Stock Price Prediction using Python

Description:

This Python script demonstrates stock price prediction using historical data from Yahoo Finance. It uses features like Open, High, Low, and Volume to predict the next day's closing price. The Linear Regression model is trained and evaluated using Mean Squared Error and R^2 score, and the predicted vs actual closing prices are visualized using matplotlib.

Python Code:

```
# Import libraries
import yfinance as yf
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt

# 1. Select and download stock
stock = 'AAPL'
data = yf.download(stock, start='2020-01-01', end='2023-12-31')
data.reset_index(inplace=True)

# 2. Prepare features and target
data['Next_Close'] = data['Close'].shift(-1)
df = data[['Open', 'High', 'Low', 'Volume', 'Next_Close']].dropna()
X = df[['Open', 'High', 'Low', 'Volume']]
y = df['Next_Close']

# 3. Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)

# 4. Train model
model = LinearRegression()
# model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# 5. Predictions and evaluation
y_pred = model.predict(X_test)
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
print("R^2 Score:", r2_score(y_test, y_pred))

# 6. Plot Actual vs Predicted
plt.figure(figsize=(12,6))
plt.plot(y_test.values, label='Actual Closing Price', color='blue')
plt.plot(y_pred, label='Predicted Closing Price', color='red')
plt.title(f"{stock} Stock: Actual vs Predicted Closing Prices")
plt.xlabel('Days')
plt.ylabel('Price (USD)')
plt.legend()
plt.tight_layout()
plt.show()
```