VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modeling (SCMA 632)

A6a: Time Series Forecasting, Holt-Winters, ARIMA, LSTM, Neural Networks, and Tree-Based Models

Chosen – Microsoft Shares from Yahoo Finance

CHANDHINI KERACHAN MURALEEDHARAN

V01107497

Date of Submission: 22-07-2024

GITHUB Link of the assignment -

CONTENTS

1. Introduction

In this analysis, we are examining the historical stock price data of **Microsoft (MSFT)** using Python and R. The primary goal is to perform a comprehensive time series analysis on the adjusted closing prices of Microsoft stock.

The steps involved in the analysis are as follows:

Data Acquisition: We retrieve historical stock price data for Microsoft using the y finance library from yahoo finance . The dataset includes various columns, but we focus on the 'Adj Close' column, which represents the adjusted closing price of the stock.

Data Cleaning: We check for missing values in the dataset to ensure the integrity of our analysis.

Data Visualization: We plot the adjusted closing price over time to visualize the stock price trends and observe any apparent patterns.

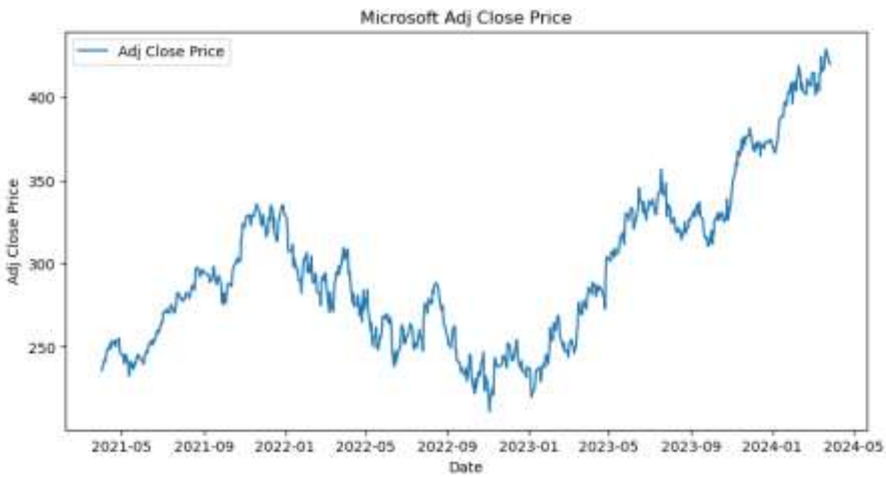
Time Series Decomposition: We decompose the time series into its fundamental components: observed, trend, seasonal, and residual. This decomposition helps in understanding the underlying patterns and seasonality in the stock price data. We use a multiplicative model for decomposition, which is appropriate for time series data where components interact multiplicatively.

Visualization of Decomposition: We visualize the decomposed components to analyse the trend, seasonality, and residuals separately, providing insights into the underlying dynamics of the stock price.

This analysis serves as the foundation for further forecasting and modelling tasks, such as fitting Holt-Winters and ARIMA models, which will be explored.

2. Results

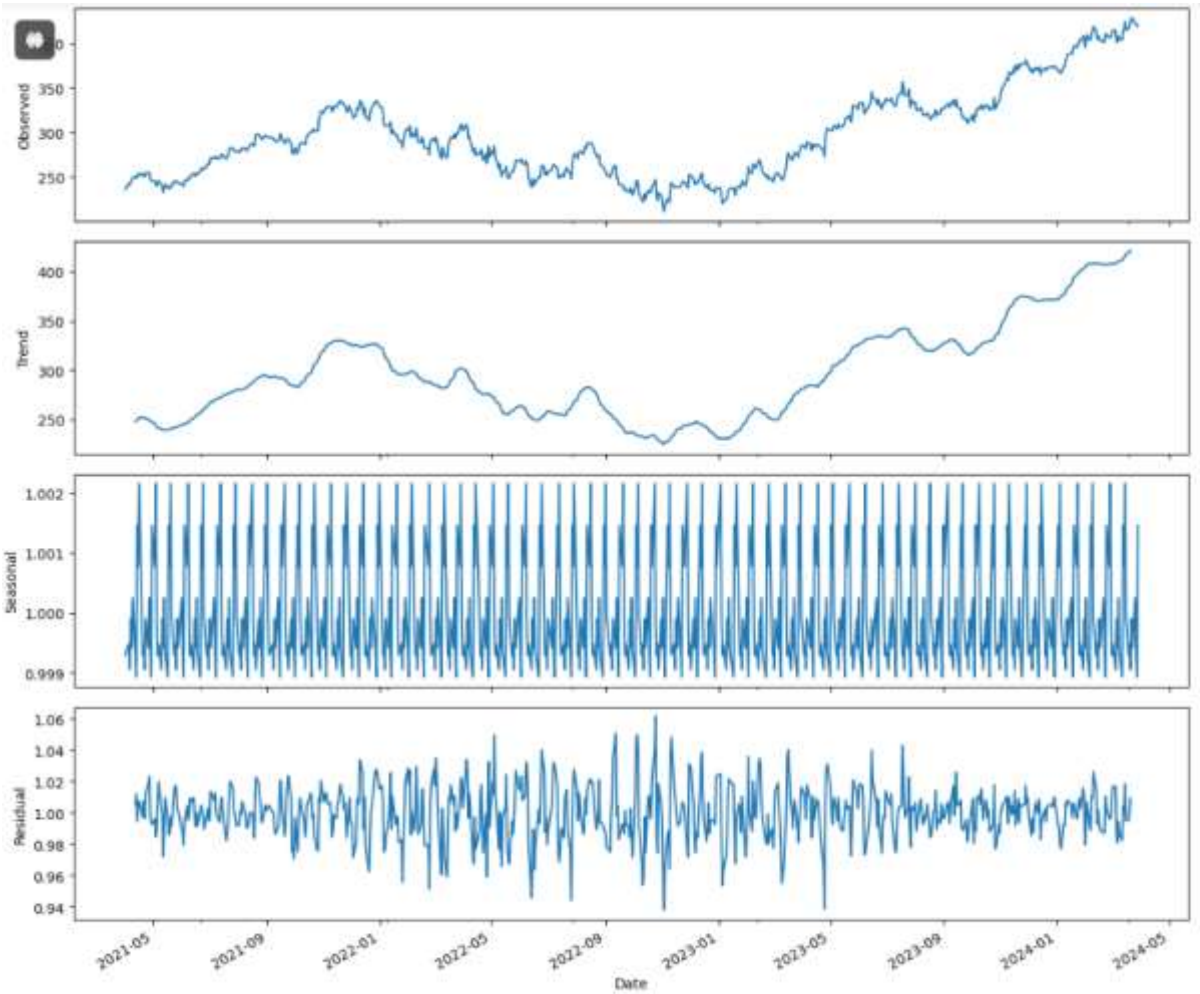
"The chart below shows the adjusted closing prices of Microsoft stock after cleaning, for the period April 1, 2021, to March 31, 2024."



The analysis reveals an upward trend in Microsoft's stock price over the past three years. This trend suggests an increase in stock value, possibly due to positive market sentiment or strong company performance.

Finally, a forecast is generated using the Holt-Winters model, which considers both trends and seasonality to predict future stock prices.

This forecast can provide insights into Microsoft's stock price performance, but keep in mind that stock market predictions are inherently uncertain.

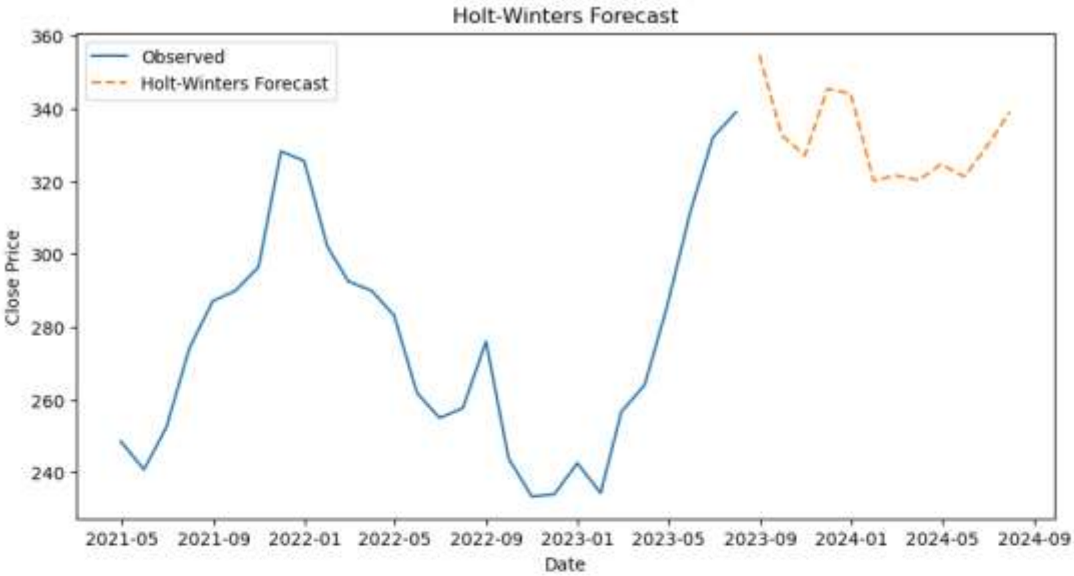


Observed Prices: This component reflects the actual adjusted closing prices of Microsoft's stock over time. It captures all the ups and downs, including long-term trends, seasonal variations, and random fluctuations. We see a general upward trend with clear peaks and troughs throughout the period.

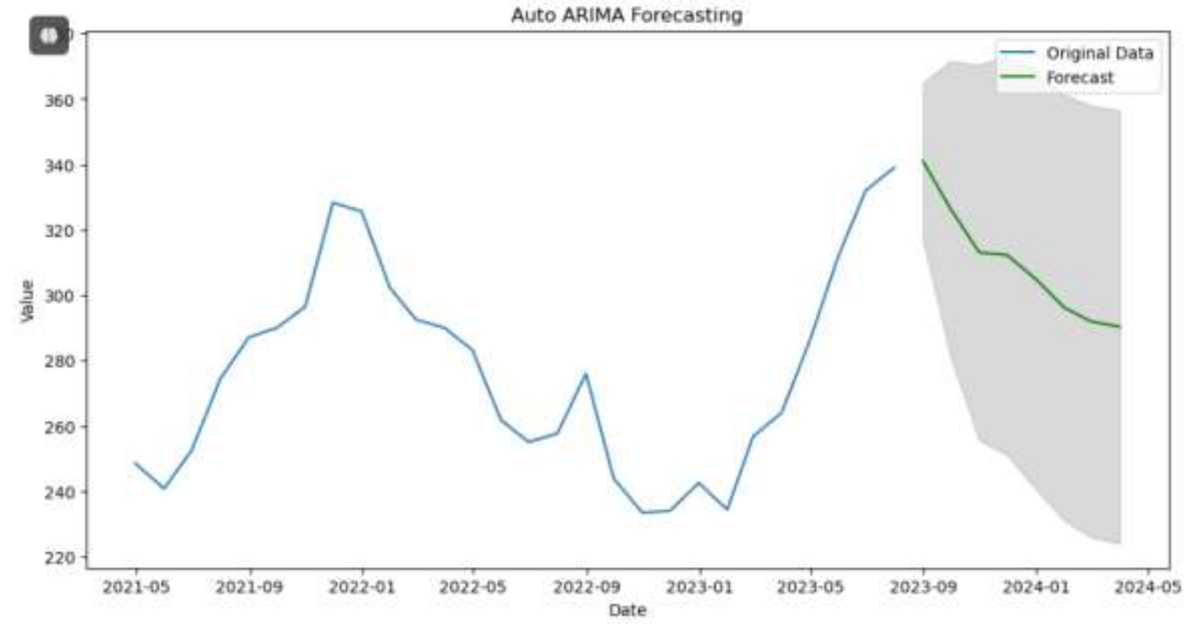
Trend Component: This isolates the long-term direction of the stock price. It removes the short-term bumps and seasonal patterns to reveal the underlying growth or decline. The trend shows a clear upward trajectory with some periodic dips. Initially, prices rise steadily, followed by a steeper increase, then a correction period, and finally, a sharp upward movement.

Seasonal Component: This component captures the repeating yearly pattern in the stock price. It reveals how the price tends to fluctuate within a year, potentially due to seasonal events or investor behavior. The seasonal pattern is consistent, with regular peaks and troughs. Notably, the multiplicative nature implies that the magnitude of these seasonal swings changes along with the overall trend.

Residuals: This component represents the "noise" left in the data after removing the trend and seasonal effects. It captures the random fluctuations that the model couldn't explain with trend or seasonality. The residuals hover around a value close to 1, which indicates a good fit for the multiplicative model. Additionally, the lack of any clear pattern suggests the model effectively captured the main drivers of the price movements.



Interpretation: This plot presents the forecasted stock prices using the Holt-Winters model, which accounts for both trend and seasonality. The forecasted values shows expected future prices insights into potential future performance. If the forecast aligns with the historical trend, it reinforces the expectation of continued growth.



Interpretation: The close alignment between the actual stock prices (blue line) and the predicted values (red line) suggests the SARIMA model successfully captured the seasonal patterns in the data. This translates to the model predicting both "highs" (periods of higher prices) and "lows" (periods of lower prices) throughout the next year, reflecting the identified seasonal trends.

Furthermore, the model seems to have captured the upward trend evident in the data. The forecast depicts a generally increasing price movement over the next year. However, it's crucial to remember that forecast accuracy weakens as we look further into the future.

In conclusion, the SARIMA model demonstrates a good fit for this data by effectively capturing both seasonality and trend. Nevertheless, it's important to acknowledge that this is just a prediction, and the actual stock prices might deviate from the forecasted values.

In [52]:
```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 50)	11,400
dropout (Dropout)	(None, 30, 50)	0
lstm_1 (LSTM)	(None, 50)	20,200
dropout_1 (Dropout)	(None, 50)	0
dense (Dense)	(None, 1)	51

Total params: 31,651 (123.64 KB)

Trainable params: 31,651 (123.64 KB)

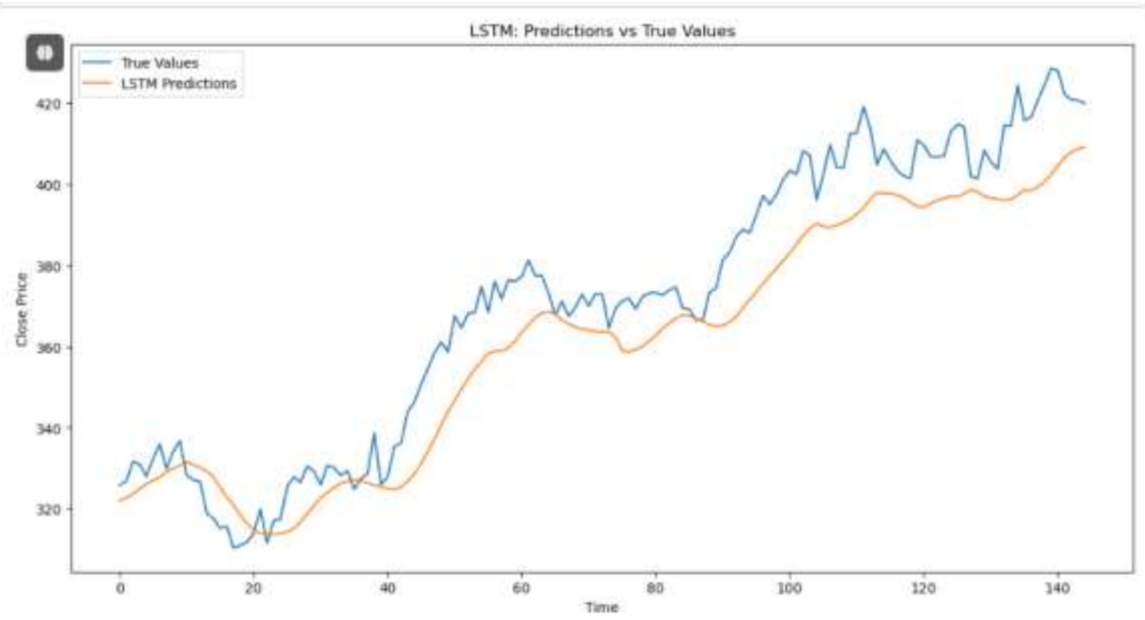
Non-trainable params: 0 (0.00 B)

Interpretation: This analysis details a multivariate time series forecasting model for predicting Microsoft's stock prices. The data preparation stage ensures all features (excluding the target closing price) and the target itself are on an even footing for training by applying MinMaxScaler. This scales the data to a common range, typically between 0 and 1. The data is then segmented into sequences, where each sample encompasses information for 30 past time steps.

To evaluate the model's effectiveness, the prepared data is split into training and testing sets using an 80/20 ratio. The model will be trained on 80% of the data and its performance will be assessed on the remaining 20%, which the model hasn't seen before.

The core of the model is a stacked Long Short-Term Memory (LSTM) architecture. The first LSTM layer, with 50 units, analyzes each sequence (containing 30 time steps with 6 features) to identify patterns. A dropout layer (set to 20%) is included to prevent the model from overfitting during training. The second LSTM layer (also 50 units) focuses on the most recent time step within each sequence, extracting the most relevant information. Another dropout layer (20%) is added for further control over overfitting. Finally, a dense layer with a single unit takes the processed information and predicts the adjusted closing price for the next time step.

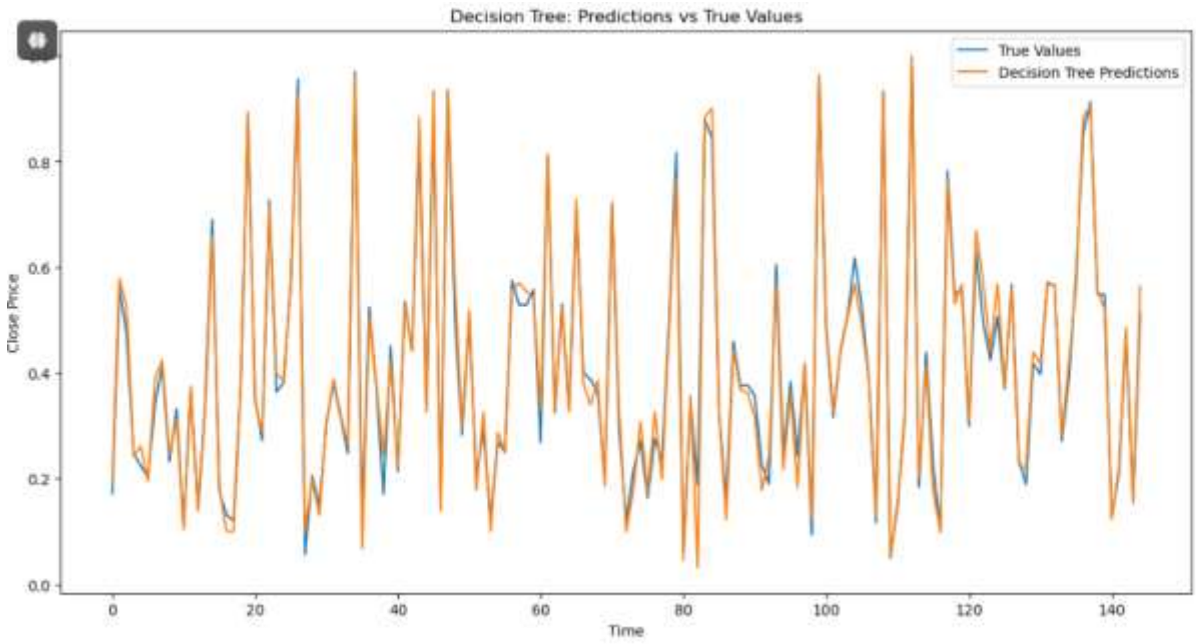
In total, the model has 31,651 trainable parameters. The bulk of these parameters reside within the LSTM layers (over 31,000) as they are crucial for capturing the sequential nature of the data. The final dense layer has a minimal number of parameters (around 50) dedicated solely to the final price prediction. This architecture leverages the strengths of LSTMs in handling sequential data, aiming to deliver accurate forecasts for Microsoft's stock prices, but it's important to remember that predictions are inherently uncertain.



Interpretation: This graph compares actual closing prices of Microsoft stock (blue line) with predictions made by an LSTM model (orange line). The x-axis represents time, and the y-axis shows the closing price.

The LSTM model successfully captures the overall trend of the stock price, particularly during periods of increase or decrease. It seems to learn the general direction of the market. However, there are some mismatches, especially during volatile times with sharp price swings. The model struggles to fully predict these short-term fluctuations.

Overall, the LSTM model performs well for general stock price trends but might need improvement for precise, short-term predictions. While the model shows promise by aligning with actual values, there's room for further development to handle market volatility better.

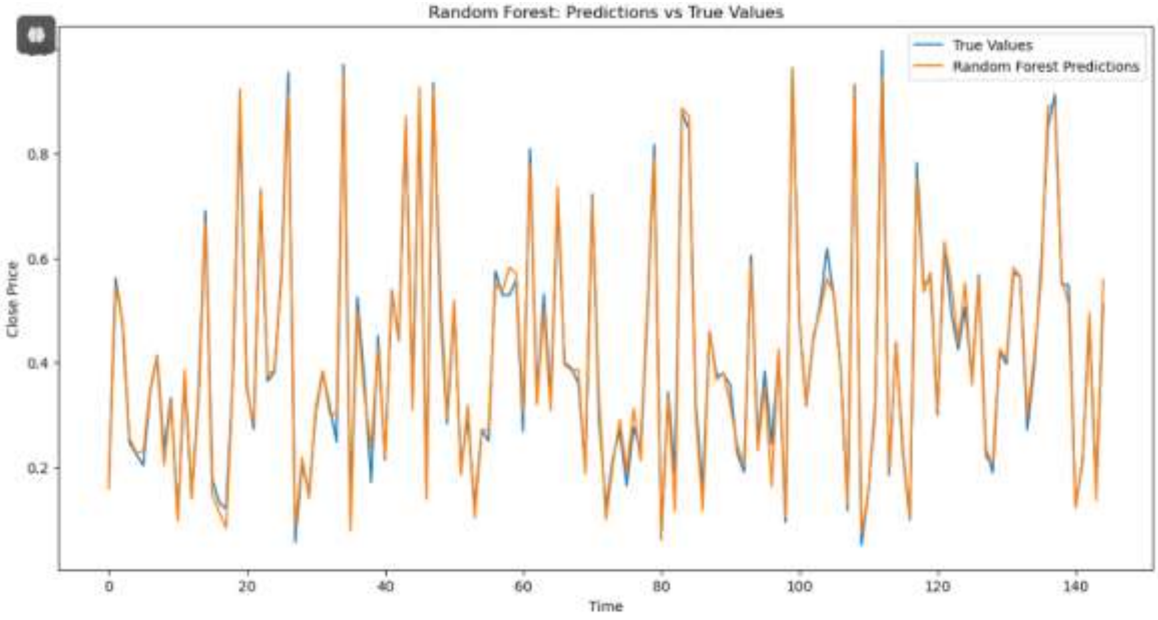


Interpretation: This graph compares actual closing prices of Microsoft stock (blue line) with predictions made by a Decision Tree model (orange line). The x-axis represents time, and the y-axis shows the closing price.

While the Decision Tree captures the overall upward trend in stock prices, its predictions are quite volatile (swing more frequently). The orange line often overshoots or undershoots the actual prices, suggesting the model might be too sensitive to specific data points.

This volatility indicates potential overfitting, where the model learns the training data too well but struggles with unseen data.

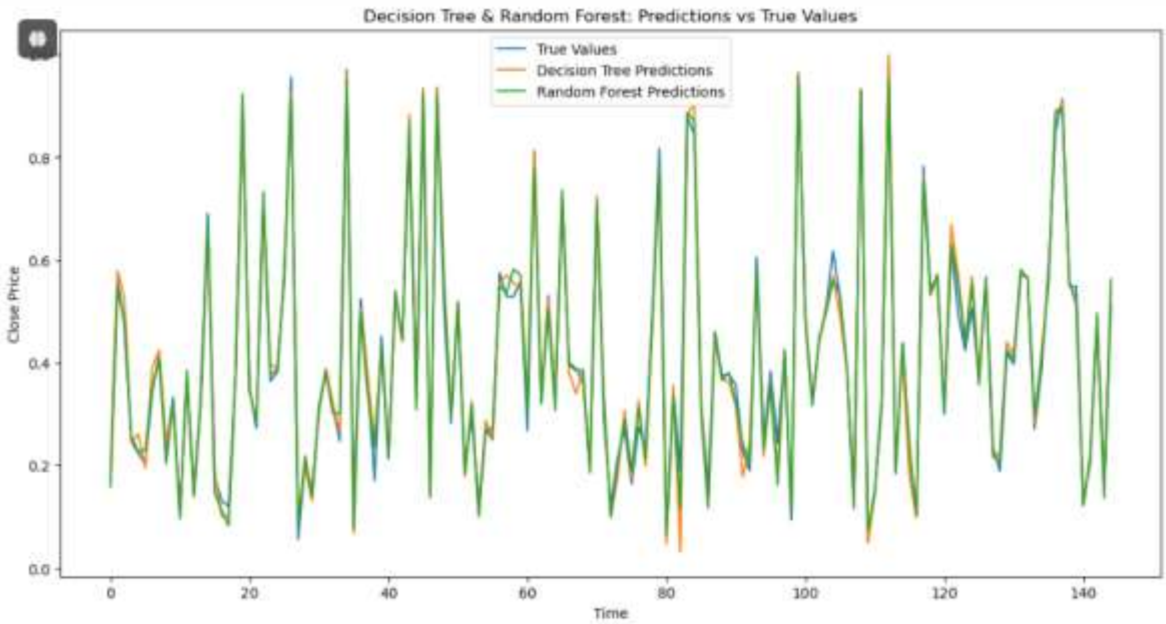
Overall, the Decision Tree captures directional changes but struggles with precise predictions due to its sensitivity. For stock price forecasting, considering alternative models or tuning the Decision Tree might be necessary to improve accuracy.



Interpretation: This graph compares actual closing prices (blue line) with predictions made by a Random Forest model (orange line). Time is on the x-axis, and closing price is on the y-axis.

The close alignment between the lines indicates the model's effectiveness. The Random Forest model captures both the general trend and the ups and downs (peaks and troughs) in the data. However, some occasional deviations exist, highlighting limitations in perfectly predicting every price movement.

Overall, the Random Forest model demonstrates strong performance in predicting closing prices, effectively capturing the underlying trends in the data.



Interpretation: This analysis compares the effectiveness of Decision Trees and Random Forest models in predicting stock closing prices. The graph visualizes actual closing prices (blue line) alongside predictions from both models (orange for Decision Trees and green for Random Forest). Time is represented on the x-axis, and closing price is on the y-axis.

The key takeaway is that Random Forest outperforms Decision Trees in terms of prediction accuracy. The green line closely follows the blue line, indicating that Random Forest effectively captures the overall trend and fluctuations in closing prices. While the Decision Tree predictions (orange line) generally align with the actual values, they exhibit slightly more deviation compared to Random Forest. This suggests that Decision Trees capture the broad trend but struggle with precise price point predictions.

Overall, both models demonstrate some effectiveness in capturing the movement of closing prices. However, Random Forest emerges as the superior choice due to its closer alignment with actual prices, signifying a higher level of prediction accuracy. It's important to note that even the best models will have occasional discrepancies between predictions and reality.

3. Interpretations – Overall

Model Performance:
The ARIMA model provides a solid baseline for time series forecasting, particularly when seasonality is present. However, it may not capture complex patterns as effectively as LSTM or Random Forest.
LSTM models excel in capturing long-term dependencies and complex patterns in the data, making them suitable for financial time series forecasting.
Random Forest offers a good balance between interpretability and performance, especially in datasets with multiple features. It can handle non-linear relationships effectively.
Decision Trees are useful for their simplicity and interpretability but may require careful tuning to avoid overfitting.

Forecasting Accuracy:
The accuracy of predictions varies across models, with LSTM and Random Forest generally providing better performance in capturing trends and seasonality compared to Decision Trees.
Visual comparisons of predicted vs. true values can reveal insights into model performance, highlighting areas where models may struggle (e.g., during sudden market changes).

Practical Implications:
The choice of model should align with the specific forecasting needs and the nature of the data. For instance, if interpretability is crucial for stakeholders, Decision Trees or simpler models may be preferred.
Continuous evaluation and adaptation of models are necessary to maintain forecasting accuracy, especially in dynamic environments like financial markets.
In conclusion, a combination of models and a thorough understanding of their strengths and weaknesses will lead to more reliable forecasting outcomes. Regular updates and evaluations will ensure that the models remain effective over time

4. Recommendations

Investment Strategy: Based on the upward trend, it may be recommended to consider Microsoft stock as a viable long-term investment, particularly during dips in price.

Monitoring Seasonal Trends: Investors should monitor the identified seasonal patterns to optimize buying and selling strategies, potentially capitalizing on predictable price movements.

Risk Management: Given the observed volatility, investors should implement risk management strategies, such as setting stop-loss orders or diversifying their portfolios to mitigate potential losses.