# Time Series Forecasting of Stock Prices Using ARIMA and Deep Learning Models

Mahesh Komma - 22098605

11 April 2025

## Introduction

Time series forecasting helps us predict future values based on past data. This is especially useful in areas like business, finance, and economics. In this report, we look at predicting the stock prices of Amazon and Johnson & Johnson using their past stock price data. Being able to predict stock prices accurately is important for making good investment choices.
We will compare three forecasting models: ARIMA (a traditional statistical method), LSTM (Long Short-Term Memory), and GRU (Gated Recurrent Unit). While ARIMA has been used for many years, LSTM and GRU are newer models that use deep learning to capture more complex patterns in data.
The purpose of this report is to find the best model for predicting stock prices by testing these models on Amazon and Johnson & Johnson's stock data. We will compare their performance using measures like Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared ($R^2$). This will help us understand which model gives the best predictions and show the pros and cons of each one.

## 1 Methodology

### Intial Data Analysis

I present the initial plots that help visualize the stock price trends of Amazon and Johnson Johnson. I created time series plots for both datasets to get an overview of the price movements over time. These plots reveal the general trends, seasonal fluctuations, and potential outliers in the data. By examining these plots, I was able to understand the behavior of the stock prices, which informed the subsequent steps in preparing the data for modeling.
**Amazon Time series plot**


Figure 1: Amazon Time series plot

**JJ Time series plot**

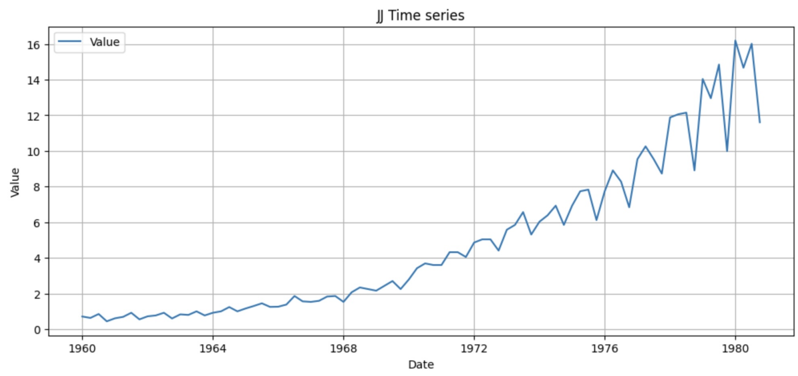



Figure 2: JJ Time series plot

### 1.1 Data Preparation

The data preprocessing involved the following steps:

1. **Handling Missing Values**: Missing data points were filled using nearby values to ensure there were no gaps in the dataset.

2. **Scaling the Data**: The data was scaled to a range of 0 to 1. This normalization step ensures that all features contribute equally to the model, preventing features with larger values from dominating the learning process.

3. **Splitting the Data**: The data was divided into two sets: 80% for training the models and 20% for testing. This allows the models to be trained on one set of data and tested on another to evaluate their performance accurately.

## 2 ARIMA Model Implementation for Amazon and Johnson & Johnson Datasets

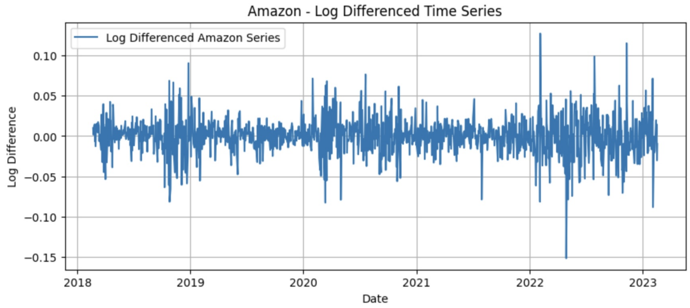
The ARIMA (AutoRegressive Integrated Moving Average) model is a widely-used technique for time series forecasting. It captures the dependencies in historical data and uses those patterns to make predictions about future values. In this analysis, ARIMA was applied to both the Amazon and Johnson & Johnson (JJ) datasets to predict future values, following a series of systematic steps.

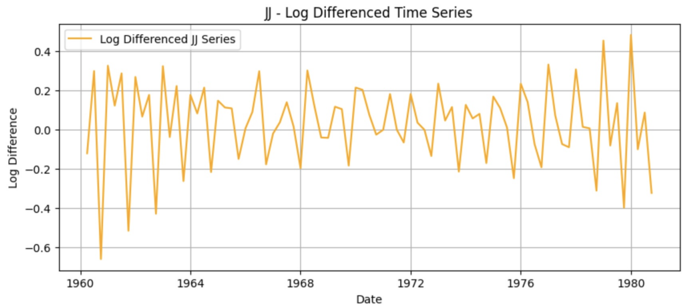
### 2.1 Stationarity Tests

Before applying the ARIMA model, it is crucial to check the stationarity of the datasets. A stationary series has constant mean and variance over time, which is a requirement for ARIMA. The Augmented Dickey-Fuller (ADF) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests were used to assess stationarity for both datasets.
**Amazon Dataset:** ADF Statistic: -1.6578, p-value: 0.4530 (indicating the series is non-stationary). KPSS Statistic: 2.9688, p-value: 0.0100 (suggesting the presence of a trend).
**Johnson & Johnson Dataset:** ADF Statistic: 2.7420, p-value: 1.0000 (indicating the series is non-stationary). KPSS Statistic: 1.3635, p-value: 0.0100 (suggesting the presence of a trend). Given the non-stationarity of both datasets, the series needed to be transformed before modeling.

### 2.2 Logarithmic Transformation and Differencing

To make the series stationary, a logarithmic transformation was applied to both datasets, followed by first-order differencing. Differencing helps remove trends and stabilize the mean of the series.
**Amazon Differenced plot**
**JJ Differenced plot**




For both datasets, the log transformation was applied to reduce variance, and then the series was differenced to make it stationary. The transformed series were plotted to visually assess their behavior after differencing.

## 2.3 ACF and PACF Plots

Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots were used to determine the appropriate values for the AR (AutoRegressive) and MA (Moving Average) terms. These plots showed the lags at which the correlations are significant, helping in the identification of the p and q parameters of the ARIMA model.

## 2.4 ARIMA Model Optimization

The ARIMA model requires the selection of three key parameters:

- p (AR term)
- d (differencing)
- q (MA term)

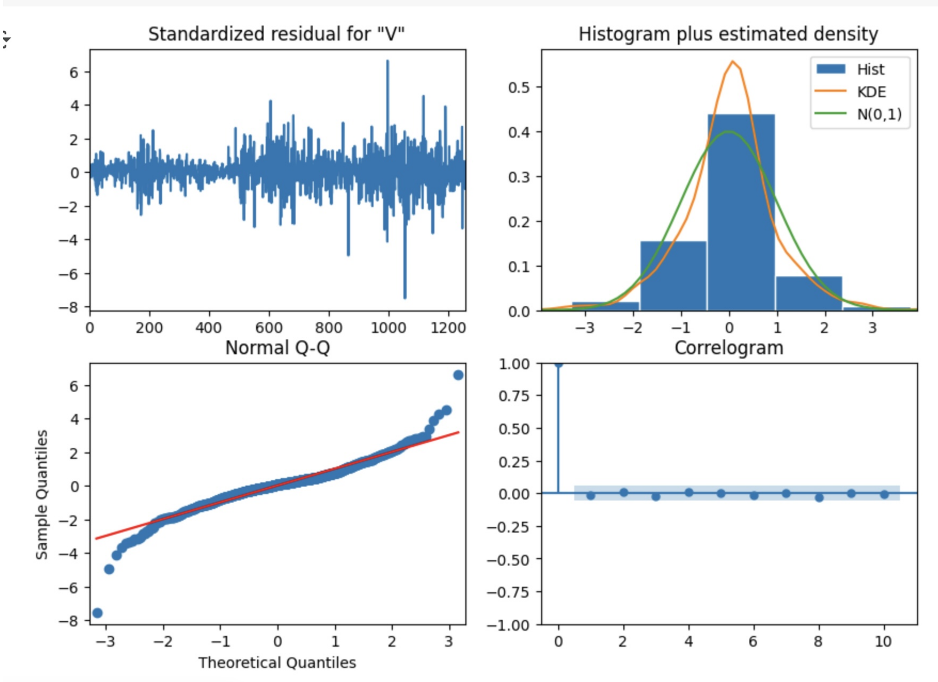
To find the best combination of these parameters, an optimization approach was used. The function optimize_ARIMA was designed to test different combinations of p and q while keeping d = 1 (since differencing was applied), and then evaluating the model using the Akaike Information Criterion (AIC). The optimal parameters were selected based on the lowest AIC value.

After optimizing the parameters, the best model for both the Amazon and JJ datasets was fitted, and the model summaries were generated.

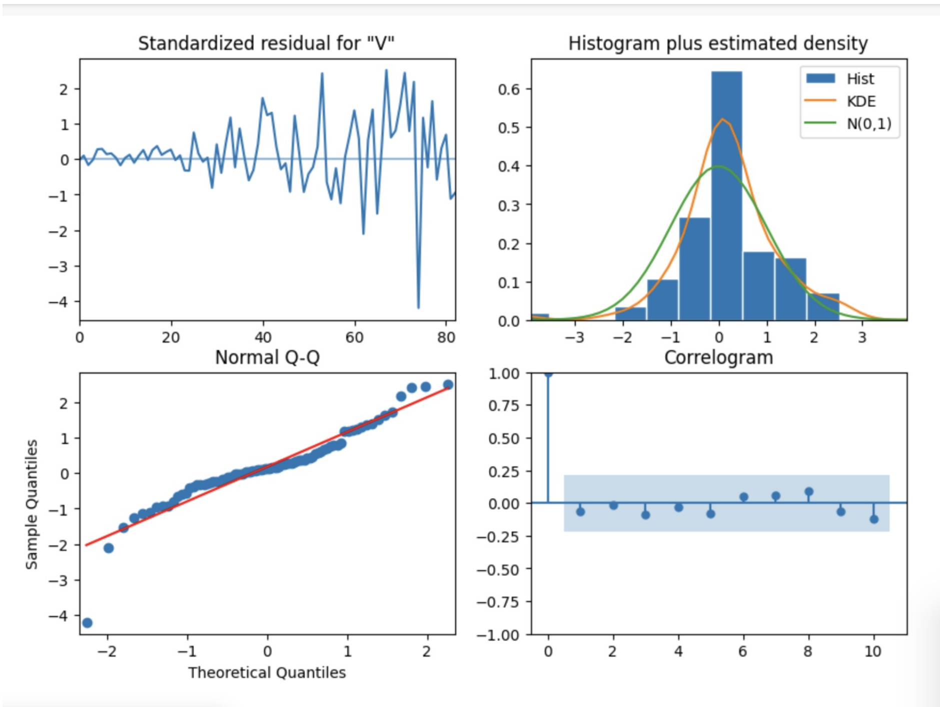
## 2.5 Model Diagnostics

To assess the fit of the ARIMA models, diagnostic plots were generated. These plots, which include residual plots and tests for autocorrelation, help to check if the model has captured the underlying patterns in the data effectively.

**Amazon Diagnostics plot**




**JJ Diagnostic plot**



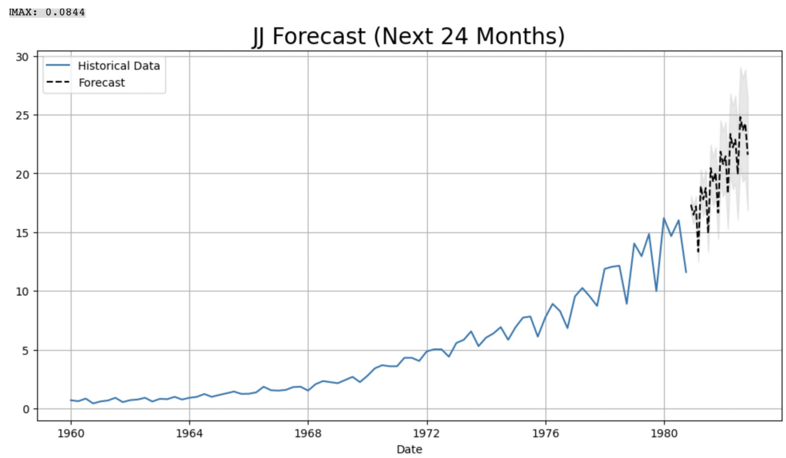

## 2.6 evaluation metrics

To assess the performance of the ARIMA models, several evaluation metrics were used, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and correlation coefficient (CORR). For the Amazon dataset, the ARIMA model achieved a low RMSE of 3.4377, MAE of 1.9761, and an impressive CORR of 0.9951, indicating strong predictive accuracy. Similarly, the Johnson & Johnson dataset showed an RMSE of 0.3999, MAE of 0.279, and a CORR of 0.9958, further confirming the model's reliability. After evaluation, the models were used to forecast the next 24 months of data. Forecast plots were generated with 95% confidence intervals, offering a clear visual of the expected trends and variability in future values based on historical patterns.

## 2.7 Forecast using Arima

For forecasting future values, ARIMA models were fitted to the Amazon and Johnson & Johnson time series data. Based on AIC optimization, the best-fit model for the Amazon dataset was ARIMA(2,1,2), while the Johnson & Johnson dataset was best modeled using ARIMA(6,1,3). These models effectively captured the underlying patterns in each dataset after differencing to achieve stationarity. Using the fitted models, forecasts for the next 24 months were generated along with 95% confidence intervals. The Amazon forecast revealed a consistent upward trend, while the Johnson  Johnson predictions indicated moderate growth with less volatility. These forecast plots provided valuable insights into the future behavior of each time series based on historical data trends.
**Amazon Forecast plot**


**JJ Forecast plot**




# 3 LSTM & GRU Model

## 3.1 Data Splitting

Before training the LSTM and GRU models, the data was first scaled using MinMaxScaler to bring all values between 0 and 1. A window size of 12 was used, which means each input to the model had 12 previous time steps, and the target was the next value. The data was then split into training and testing sets with an 80% and 20% ratio, so the models could learn from most of the data and be tested on unseen data.

## 3.2 Model Architecture

Two deep learning models were used: LSTM and GRU. Both models had:

- One LSTM or GRU layer with 64 units and ReLU activation.
- One Dense layer with 1 unit to make the prediction.
- The model used Adam optimizer and Mean Squared Error (MSE) loss.
- Training was done for 50 epochs with a batch size of 16.

This structure helped the models learn from past values and make future predictions.

### 3.3 Results

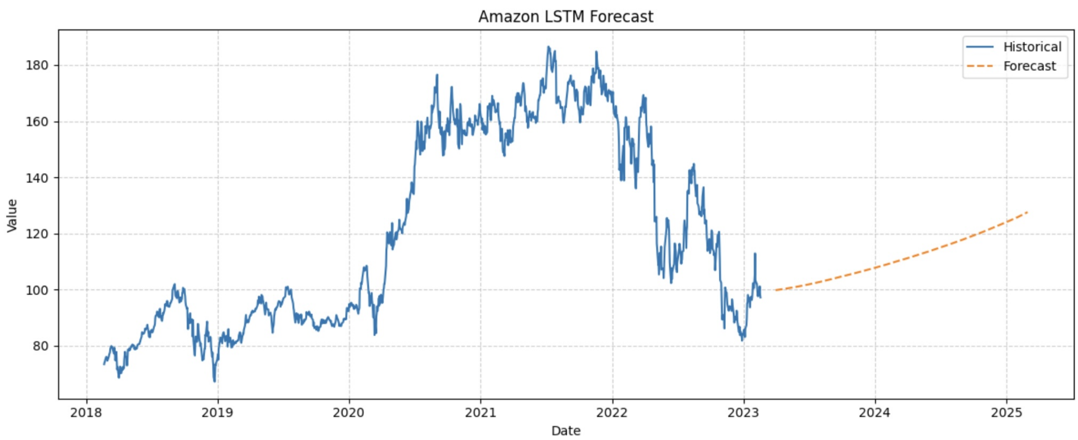
The models were evaluated using RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error). These metrics show how close the model's predictions were to the real values.

| Model | RMSE | MAE |
|---|---|---|
| LSTM | 5.00 | 3.95 |
| GRU | 3.95 | 3.05 |

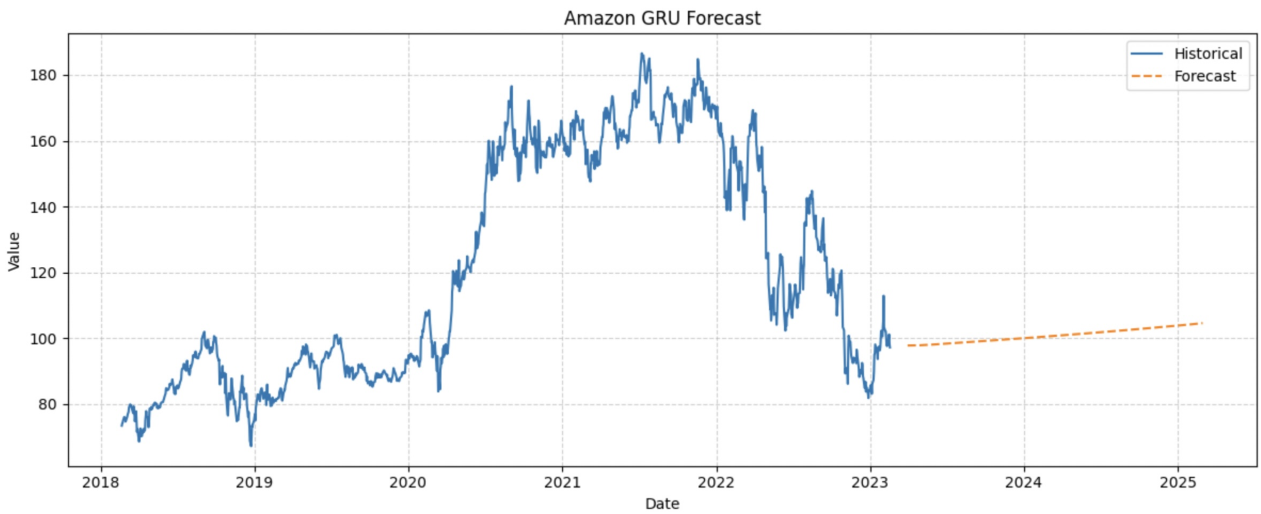
Table 1: Performance of LSTM and GRU models on Amazon dataset

From the results, the GRU model performed better than LSTM with lower error values, which means it predicted more accurately.

**LSTM & GRU Forecast plots of Amazon**

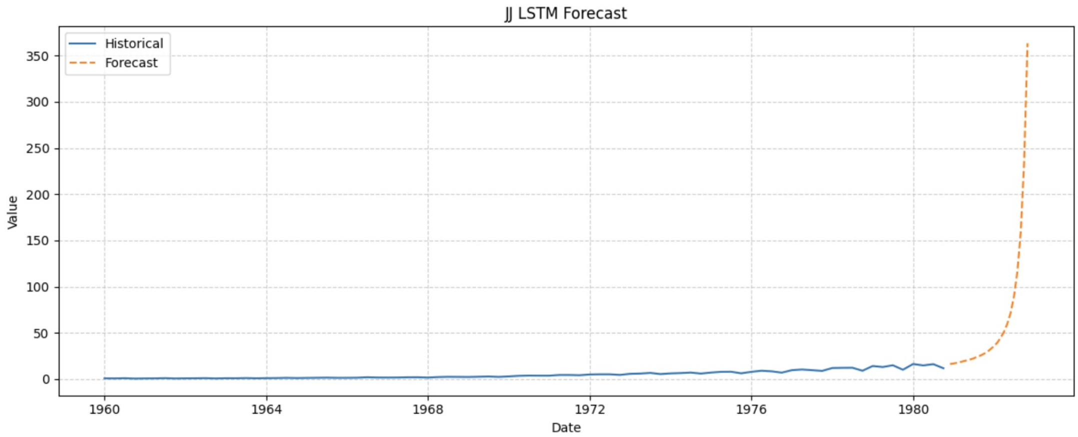



**Amazon GRU Forecast Plot**

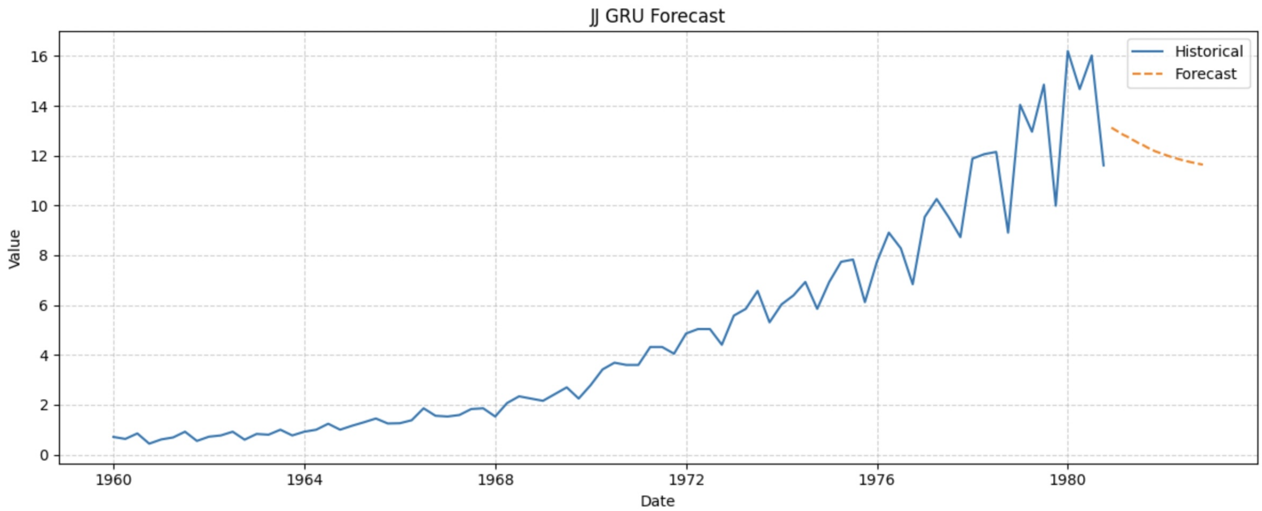


### LSTM & GRU Forecast plots for JJ

### JJ LSTM Forecast Plot




### JJ GRU Forecast Plot




## 4 Conclusion

Based on the comparison of results, the following conclusions can be made:

- For the Amazon dataset, ARIMA (2,1,2) is the most effective model, outperforming both LSTM and GRU in terms of RMSE and MAE. - For the Johnson & Johnson dataset, the GRU model provided the best results, closely followed by ARIMA and LSTM models, with GRU having a slight edge in accuracy.

In general, ARIMA appears to be a strong candidate for forecasting time-series data with simpler patterns, like the Amazon dataset, while GRU outperforms the other models for the Johnson & Johnson dataset, highlighting the potential advantage of using deep learning models for more complex time-series patterns.