# Stock Price Prediction

## Team no. 9

**Problem:** Predicting the future price of a stock or financial asset accurately is a complex and challenging task due to various factors, including market dynamics, economic indicators, and investor sentiment.

**Objective**: Develop a stock price prediction system that leverages historical data, market trends, and machine learning to provide accurate short-term and long-term stock price forecasts for a given company’s stock.

**Scope**: The system will focus on predicting stock prices for a specific set of companies or assets, and it will provide users with forecasts to support their investment decisions.

Design Thinking Process:

**1. Empathize**: Understand the needs and pain points of investors, traders, and analysts who rely on stock price predictions. Gather insights through interviews, surveys, and market research.

**2. Define**: Clearly define the problem statement, objectives, and success criteria for the stock price prediction system. Set specific goals for prediction accuracy and usability.

**3. Ideate**: Brainstorm potential solutions, considering various data sources, algorithms, and technologies. Explore how user-friendly interfaces can present predictions effectively.

**4. Prototype**: Create a prototype or mockup of the system’s user interface and demonstrate how predictions will be delivered to users. Collect feedback from stakeholders.

**5. Test**: Test the prototype with users to gather feedback and refine the design. Ensure that the system addresses the identified needs and concerns of users.

Phases of Development:

**1. Install Dependencies:**

* Common dependencies may include Python programming language , libraries such as scikit-learn and data sources.
* Install the required dependencies such as numpy, pandas and matplotlib.pyplot using package managers like pip for Python.

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error

**2. Data Collection and Preprocessing:**

- Gather historical stock price data, financial indicators, and relevant news sentiment data.

- Clean, normalize, and preprocess the data to make it suitable for machine learning.

**3. Feature Engineering:**

- Create meaningful features, such as moving averages, price-to-earnings ratios, and technical indicators.

- Select relevant features and discard irrelevant ones to improve prediction accuracy.

**4. Model Development:**

- Choose machine learning algorithms, such as regression models, time series forecasting models, or deep learning models.

- Train and fine-tune the selected models using historical data.

**5. Validation and Testing:**

- Split the data into training, validation, and test sets.

- Evaluate the model’s performance using various metrics like Mean Absolute Error (MAE) or Root Mean Square Error (RMSE).

**6. Deployment:**

- Deploy the stock price prediction system to a user-friendly platform, such as a web application or mobile app.

- Ensure the system is accessible to users and provides real-time or scheduled predictions.

**7. Monitoring and Maintenance:**

- Continuously monitor the system’s performance and retrain the model as new data becomes available.

- Address issues and bugs, and improve the system based on user feedback.

**8. Feedback Loop and Iteration:**

- Collect feedback from users on the accuracy of predictions and user interface design.

- Iterate on the system, making improvements and adjustments based on user needs and market changes.

This development process combines design thinking principles with a structured approach to creating a stock price prediction system that meets user needs, is technically robust, and evolves with changing market conditions.

To create a stock price prediction model, you would typically work with a stock price dataset, preprocess the data to make it suitable for analysis, and extract relevant features. Here’s an overview of the stock price dataset, data preprocessing steps, and feature extraction techniques:

Stock Price Dataset:

**Source**: Stock price data is obtained from Microsoft Corporation (MSFT) Stocks from 13/3/1986 To 8/1/2020.

**Attributes**: The dataset typically includes attributes such as date, opening price, closing price, high and low prices, trading volume, and possibly additional financial indicators like dividend yield or earnings per share.

**Frequency**: Stock price data can be collected at various frequencies, such as daily, hourly, or minute-by-minute, depending on the analysis needs.

Data Preprocessing Steps:

**1. Data Cleaning:**

- Remove missing or duplicate data points.

- Handle outliers, which can be common in financial data and can distort analysis.

**2. Data Normalization:**

- Normalize the data to ensure that different features are on the same scale. Common techniques include min-max scaling or z-score normalization.

**3. Handling Time Series Data:**

- If working with time series data, make sure the data is sorted chronologically.

- Consider resampling to a consistent time interval (e.g., daily or weekly) if the data is collected at irregular intervals.

**4. Handling Missing Data:**

- Decide on a strategy for handling missing data, such as forward-fill, backward-fill, interpolation, or removal of rows with missing values.

**5. Feature Engineering:**

- Calculate additional features that can be useful for stock price prediction, such as moving averages, relative strength index (RSI), and exponential moving averages (EMA).

- Compute returns and log-returns, which are often used as target variables for prediction models.

Feature Extraction Techniques:

**1. Moving Averages:**

- Simple Moving Average (SMA): Calculate the average of prices over a specified window of time.

- Exponential Moving Average (EMA): Give more weight to recent prices, making it sensitive to recent changes.

**2. Technical Indicators:**

- Relative Strength Index (RSI): Measures the speed and change of price movements, indicating overbought or oversold conditions.

- Stochastic Oscillator: Measures the location of a current price in relation to its price range over a specific period.

- Bollinger Bands: Consist of a middle band (SMA) with upper and lower bands representing standard deviations from the middle band.

**3. Volatility Measures:**

- Historical Volatility: Measures the standard deviation of returns over a specific period.

- Implied Volatility: Derived from option prices and reflects market expectations.

**4. Lagged Features:**

- Include lagged (past) values of the target variable or other features to capture time-dependent patterns.

**5. Sentiment Analysis:**

- Incorporate sentiment scores or news sentiment related to the stock as additional features.

**6. Economic Indicators:**

- Consider adding macroeconomic indicators (e.g., GDP, unemployment rates) that can influence stock prices.

**7. Fundamental Data:**

- Include financial data like earnings per share, price-to-earnings ratios, and dividend yield as potential features.

The choice of preprocessing steps and feature extraction techniques can vary depending on the specific stock price prediction problem and the machine learning models being used. Experimentation and feature selection are often required to identify the most relevant features for accurate predictions.

The choice of machine learning algorithm, model training, and evaluation metrics for a stock price prediction problem statement is a critical decision in building an effective prediction model. Here’s a detailed explanation of these aspects:

**1. Choice of Machine Learning Algorithm:**

* Regression Algorithms: Stock price prediction is essentially a regression problem, where you aim to predict a continuous numerical value (stock price).
* **Linear Regression**: A simple and interpretable choice that models the relationship between independent variables (features) and the target (stock price) with a linear equation.

**2. Model Training:**

**Data Splitting**: Divide your dataset into training, validation, and test sets. The training set is used to train the model, the validation set is used to fine-tune hyperparameters, and the test set is used to evaluate the model’s performance.

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2,random\_state=42)

**3. Evaluation Metric:**

**Mean Squared Error (MSE):** MSE measures the average squared difference between predictions and actual values. It penalizes larger errors more than MAE.

mse = mean\_squared\_error(y\_test, y\_pred)

The choice of algorithm and evaluation metrics depends on the specific characteristics of the stock price prediction problem, such as the time horizon (short-term or long-term), the presence of noisy data, and the level of predictability in the market. It’s important to experiment with different approaches and rigorously evaluate model performance to select the most effective solution.

Data visualization:

Data visualization is the process of representing data graphically to help people understand, interpret, and make sense of data. It involves creating visual representations of data, which can include charts, graphs, maps, and other graphical elements. Data visualization is a powerful tool for conveying complex information, identifying patterns, trends, and outliers, and making data-driven decisions.

When creating data visualizations, consider the audience, the purpose of the visualization, and the characteristics of the data you're working with. Choose the appropriate type of visualization that effectively conveys the message you want to communicate. Additionally, use colors, labels, and annotations to enhance the clarity of your visualizations and ensure they provide valuable insights.

