**Stock Price and Volume Time Series Analysis Using Times Series Models: Best Model and Stock Price Prediction**

Suvechhya Pokhrel

University of Denver

COMP 4449: Data Science Capstone

Dr. Sada Narayanappa

October 19, 2023

# **Introduction**

**Purpose**

The goal of this project is to develop and implement a predictive model that accurately forecasts stock prices for a diverse portfolio of symbols based on historical data from January 2021 to December 2022. The model aims to provide valuable insights for investment decisions and potentially enhance portfolio performance.

**Significance**

The significance of this report lies in its potential to revolutionize the way individuals manage their portfolios. By developing predictive models for optimizing stock market entry and exit timing, this project aims to provide tools that can help everyday investors make informed and profitable investment decisions. With the growing importance of personal finance and retirement planning, these models have potential to enhance financial security for people.

**Research Question(s)**

Portfolio management: How can predictive models be developed to optimize stock market entry and exit timing?

**Description of the Dataset**

The dataset contains weekly closing prices and trading volume of 11,165 unique ticker symbols. It consists of four columns:

1. ‘Price’ represents the weekly closing stock prices in dollars and serves as the target variable.
2. ‘Symbol’ lists the ticker symbols that represent specific entities, such as stocks or companies.
3. ‘Trading Volume’ represents the weekly trading volume for each entity, measured in thousands.
4. ‘Date’ provides the weekly timestamps, ranging from January 1, 2021, to December 30, 2022, and serves as the independent variable.

# **Data Preprocessing**

**Data Preparation**

First, we consolidated the 105 excel sheets, each representing the data for a different week, into a single data frame with four columns. Subsequently, we conducted a thorough inspection of the data for missing values. We identified 27 rows containing NaN symbol values and dropped them from the data frame.

Further investigation involved assessing the count of unique dates in the ‘Date’ column and inspecting the frequency of data grouped by ‘Symbol’. This analysis revealed the presence of symbols with incomplete records across all dates, prompting their elimination from the dataset. As a result, we retained 6,894 unique symbols for further analysis.

Finally, we changed the data type of the ‘Date’ column to datetime64 to enable efficient time-based analysis and visualization.

**Exploratory Data Analysis**

To further explore the data, and select the symbols for our focus, we initially narrowed down our options by choosing a subset of the top 100 symbols with the highest average trading volume. The average trading volume for each symbol was evaluated by averaging the trading volumes records for 105 weeks. We then employed hierarchical clustering to group similar stocks based on historical price data. The distance threshold for clustering was set at a correlation of 2 between price and trading volume. Subsequently, we randomly selected one symbol from each cluster to streamline our analysis. The hierarchical clustering method generated a total of 23 clusters, and the selected symbols are as follows: 'NLY', 'AUY', 'PSQ', 'SWN', 'SOXS', 'AAPL', 'KMI', 'TSLA', 'SPY', 'NOK', 'FCX', 'CTRM', 'RIG', 'METX', 'SPXS', 'SH', 'TQQQ', 'BBD', 'FXI', 'ABEV', 'VALE', 'CEI', 'ITUB'.

Next, we checked how the features are distributed for the symbols above. Looking at the summary statistics table Figure 1, we observed that TSLA, SPY, and AAPL had the highest average stock price. TSLA had the highest average stock price of $701, while CEI has the lowest price of $0.81. When assessing the trading volume column, we see that TQQQ has the highest average trading volume of 101,533, while SPXS has the lowest average trading volume of 15,860. Consequently, both the price and volume columns exhibit a wide range of data indicating the necessity for data scaling. We also noted that both the price and volume columns contained 105 data points, indicating the absence of missing values and rows.

Time series analysis relies on the assumption of data stationarity. Hence, we performed an Augmented Dickey-Fuller (ADF) test to assess the stationarity of time series data for each symbol. Our test indicated that the data for all symbols were stationary. We also employed Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) to identify the order of AR(AutoRegressive) and MA (Moving Average) components of the ARIMA model. The ACF and PACF function suggested the presence of AR(1) term in our time series models. Additionally, we created time series plots for each symbol to help us detect trends, seasonality, and irregular patterns that aid our modeling decisions. After a comprehensive analysis of the data, we decided to use Auto ARIMA and LSTM time series models for our financial time series forecasting.

**Data Splitting and Scaling**

When working with time series data, it is recommended to use the more recent data for testing and the older data for training. We split the data into training data, consisting of the initial 80% of the records, and testing data, which comprises the most recent 20% of the data. An example plot illustrating the distribution of training and testing data is presented in Figure 2.

Given the substantial differences in scales between our price and volume columns, coupled with extensive min-max range across various symbols, we implemented standard scaling for this dataset.

# **Model Building and Evaluation**

**Model Building**

To address our research question, we employed both an ARIMA model and an LSTM model. For our ARIMA model, we utilized the ‘auto\_arima’ function from the ‘pmdarima.arima’ library. However, the ARIMA model’s predictions failed to capture fluctuations in stock prices. The prediction graph appeared flat and smooth in comparison to the training and test data. Furthermore, the model’s predictions deviated significantly from the actual prices. While the root mean squared error (RMSE) values were low, the relatively small scale of our data indicated the ARIMA model’s performance was less than optimal. It is essential to note that the model might not be providing reliable forecasts. Given our intention to utilize these forecasts for financial and investment decisions, enhancing the model’s accuracy becomes a priority.

Next, we turned to the LSTM model for our forecasting efforts. We used the LSTM function within the Keras library of TensorFlow to build and assess our LSTM model. To determine the model’s performance, we evaluated both the training Mean Squared Error (MSE) and test MSE. Both the training and test MSE values were notably low, indicating that our model adeptly fits the training data, effectively capturing its patterns and variability. Furthermore, the model generalizes well to unseen data, thereby enhancing the reliability of our predictions.

**Figure 1**

*Summary statistics table to show how the features are distributed*

*A table of numbers with a white background

Description automatically generated with medium confidence*

**Figure 2**

*RIG Train and Test Data Distribution*

*A graph showing a number of data

Description automatically generated with medium confidence*

**Model Optimization (Hyperparameter Tuning) And Model Selection**

Next, we delved into the process of optimizing our model to enhance its performance. This involved conducting hyperparameter tuning for our LSTM model, as illustrated in Figure 3. The outcomes were notable- hyperparameter tuning led to a substantial improvement in the training MSE. However, it’s worth noting that the test MSE for the ‘NLY’ symbol did not improve post-tuning; instead, it increased. In contrast, for all other symbols, the test MSE displayed significant improvement following the tuning process. We predicted the future prices of the stock using the best model achieved through hyperparameter tuning, as shown in Figure 4. The LSTM model picks the fluctuations in price much better than the ARIMA model.

**Figure 3 Figure 4**

*Hyper Parameter Tuning for LSTM Model Predicted and Original Price Data for RIG*

*A screenshot of a computer code

Description automatically generated*A graph of a price

Description automatically generated with medium confidence

# **Conclusion**

**Conclusion, Lessons Learned and Recommendations**

In our analysis, the LSTM model proved to be the most effective in predicting stock prices with lowest MSE scores while exhibiting minimal overfitting.

As we look ahead to future analyses, we have several key considerations:

1. We plan to experiment with various thresholds for hierarchical clustering to see how they affect the resulting clusters and whether they provide any meaningful insights.
2. We intend to define specific trading signals, such as buying a stock when the 16-week moving average is greater than the 50-week moving average and selling it when the opposite occurs.
3. We aim to identify stable stocks and the high performing ones and combine them to create a diversified portfolio.
4. We aim to develop and test trading strategies based on technical indicators such as RSI, MovingAverage, MACD, and other relevant metrics.

In summary, our analysis has provided valuable insights into stock price prediction. The lessons learned will guide our future analyses for more informed and effective portfolio management and trading decisions.