In this project, I have utilized several Python packages:

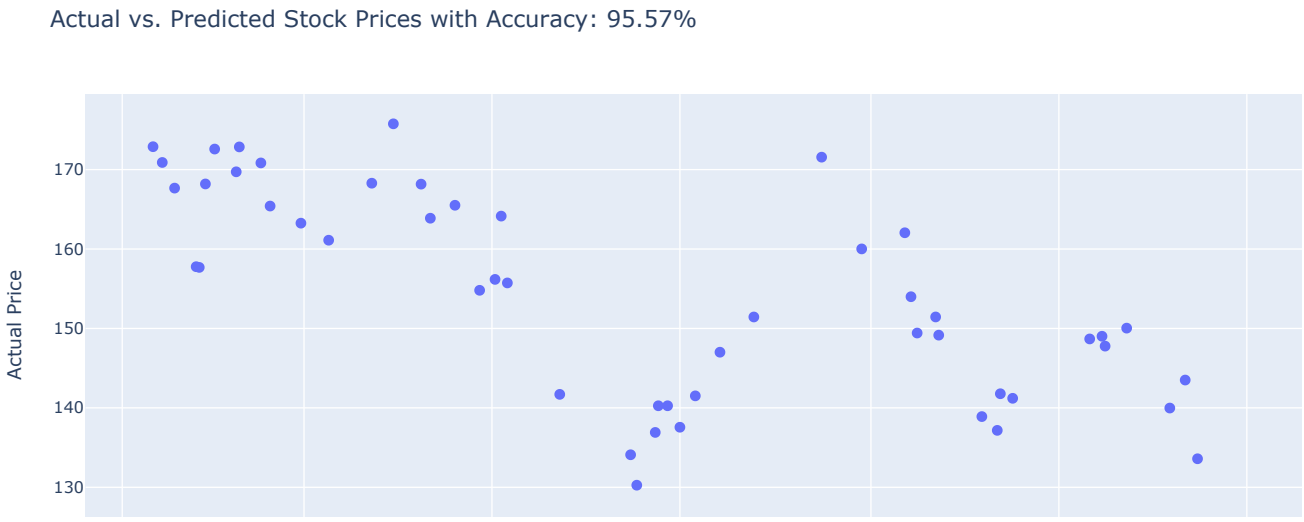
Pandas: Used for managing and structuring data. Pandas provides the DataFrame data structure, which facilitates data manipulation and analysis.

yfinance: This Python package allows fetching historical stock market data, enabling various analyses based on stock market trends and patterns.

scikit-learn (sklearn): Employed for performing linear regression and analyzing data. Scikit-learn offers a wide range of machine learning algorithms, including regression models, for predictive modeling and analysis tasks.

Plotly Express: Chosen for plotting interactive visualizations. Plotly Express provides a high-level interface for creating various types of plots, including scatter plots, line plots, and bar charts. Its interactive capabilities enhance data exploration and presentation.

```
import yfinance as yf
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
import plotly.express as px


tickerSymbol = 'AAPL'
tickerData = yf.Ticker(tickerSymbol)
data = tickerData.history(period='1d', start='2022-1-1', end='2023-01-01')


X = data[['Volume', 'Open']]
Y = data['Close']

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, random_state= 42)



model = LinearRegression()

model.fit(X_train, Y_train)

Y_pred = model.predict(X_test)



accuracy = r2_score(Y_test, Y_pred)

scatter_data = pd.DataFrame({'Date': Y_test.index, 'Actual Price': Y_test.values, 'Predicted Price': Y_pred})

fig = px.scatter(scatter_data, x='Date', y='Actual Price', hover_data=['Predicted Price'],
                 title=f'Actual vs. Predicted Stock Prices with Accuracy: {accuracy:.2%}')
fig.update_traces(mode='markers', marker=dict(size=8))
fig.show()
```


Actual vs. Predicted Stock Prices with Accuracy: 95.57%
Actual Price
170
160
150
140
130