Phase 2

An Explanation of Stock Price Prediction :-

- Stock price prediction is the process of using various analytical and forecasting techniques to estimate the future price movement of a particular stock. Investors and traders seek to predict whether a stock's price will go up or down to make informed investment decisions. This prediction is based on factors such as historical price patterns, technical analysis, fundamental analysis of the company's financial health, and external market influences.

- It's important to note that stock price prediction is inherently uncertain and carries risks because the stock market is influenced by a multitude of unpredictable factors. Predictive models and tools can provide insights, but they cannot guarantee accurate future prices, and investment decisions should be made with caution and based on a comprehensive analysis of available information.

About the Datasets :-

- I got this dataset from www.Kaggle.com
- Every data set I found to predict a stock price (investing) aims to find the price for the next day, and only for that stock.But in practical terms, people like to find the best stocks to buy from an index and wait a few days hoping to get an increase in the price of this investment.
- Rows are grouped by companies and their age (newest to oldest) on a common date.

  The first column is the company. The following are the age, market, date (separated by year, month, day, hour, minute), share volume, various traditional prices of that share (close, open, high…), some price and volume statistics and target.

The target is mainly defined as 1 when the closing price increases by at least 5% in 5 days (open market days). The target is 0 in any other case.

Details About the columns :-

- The Stock Price Prediction Dataset on Kaggle contains historical stock prices of various companies. The Dataset is available in a CSV format and contains 7 columns :

1. **Date**:

   The date on which the stock price was recorded.

2. **Open**:

   The opening price of the stock on that day.

3. **High**:

The highest price of the stock on that day.

4. **Low**:

   The lowest price of the stock on that day.

5. **Close**:

   The closing price of the stock on that day.

6. **Adj Close**:

   The adjusted closing price of the stock on that day.

7. **Volume**:

   The volume of stocks traded on that day.

Libraries to be used here :-

To perform stock price prediction, you can use various libraries in Python. Some of the popular libraries are:

1.NumPy:

It is a library for the Python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays

2. Pandas:

It is a fast, powerful, flexible, and easy to use open source data analysis and manipulation tool.

3. Matplotlib:

It is a plotting library for the Python programming language and its numerical mathematics extension NumPy

4. Scikit-learn:

It is a free software machine learning library for the Python programming language

5. Statsmodels:

it is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

- To install these libraries, I used pip command in my terminal or command prompt. For example, to install NumPy, I run pip install Numpy in my terminal or command prompt. Similarly, I install other libraries by replacing numpy with the name of the library I want to install.

The way to train and test :-

To train and test a machine learning model, we generally divide the original dataset into training data and test data The training data is used to train the model, while the test data is used to evaluate the performance of the model Here are the steps to train and test a machine learning model:

1. **Split the dataset:**
   Split the original dataset into two subsets: training data and test data. The training data should be larger than the test data.

2. **Train the model:**
   Train the machine learning model using the training data.

3. **Test the model:**
   Test the machine learning model using the test data.

To split the dataset, you can use libraries such as Scikit-learn Once you have split the dataset, you can use libraries such as TensorFlow or Keras to train and test your machine learning model.

Examples:

```
model.fit(X_train, y_train)


testing_predictions = model.predict(X_test)
```

Explanation of my topic :-

- Stock price prediction is the process of using machine learning algorithms to forecast the future price of a stock. The prediction is based on historical data, which is used to train the model. The model then uses this training data to make predictions about future stock prices.
- There are many different machine learning algorithms that can be used for stock price prediction, including linear regression, decision trees, and neural networks. These algorithms work by analyzing patterns in the historical data and using these patterns to make predictions about future prices.
- One example of a machine learning algorithm that can be used for stock price prediction is **XGBoost**. This algorithm uses a technique called gradient boosting to create an ensemble of decision trees that can be used to make predictions about future stock prices.
- To predict stock prices using machine learning, you need to have access to historical data for the stock you want to predict. You can then use this data to train your machine learning model and make predictions about future prices.

- It’s important to note that stock price prediction is not an exact science, and there are many factors that can influence the price of a stock. While machine learning algorithms can be helpful in making predictions, they should not be relied upon as the sole source of information when making investment decisions.

Measures used for accuracy checks :-

Accuracy is a common metric used to check the performance of classification models. It measures the ratio of correctly predicted instances to the total instances in the dataset.However, depending on the specific problem and goals, other metrics may be more informative, such as:

- **Precision:**
  Measures the ratio of true positive predictions to the total positive predictions. It's useful when false positives are costly.

- **Recall (Sensitivity):**
  Measures the ratio of true positive predictions to the total actual positives. It's useful when false negatives are costly.

- **F1 Score:**

A combination of precision and recall, balancing both metrics. It's useful when there is an uneven class distribution.

- **Specificity:**
  Measures the ratio of true negatives to the total actual negatives. It's useful for imbalanced datasets.

- **ROC AUC (Receiver Operating Characteristic - Area Under the Curve):**
  It evaluates the model's ability to distinguish between classes, particularly in binary classification problems.

- **Mean Absolute Error (MAE) and Mean Squared Error (MSE):**
  Commonly used for regression tasks to measure the difference between predicted and actual values.

The choice of metrics depends on the nature of the problem and the trade-offs you want to make between different types of errors (e.g., false positives vs. false negatives). It's important to select metrics that align with your specific goals and the characteristics of your data.