IBM Applied Data Science Project

Stock Price Prediction

PROBLEM DEFINITION:

The primary goal of this project is to create a predictive model for stock price forecasting using

historical market data. Specifically, we aim to predict the future stock prices of Microsoft (MSFT) to

help investors make more informed decisions and potentially improve their investment strategies.

DESIGN THINKING:

1. Data Collection:

- Collect historical stock market data for Microsoft from the provided dataset.
- Key features include date, open price, close price, volume, and other relevant indicators.

2. Data Preprocessing:

- Clean the data: Handle missing values, outliers, and inconsistencies.
- Feature selection: Identify the most relevant features for forecasting.
- Data transformation: Convert categorical features (if any) into numerical representations.
- Time series data formatting: Ensure data is organized by date and sorted chronologically.

3. Feature Engineering:

- Create additional features that can enhance the predictive power of the model.
- Moving averages (e.g., 7-day, 30-day, etc.)
- Technical indicators (e.g., RSI, MACD, Bollinger Bands)
- Lagged variables (using past stock prices and indicators as features)

4. Model Selection:

Choose appropriate time series forecasting algorithms for stock price prediction.

Potential choices include:

- ARIMA (Auto Regressive Integrated Moving Average)
- LSTM (Long Short-Term Memory)
- Other machine learning models like Random Forest, XG Boost (for comparison)

5. Model Training:

- Split the data into training and validation sets, maintaining chronological order.
- Train the selected model(s) using the preprocessed data.

6. Evaluation:

Assess the model performance using relevant time series forecasting metrics, such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and possibly others.

Compare the performance of different models to select the best-performing one.

Stock Price Prediction Analysis Approach

1. Problem Understanding:

Clearly define the problem you aim to solve: Predict future stock prices.

Understand the business context: What are the objectives and constraints of the project?

Define the scope: Which stocks or indices are you forecasting? What is the prediction horizon (e.g., daily, weekly, monthly)?

2. Data Collection:

Identify and gather historical stock market data from reliable sources (e.g., Yahoo Finance, Alpha Vantage).

Collect a diverse set of features, including price, volume, and potentially external factors (e.g., economic indicators, news sentiment) that might impact stock prices.

3. Data Exploration and Preprocessing:

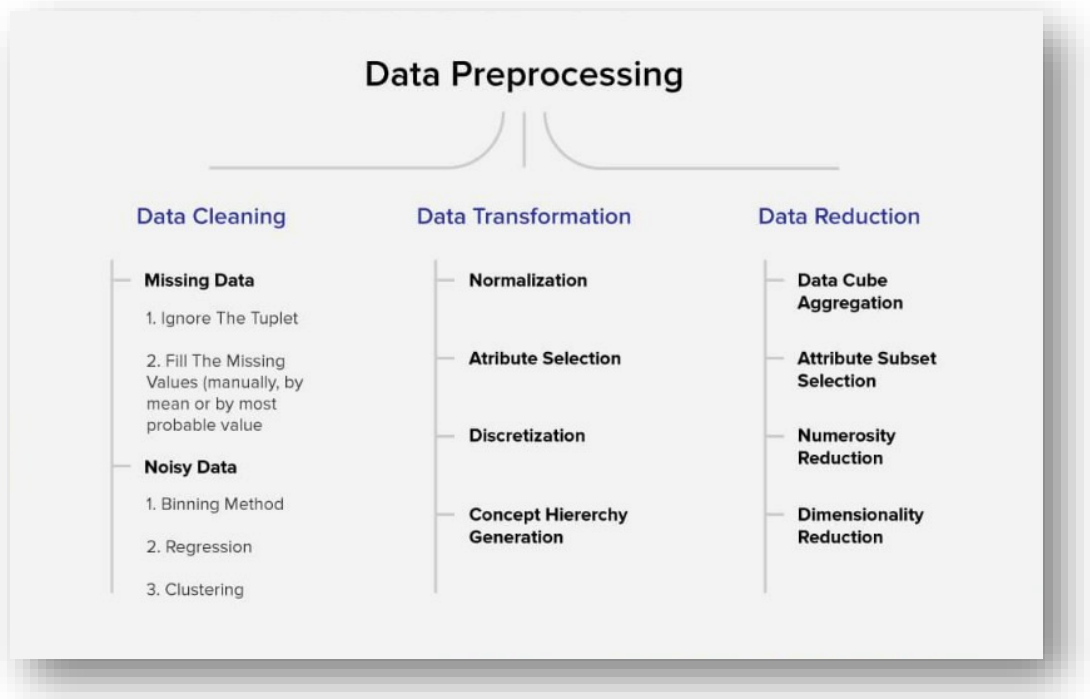
Perform initial data exploration to understand the dataset structure and quality.

Check for missing data, outliers, and data integrity issues.

Visualize time series data to identify trends, seasonality, and patterns.

Convert date columns into datetime objects and set them as the index.

Calculate returns or log-returns, as these are often used in stock price prediction.



4. Feature Engineering:

Create additional relevant features to improve model performance, such as moving averages, technical indicators (e.g., RSI, MACD), and lagged variables.

Consider encoding categorical variables or sentiment scores.

Normalize or scale features to ensure they have similar ranges.

5. Data Splitting:

Split the dataset into training, validation, and test sets.

Ensure that the validation and test sets are future timestamps, maintaining a chronological order.

6. Model Selection:

Choose appropriate machine learning or time series forecasting models:

For traditional time series data: ARIMA, SARIMA, Prophet.

For deep learning: LSTM, GRU, or other RNN-based models.

For machine learning: Random Forest, XG Boost, or SVR.

Consider multiple models for improved accuracy.

7. Model Training:

Train the selected models using the training dataset.

Experiment with different hyperparameters and architectures.

Monitor training progress and avoid overfitting.

8. Model Evaluation:

Evaluate model performance using relevant metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and directional accuracy.

Visualize predictions against actual stock prices to assess model fit.

Compare different models and select the best-performing one.

9. Hyperparameter Tuning:

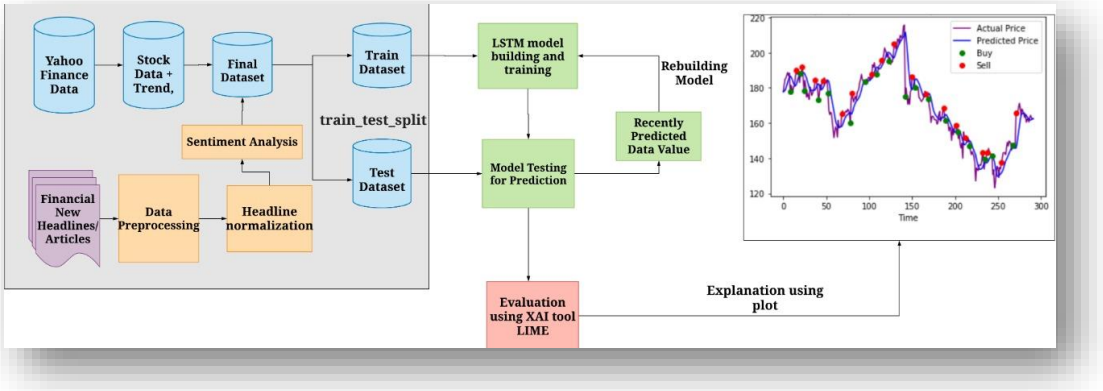
Fine-tune model hyperparameters using techniques like grid search or Bayesian optimization.

Re-train the model with optimized hyperparameters.

10. Model Interpretation:

Understand the importance of features using feature importance scores (e.g., SHAP values for tree-based models).

Explore model predictions to identify patterns and trends.



11. Deployment and Monitoring:

Deploy the trained model in a production environment if applicable.

Set up monitoring to track model performance and retrain as necessary.

12. Reporting and Communication:

Summarize the project in a report or presentation.

Explain the model predictions and limitations.

Provide actionable insights for investors based on the model forecasts.

Communicate results effectively to stakeholders.

13. Maintenance and Updates:

Regularly update the model with new data to keep it relevant.

Continuously monitor model performance and retrain as needed to adapt to changing market conditions.

This analysis approach provides a structured framework for developing and deploying a stock price prediction model, ensuring that we cover essential steps from data collection to model deployment and maintenance.