

ARTIFICIAL INTELLIGENCE

Stock Price Movement Analysis

PROJECT REPORT

BACHELORS IN TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING(AI)



Submitted By:

SHIKHAR MAHESHWARI

CSEAI

202401100300230

TABLE OF CONTENT

S.No.	Topic	Page No.
1.	Introduction	3
2.	Methodology	4
3.	Code	5
4.	Output Screenshots	7
5.	Conclusion	8

INTRODUCTION

Background

Stock price movement analysis is a crucial aspect of financial forecasting that helps investors and traders make informed decisions. Stock prices fluctuate due to various factors, including market trends, economic conditions, and investor sentiment. Understanding these movements can aid in making strategic investment choices.

Objective

The goal of this project is to analyze stock price trends using historical data and various analytical techniques. The dataset consists of daily stock prices, including Open, High, Low, Close, and Volume. The analysis includes moving averages, trend detection, and volume insights to better understand the stock's behavior.

Scope

- **Automated Traffic Data Collection**
- **Real-time Signal Adjustment**
- **AI-based Optimization Algorithm**
- **Simulation of Traffic Light Execution**

METHODOLOGY

The following steps were taken to analyze stock price movements:

- 1) **Data Collection:** The dataset containing stock prices was uploaded and read using Python's Pandas library.
- 2) **Data Preprocessing:** Checked for missing values and formatted the Date column.
- 3) **Exploratory Data Analysis:**
 - ➔ Calculated the highest and lowest stock prices.
 - ➔ Computed moving averages to identify the trends.
 - ➔ Analyzed trading volume to identify unusual activity.
- 4) **Visualization:** Used Matplotlib to plot stock trends and volume variations.

CODE

```
import pandas as pd

import matplotlib.pyplot as plt

# Load dataset

file_path = "/content/stock_data.csv"


df = pd.read_csv(file_path)

# Convert 'Date' column to datetime

df['Date'] = pd.to_datetime(df['Date'])

# Display first few rows

print(df.head())

# Moving Average Calculation 
```

```

df['SMA_10'] = df['Close'].rolling(window=10).mean()

# Filter dataset to avoid NaN values at start
df_valid = df.dropna()

# Create figure with two subplots
fig, ax1 = plt.subplots(figsize=(12,6))

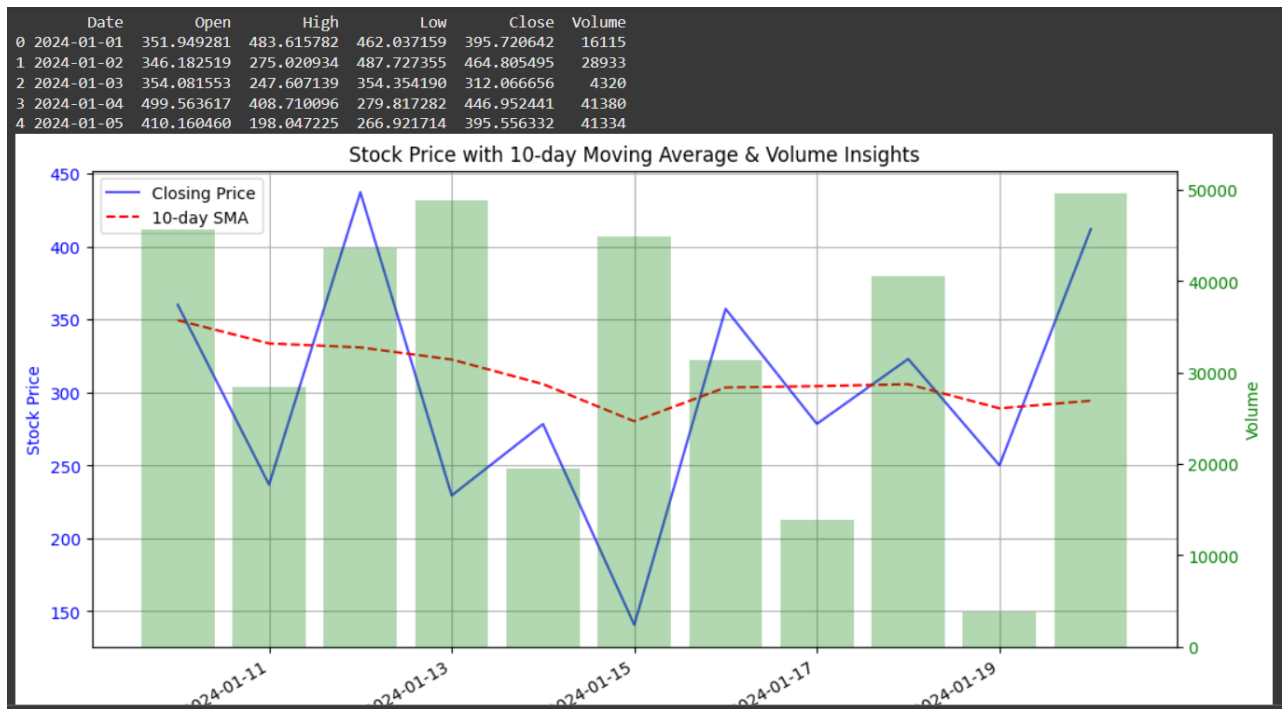
# Plot Closing Price & 10-day SMA on primary axis
ax1.set_xlabel("Date")
ax1.set_ylabel("Stock Price", color='blue')
ax1.plot(df_valid['Date'], df_valid['Close'], label="Closing Price", color='blue', alpha=0.7)
ax1.plot(df_valid['Date'], df_valid['SMA_10'], label="10-day SMA", color='red', linestyle="dashed")
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc="upper left")
ax1.grid(True)

# Create secondary axis for Volume
ax2 = ax1.twinx()
ax2.set_ylabel("Volume", color='green')
ax2.bar(df_valid['Date'], df_valid['Volume'], color='green', alpha=0.3, label="Volume")
ax2.tick_params(axis='y', labelcolor='green')

# Title & show plot
plt.title("Stock Price with 10-day Moving Average & Volume Insights")
fig.autofmt_xdate() # Rotate date labels
plt.show()

```

OUTPUT SCREENSHOT



CONCLUSION

The stock price movement analysis provided valuable insights into market trends. The findings indicate how stock prices fluctuate based on historical trends and volume activity. By incorporating moving averages and visualizing price movements, traders can make better investment decisions. Future enhancements may include applying machine learning models for predictive analysis and incorporating more advanced technical indicators.

Future Enhancements

To further improve stock price movement analysis, the following enhancements can be considered:

- **Machine Learning Models:** Implement predictive models such as Linear Regression, Random Forest, or LSTMs to forecast future stock prices.

- **Sentiment Analysis:** Integrate AI-based sentiment analysis from news articles and social media to assess market mood.
- **Advanced Technical Indicators:** Use Bollinger Bands, MACD, and RSI for more in-depth trend analysis.
- **Automated Trading System:** Develop AI-based trading bots to make real-time investment decisions.
- **Big Data Integration:** Leverage large-scale financial data sources for better accuracy in predictions.

References/Credits

- Stock dataset source: [TRADING VIEW]
- Pandas Documentation: <https://pandas.pydata.org/>
- Matplotlib Documentation: <https://matplotlib.org/>