NVIDIA Stock Price Prediction using ARIMA & LSTM

Project Description

This project uses historical NVIDIA (NVDA) stock data to build and evaluate time series forecasting models. Two different approaches are applied: ARIMA (AutoRegressive Integrated Moving Average) and LSTM (Long Short-Term Memory) neural networks. The goal is to predict future stock prices and compare model performance.

Features

- Downloads NVIDIA stock data using yfinance
- Data cleaning and preprocessing
- Visualizes historical stock prices
- Implements ARIMA for statistical forecasting
- Implements LSTM for deep learning-based forecasting
- Plots predicted vs actual stock prices
- Calculates and displays Mean Squared Error (MSE) for model evaluation

Installation & Requirements

Make sure you have Python 3.7 or higher installed.

Install the required Python libraries:

```
pip install yfinance numpy pandas matplotlib statsmodels scikit-learn tensorflow
```

Usage

1. Clone or download this repository.
2. Place the `NVDA_stock_data.csv` in the `data/` folder (or let the script download it).
3. Run `Main.py`:

```
python Main.py
```

4. The script will visualize stock prices, train ARIMA and LSTM models, and plot the predictions.

How it Works

- ARIMA: A statistical model that captures autocorrelations in time series data.
- LSTM: A type of Recurrent Neural Network capable of learning long-term dependencies in sequence data.

Both models are trained on 80% of the data and tested on the remaining 20%.

Outputs

The project generates several plots:

- NVDA Stock Closing Price Over Time
- ARIMA Predicted vs Actual Prices
- LSTM Predicted vs Actual Prices

It also prints the Mean Squared Error (MSE) for both models in the console.

License