

README

Trading Algorithm for Stock Forecasting and Trading

This project implements a trading algorithm using four different forecasting methods:

1. Moving Average
2. Exponential Smoothing
3. ARIMA (AutoRegressive Integrated Moving Average)
4. LSTM (Long Short-Term Memory networks)

The algorithm fetches stock data for selected tickers, forecasts the stock prices using each method, then trades based on the LSTM forecast. Performance metrics are also calculated for each model and the overall trading strategy.

Prerequisites

To run the code, you need the following libraries:

- `yfinance`: Fetch stock market data from Yahoo Finance.
- `pandas`: For data manipulation.
- `numpy`: For mathematical operations.
- `statsmodels`: For time series forecasting.
- `keras`: For deep learning model LSTM.
- `tensorflow`: Backend for Keras.
- `sklearn`: For data scaling and metrics.
- `matplotlib`: For plotting results.

You can install these libraries using the following:

```
1 | pip install yfinance pandas numpy statsmodels keras tensorflow sklearn matplotlib
```

Running the Code

1. Use Jupiter notebook to open and run this file. We use Google Colab here.

What to Expect?

Upon running the code:

- The algorithm fetches the data for the selected tickers.
- Forecasts the stock prices using each of the four models.

- Plots the true stock prices and forecasted values.
- For each model, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are displayed.
- Trades based on the LSTM's forecast.
- Visualizes buy and sell points for each ticker.
- Displays the final portfolio value, annualized return, and Sharpe ratio.

Parameters

- `tickers`: List of stock tickers to forecast and trade.
- `end_date`: Today's date, the last date of the dataset.
- `test_start_date`: The start date for the test dataset.
- `train_start_date`: The start date for the training dataset.

Note: Adjust the `initial_investment` in the `# Initialize the portfolio` section to simulate with different initial funds.