

Assignment [AtomCamp]

(Time Series Analysis & Forecasting)

Dataset:

Use stock market data from Yahoo Finance (<https://finance.yahoo.com/>) using the yfinance Python package. You may choose any publicly listed company (e.g., AAPL, MSFT, GOOGL, TSLA, etc.). Make sure the dataset spans at least 3–5 years for meaningful analysis.

Objective

This assignment will guide you through a complete time series workflow—from exploratory analysis and classical forecasting using ARIMA to deep learning models using LSTM. You will gain hands-on experience in time series decomposition, moving average analysis, statistical modeling, and sequence-based neural networks.

By the end, you will be able to:

- Perform time series decomposition and trend analysis.
- Forecast stock prices using classical ARIMA models.
- Benchmark them against deep learning models like LSTM.
- Interpret and visualize model predictions effectively.

Tasks

1. Time Series Analysis

- Download historical daily stock prices using yfinance.
- Visualize the Close price over time.
- Create and interpret:
 - Rolling moving averages (7-day, 30-day)
 - Seasonal decomposition of the series (using `seasonal_decompose`)
 - Plot ACF/PACF to understand lags and correlations.
- Comment on stationarity and apply differencing if needed.

2. ARIMA Benchmark

- Fit an appropriate ARIMA or SARIMA model on the Close price.
- Use AIC to guide order selection.
- Forecast for the next 30 days.
- Plot:
 - Forecast with confidence intervals
 - Overlay on true values (if available)
- Report:
 - MAE and RMSE
 - Your reasoning for chosen (p,d,q) values

3. Deep Learning Forecasting with LSTM

- Prepare the dataset for supervised learning using a sliding window approach.

- Split into train/test and scale the data appropriately.
- Build an LSTM model to forecast the next day's Close price:
 - Input: last n days of Close prices
 - Output: next day Close price
- Train and evaluate:
 - MAE and RMSE
 - Plot true vs predicted on test set

4. Visualizations & Interpretation

- Include side-by-side plots comparing ARIMA and LSTM forecasts.
- Reflect in 3–5 lines: which approach performed better and why? - Consider model assumptions, flexibility, and prediction accuracy.

Deliverables

Submit a Jupyter Notebook that includes:

- All code (clean, commented, and organized)
- Plots and charts (time series, rolling averages, forecasts)
- Evaluation metrics
- Interpretation of your findings
- Final reflection comparing ARIMA vs LSTM