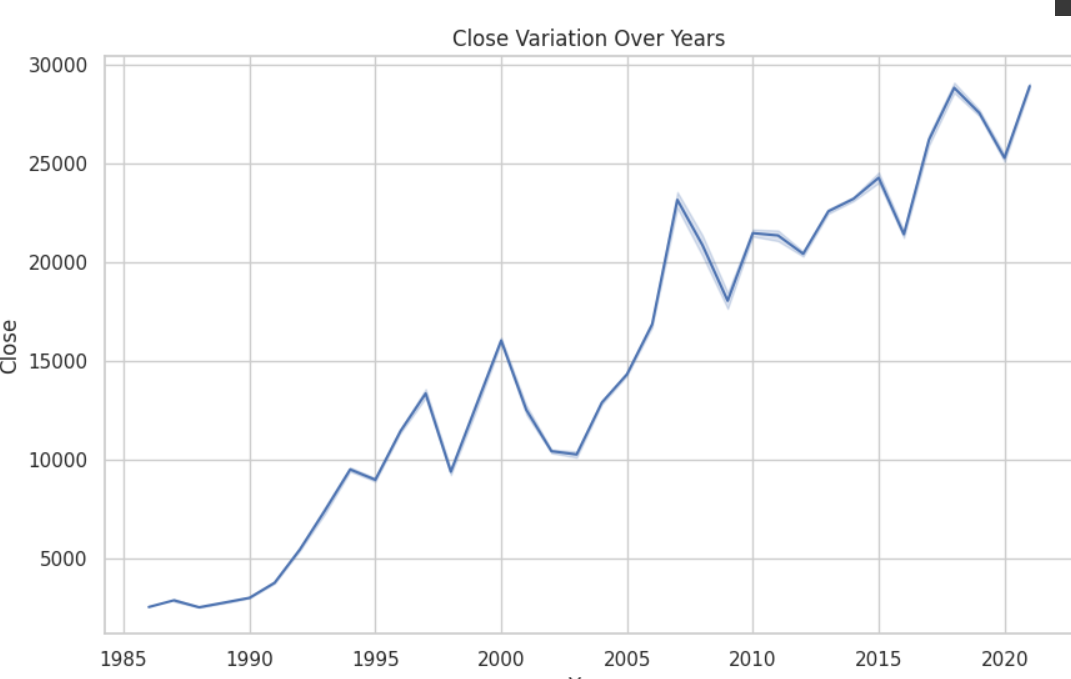
In the exploratory data analysis (EDA) phase of our machine learning project, which centered on predicting stock prices, we strategically incorporated a diverse array of visualization techniques, placing a particular emphasis on heatmaps and various graphs, to distill meaningful insights from the dataset. The correlation heatmap, a cornerstone of our analysis, visually represented the correlation matrix, providing a comprehensive snapshot of the relationships among various stock-related features. This heatmap proved to be a potent tool for discerning both strong and weak correlations, aiding in the identification of potential leading indicators or factors influencing stock price movements. Going beyond this, we leveraged scatter plots and line graphs to vividly illustrate the dynamic relationships between pivotal financial indicators and stock prices over time. This visual exploration offered a nuanced understanding of how fluctuations in these indicators might impact stock prices, thereby influencing our feature selection and engineering strategies.

Regarding the color scheme in the heatmap, typically, red and green colors are used to represent positive and negative correlations, respectively. Darker shades of these colors signify stronger correlations, while lighter shades indicate weaker correlations. The use of color in heatmaps provides an intuitive way to interpret the strength and direction of relationships between variables, facilitating a quicker and more effective understanding of the data. In our specific context, the red and green colors in the correlation heatmap aided us in identifying the nature and significance of correlations among various stock-related features, crucial for informing our predictive modeling framework.

Here is the heatmap and various graphs of HSI index in our project







**Feature Engineering**

Feature engineering is a crucial aspect of data preprocessing in machine learning and data analysis. It involves creating new features from existing ones or external data sources to improve model performance and extract meaningful insights. feature engineering plays a vital role in constructing predictive models for stock price movements or market trends

In our project, we incorporated several new features to enhance the predictive power of our model. These features include Relative Strength Index (RSI), Exponential Moving Average (EMA), and sentiment analysis scores obtained from the Google News API.

Relative Strength Index (RSI):

RSI is a momentum oscillator that measures the speed and change of price movements. It oscillates between 0 and 100 and is typically used to identify overbought or oversold conditions in a stock or market. To calculate RSI, we utilized historical price data and applied a formula to compute the relative strength based on the average gains and losses over a specified period. This feature provides valuable information about the momentum of the stock, helping us identify potential trend reversals or continuation patterns.

Exponential Moving Average (EMA):

EMA is a type of moving average that places more weight on recent data points, making it more responsive to price fluctuations compared to simple moving averages. By incorporating EMA as a feature, we capture the short-term trends in stock prices. EMA is calculated using a smoothing factor that exponentially decreases as we move further back in time. This feature helps us identify the direction and strength of the current price trend, aiding in making more informed trading decisions.

Sentiment

Sentiment analysis involves determining the sentiment or tone of textual data, such as news articles or social media posts. We utilized the Google News API to fetch recent news articles related to the stocks or markets of interest. We then applied natural language processing techniques to analyze the sentiment expressed in these articles. By quantifying the sentiment as a numerical score (positive, negative, or neutral), we converted qualitative textual data into a quantitative feature that can be used in our predictive models. Sentiment analysis provides valuable insights into market sentiment, investor sentiment, and potential catalysts driving stock price movements.

Model Training/ Testing

A customized version of the BERT model called Fin BERT was created with sentiment analysis in the financial industry in mind. Understanding investor sentiment, market sentiment, and forecasting market movements all depend on sentiment analysis. Fin BERT makes use of the pre-trained BERT model, which is good at capturing contextual data. It is optimized for financial text data, taking into account linguistic quirks unique to the financial industry. Using a sizable dataset of financial documents as training material, Fin BERT learns to correlate sentiment labels with words, phrases, and sentences. Fin BERT can reliably categorize financial text after it has been taught, opening up possibilities for risk assessment, trading techniques, financial news summary, and market sentiment analysis. Making educated decisions based on market sentiment signals is made possible by Fin Bert’s precise sentiment classification capabilities.

Recurrent neural network (RNN) architecture known as Long Short-Term Memory (LSTM) was developed to solve the vanishing gradient issue that arises in conventional RNNs while striving to learn long-range dependencies in sequential input. Long short-term memory (LSTM) has grown to be an essential component of many contemporary deep learning architectures, especially for tasks involving sequential input, such as natural language processing, speech recognition, time series prediction, and more. To store information over lengthy sequences and recognize and retain long-term dependencies in the data, LSTM networks need specialized memory cells. The input gates, forget gates, and output gates comprise the three primary parts of these memory cells. A sigmoid activation function governs each gate and controls the amount of information that passes through the cell.