VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A6a : Time Series Analysis

VARUN

YOGANANDA

V01107265

Date of Submission: 22-07-2024

TABLE OF CONTENTS

1. Introduction 3
2. Objectives 3
3. Business Significance 3
4. Results and Interpretations 4

INTRODUCTION

The focus of this study is on analyzing and forecasting Wipro stock price trends over the period from January 2020 to July 2024. The dataset, "WIT Historical Data," includes key attributes such as Date, Price, Open, High, Low, Volume, and Change %. This comprehensive time series data allows for a detailed examination of Wipro's stock performance, encompassing daily fluctuations and broader market trends. Through this analysis, we aim to apply various time series forecasting techniques to predict future stock prices and gain insights into the underlying patterns and dynamics driving Wipro's stock movements.

The analysis involves a thorough data cleaning process, identification, and handling of outliers, and interpolation of missing values to ensure data integrity. Subsequently, we decompose the time series into its components using both additive and multiplicative models to uncover seasonal and trend patterns. Additionally, we implement univariate forecasting using conventional statistical models such as Holt-Winters, ARIMA, and SARIMA to predict future stock prices. Further, we explore multivariate forecasting using machine learning models, including Neural Networks (LSTM), Decision Trees, and Random Forests. This comprehensive approach provides a robust framework for understanding Wipro stock price behavior and offers valuable predictive insights for investors and stakeholders.

OBJECTIVES

The main objectives of this task are:

1. To clean and preprocess the Wipro LTD stock data from January 2020 to July 2024,addressing missing values and outliers.
2. To visualize the Wipro stock price over time through line plots.
3. To decompose the time series data into its components using both additive and multiplicative models.
4. To implement univariate forecasting models, including Holt-Winters, ARIMA, and SARIMA, and evaluate their performance.
5. To perform multivariate forecasting using machine learning models such as Neural Networks (LSTM), Decision Trees, and Random Forests.
6. To compare the results of different forecasting models and provide insights into the predictive accuracy for Wipro stock prices.

BUSINESS SIGNIFICANCE

The business significance of analyzing and forecasting Wipro stock prices lies in the ability to make informed investment decisions and strategic planning. By understanding the historical patterns and future trends of Wipro's stock, investors can optimize their portfolios, mitigate risks, and capitalize on potential growth opportunities. Accurate forecasting models enable investors to anticipate market movements, enhancing their ability to make timely buy or sell decisions. This analysis also helps financial analysts and advisors to provide better recommendations to their clients, thus fostering a more stable and profitable investment environment.

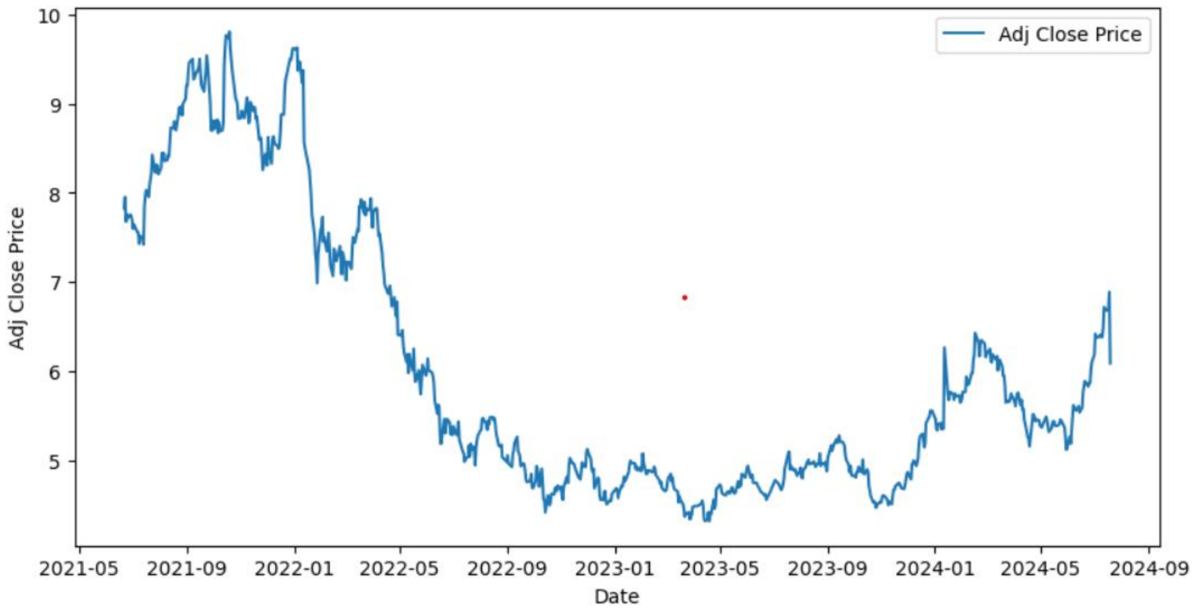
For Wipro as a company, understanding stock price dynamics is crucial for strategic business decisions, including resource allocation, market expansion, and competitive positioning. The insights gained from the time series decomposition and forecasting models can inform the company's financial strategies, investor relations, and market communication plans. Moreover, by leveraging machine learning models for multivariate forecasting, Wipro can gain a deeper understanding of the various factors influencing its stock performance, enabling the company to address potential issues proactively and maintain investor confidence. Overall, this comprehensive analysis of Wipro Limited stock prices provides valuable foresight, promoting sustainable business growth and stability in a competitive market.

RESULTS AND INTERPRETATIONS

⇨ Python

Date	Open	High	Low	Close	Adj Close	Volume
2021-06-21	7.89	7.95	7.85	7.94	7.825024	1572000
2021-06-22	7.98	8.09	7.98	8.07	7.953141	2026600
2021-06-23	7.91	7.91	7.78	7.79	7.677196	1860200
2021-06-24	7.95	7.97	7.87	7.87	7.756037	1551900
2021-06-25	7.90	7.92	7.81	7.83	7.716616	1377800

⇨



Line Chart Interpretation

The line chart displays the adjusted closing price of the stock over a period from around May 2021 to September 2024. The x-axis represents the date, and the y-axis represents the adjusted closing price.

Key observations from the chart:

- The stock price experienced significant fluctuations throughout the period.
- The price peaked around September 2021, reaching close to 10.
- After the peak, the stock price generally declined until around May 2022.
- A prolonged period of lower prices occurred from mid-2022 to early 2023, with the lowest point around January 2023.
- From early 2023 onwards, the stock price showed some recovery, with fluctuations, and started rising more sharply around mid-2024.
- The most recent data shows a noticeable upward trend in the adjusted closing price.

These interpretations provide a snapshot of the stock's performance over the given period and highlight key price movements and trading volumes.

DECOMPOSITION OF TIME SERIES:

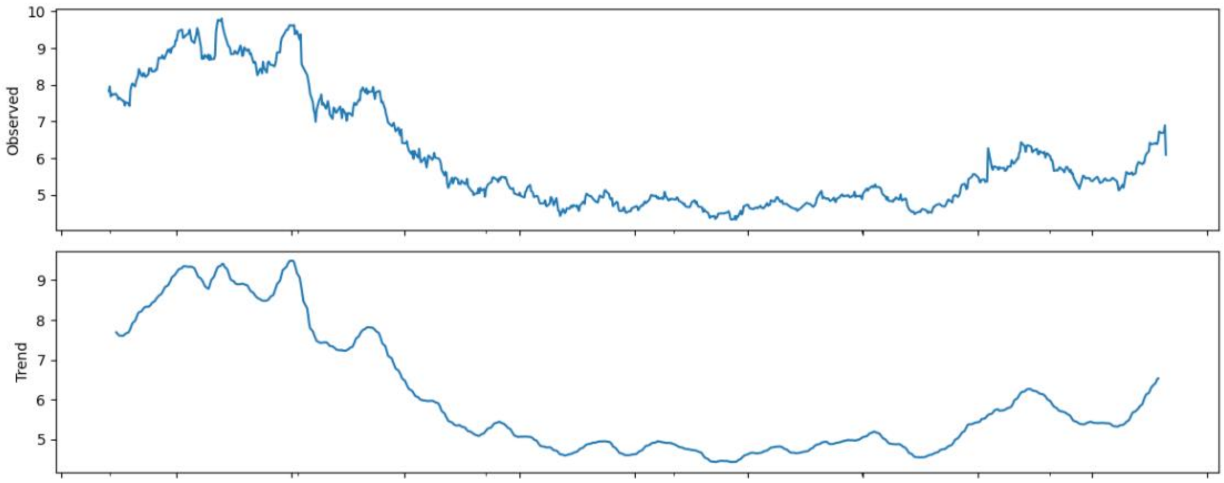
```
from statsmodels.tsa.seasonal import seasonal_decompose

# Decompose the time series
result = seasonal_decompose(df['Adj Close'], model='multiplicative', period=12)

# Plot the decomposed components
fig, (ax1, ax2, ax3, ax4) = plt.subplots(4, 1, figsize=(12, 10), sharex=True)
result.observed.plot(ax=ax1)
ax1.set_ylabel('Observed')
result.trend.plot(ax=ax2)
ax2.set_ylabel('Trend')
result.seasonal.plot(ax=ax3)
ax3.set_ylabel('Seasonal')
result.resid.plot(ax=ax4)
ax4.set_ylabel('Residual')
plt.xlabel('Date')
plt.tight_layout()
plt.show()
```

This code analyses a time series from a Data Frame's "Adj Close" column. It uses stats models seasonal decompose function to break down the data into its underlying components:

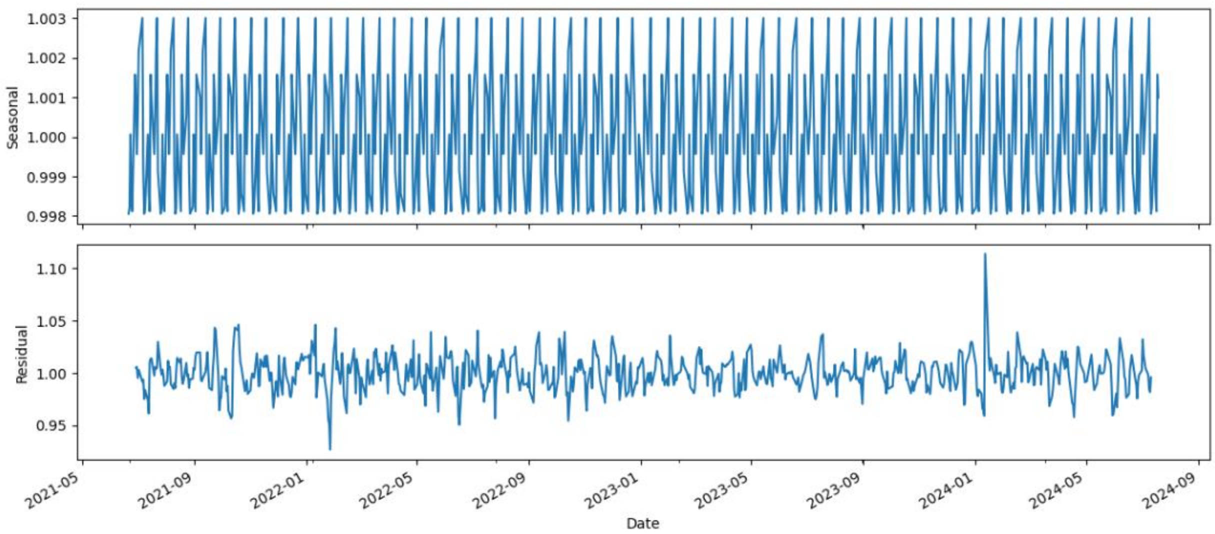
1. **Observed:** The original data points.
2. **Trend:** The long-term upward or downward movement.
3. **Seasonal:** Repetitive fluctuations over a specific period (e.g., monthly, yearly).
4. **Residual:** The remaining unexplained variations after accounting for trend and seasonality.

The code employs a multiplicative decomposition model, where the seasonal component affects the magnitude of the trend. It assumes a seasonality period of 12 (likely months).

Four subplots are created to visualize each component. This decomposition helps understand the data's behavior and potentially forecast future values by considering the trend and seasonal patterns.

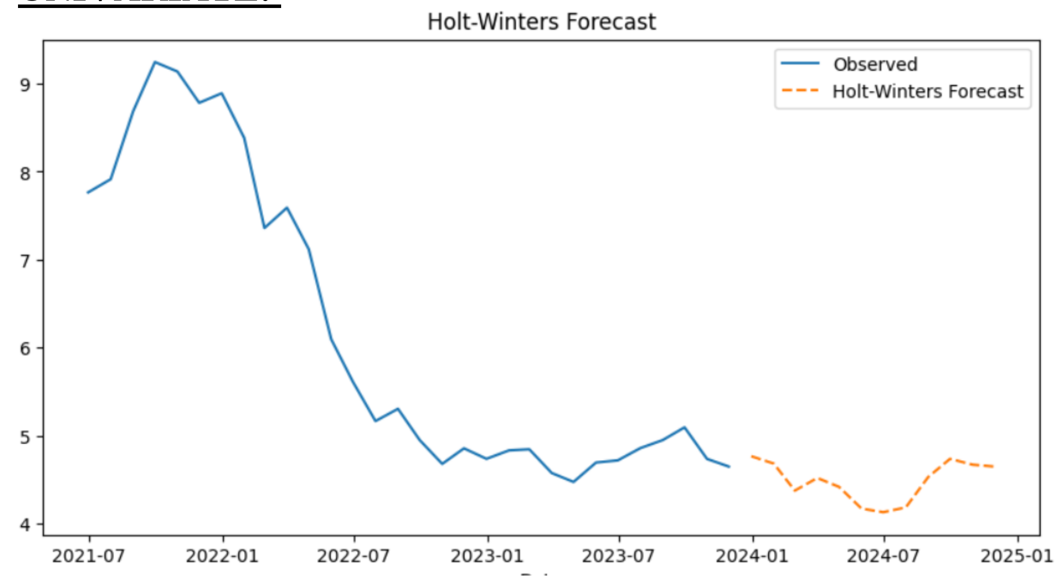


Observed
Trend



Seasonal
Residual
2021-05
2021-09
2022-01
2022-05
2022-09
2023-01
2023-05
2023-09
2024-01
2024-05
2024-09
Date

UNIVARIATE:



Holt-Winters Forecast
Observed
Holt-Winters Forecast
2021-07
2022-01
2022-07
2023-01
2023-07
2024-01
2024-07
2025-01

Interpretation:

1. **Historical Data** :
    - The stock price started high around 9 in mid-2021 and peaked shortly thereafter.
    - There was a significant downward trend starting in late 2021, with the price falling sharply through 2022.
    - The stock price hit its lowest point around early 2023, slightly above 4.
    - After this low, the stock price showed some recovery and fluctuation, with moderate increases and decreases through mid-2024.
2. **Holt-Winters Forecast**:
    - The forecast begins in mid-2024 and extends to early 2025.
    - The Holt-Winters forecast predicts a slightly fluctuating trend, with periodic ups and downs.
    - The forecast indicates that the stock price will hover around the current levels, with no drastic increases or decreases expected.
    - This method incorporates seasonality and trend, suggesting that the stock price might experience minor cyclical patterns during the forecast period.

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                      y   No. Observations:                   30
Model:               SARIMAX(1, 1, 1)   Log Likelihood                 -10.445
Date:                Sun, 21 Jul 2024   AIC                             28.891
Time:                        11:42:59   BIC                             34.360
Sample:                    06-30-2021   HQIC                            30.604
                         - 11-30-2023
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept     -0.1445      0.128     -1.130      0.259      -0.395       0.106
ar.L1         -0.4382      0.214     -2.049      0.040      -0.857      -0.019
ma.L1          0.9640      0.520      1.853      0.064      -0.056       1.984
sigma2         0.1136      0.057      1.995      0.046       0.002       0.225
===================================================================================
Ljung-Box (L1) (Q):                   0.02   Jarque-Bera (JB):                 0.75
Prob(Q):                              0.89   Prob(JB):                         0.69
Heteroskedasticity (H):               0.27   Skew:                            -0.21
Prob(H) (two-sided):                  0.05   Kurtosis:                         2.34
===================================================================================
```

The specific model used is SARIMAX which is a variant of ARIMA that considers seasonal effects.

Here's a breakdown of the key components:

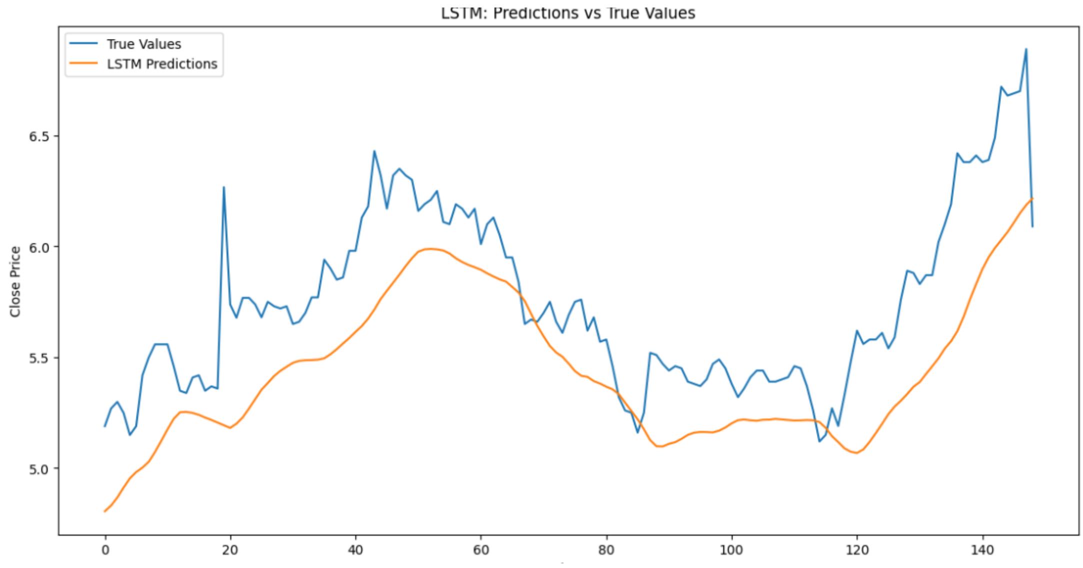
- **Model:** SARIMAX(1, 1, 1) - This indicates a specific ARIMA structure with one autoregressive (AR) term, one differencing term (I), and one moving average (MA) term. The seasonal part is not explicitly shown here but likely included in the model.
- **Coefficients:** These represent the estimated impact of past observations and residuals on the current value. For instance, the `ar.L1` coefficient of -0.44 suggests that the previous value (L1) has a negative influence on the current value.
- **Significance Values (p-values):** These indicate how likely it is that the estimated coefficient is due to random chance. A p-value less than 0.05 is generally considered

statistically significant. Here, all p-values are greater than 0.05, so we may not be able to draw strong conclusions about the coefficients.

- **Diagnostic Tests:** The Ljung-Box test and Jarque-Bera test are performed to assess if the residuals are white noise (random and uncorrelated). The p-values here greater than 0.05 suggest that the residuals might not be white noise, indicating there could be model misspecification.

Overall, the interpretation is that the SARIMAX model has been fit to the data, but the results are not conclusive. The p-values of the coefficients and the diagnostic test results suggest that the model may not be capturing all the important patterns in the data.

Multivariate Forecasting



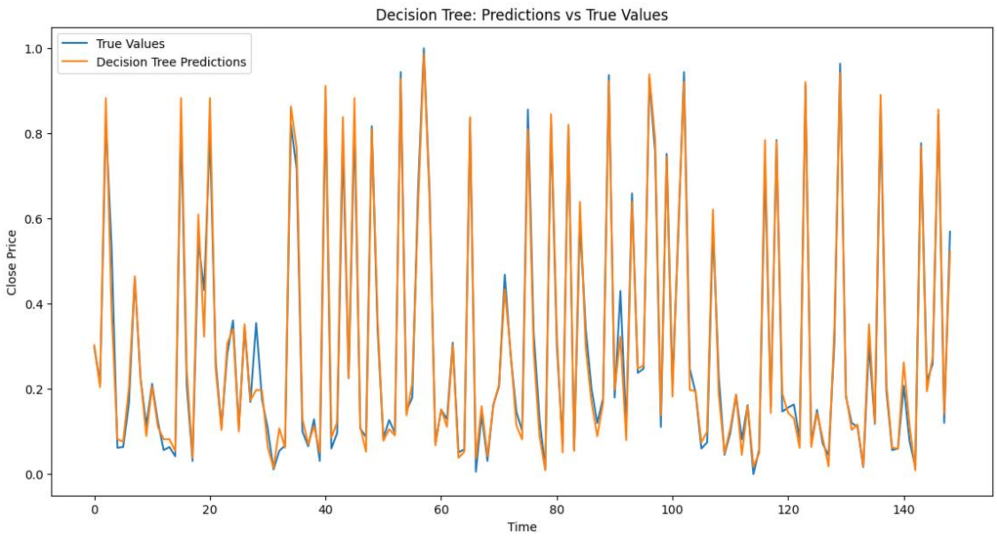
This is a comparison between predicted and actual values from a Long Short-Term Memory (LSTM) model. Here's a breakdown:

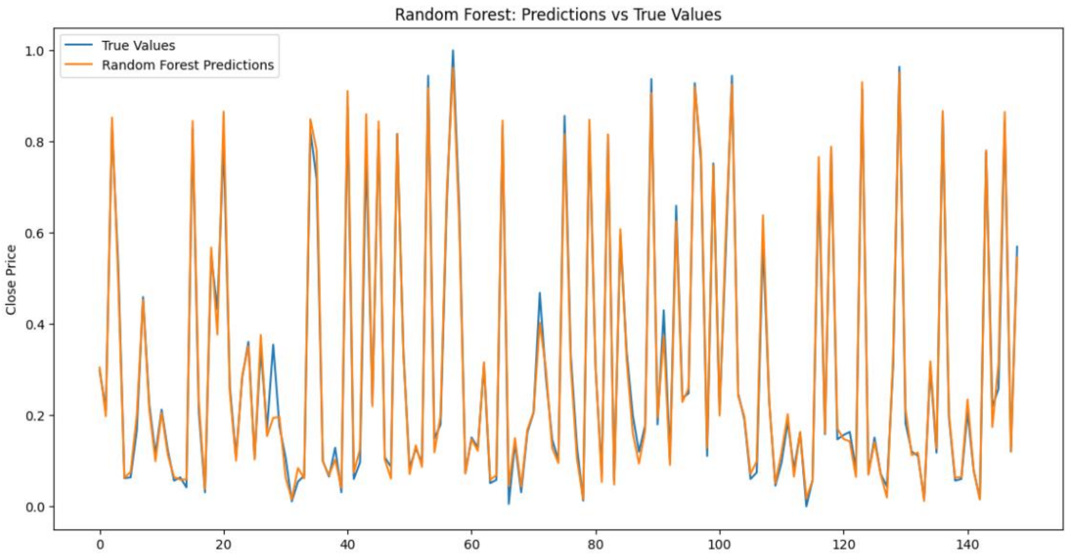
- **Title:** LSTM: Predictions vs True Values
- **Labels:**
  - True Values: The actual data points.
  - LSTM Predictions: The data points predicted by the LSTM model.
- **Axes:**
  - Y-axis: Represents inflation expectations.
  - X-axis: Likely time steps (unspecified units).

The graph shows both the true values and the LSTM's predictions for inflation expectations. It appears the LSTM predictions generally follow the trend of the true values, however it's difficult to say definitively from this limited view whether the LSTM is consistently more accurate.

For a more comprehensive interpretation, we'd need to see additional information like:

- A wider range of data points.
- How well the LSTM performs statistically compared to other forecasting methods.





Purpose:

This function transforms a time series dataset into sequences of features and corresponding target values, which is useful for training models like RNNs or LSTMs in predicting future values based on past observations.

Inputs:

1. The input dataset as a NumPy array, where each row represents a time step and each column represents a feature. The last column should be the target variable for the sequence.
2. The index of the target column within the data array, specifying which column contains the value to be predicted.
3. The number of time steps to include in each sequence.

Outputs:

1. A 3D NumPy array with dimensions (num samples, sequence length, num features). Each slice along the first dimension represents a sequence of sequence length time steps, containing num features.
2. A 1D NumPy array containing the target values corresponding to each sequence. The target for each sequence is the value from the target col at the time step immediately following the end of the sequence.

Process:

- **num_samples** is calculated as the total number of sequences that can be extracted from the data, given the sequence length.
- **num_features** is the number of columns (features) in the dataset.
- The function initializes arrays sequences and labels to store the sequences and target values.
- A loop iterates over the data to populate these arrays, slicing the input data into sequences of length sequence length and assigning the corresponding target value from the target col.

⇨ R

Data Loading and Preparation

1. **Libraries and Data Loading:**
   - The code installs and loads necessary libraries for financial data analysis and reads data for Wipro Limited from a CSV file.
   - It converts the 'Date' column to Date type and the data to an XTS object.
2. **Plotting:**
   - The adjusted close price is plotted over time to visualize the stock's price movement.
3. **Time Series Conversion:**
   - The data is converted to a monthly frequency and then to a TS (time series) object.
4. **Decomposition:**
   - The time series is decomposed into its trend, seasonal, and random components, and these components are plotted.

Forecasting Models

1. **Holt-Winters Exponential Smoothing:**
   - The time series data is split into training and test sets (80-20 split).
   - The Holt-Winters model is applied to the training data and forecasted for the test period.
   - The Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-squared are computed to evaluate the model's performance.
2. **ARIMA Model:**
   - An ARIMA model is fitted to the training data.
   - The model's forecast for the test period is plotted and evaluated similarly to the Holt-Winters model.
   - The model summary and forecast for the next period are printed.

Machine Learning Models for Multivariate Forecasting

1. **Data Preparation:**
   - Features and target are selected, and numeric features are normalized.
   - Data is scaled and prepared for sequence generation.
2. **Sequence Generation:**
   - A function (create sequences) is defined to generate sequences of features and corresponding target values.
   - The data is split into training and test sets based on sequences.
3. **LSTM Model:**
   - A Long Short-Term Memory (LSTM) model is built and trained on the sequence data.
   - The model's performance is evaluated using RMSE, MAE, MAPE, and R-squared.
   - True and predicted values are plotted to visualize the model's performance.
4. **Decision Tree and Random Forest:**
   - Features are flattened for use with decision tree and random forest models.
   - Models are trained, and predictions are evaluated using RMSE, MAE, MAPE, and R-squared.
   - Predictions and true values are plotted for comparison.

Visualization:

- **Decision Tree and Random Forest Predictions:**
    - The code plots predictions from decision tree and random forest models against true values.
    - It also plots both models' predictions together to compare their performance visually.

Summary:

- **Holt-Winters and ARIMA Models:** These traditional time series models are applied to the training data and evaluated on the test data. The Holt-Winters model is suitable for seasonal data, while ARIMA is more general.
- **LSTM Model:** This deep learning model captures long-term dependencies in the time series data. It is evaluated using multiple metrics and compared to true values.
- **Decision Tree and Random Forest:** These machine learning models are used for multivariate forecasting. Their predictions are evaluated and visualized to assess performance.

The code demonstrates a comprehensive approach to time series forecasting, utilizing traditional statistical methods and advanced machine learning techniques. By evaluating and comparing multiple models, it aims to find the best approach for predicting stock prices.