Exp no: 7 To implement program for decomposing time series data into trend and seasonality.

Date: 20/03/25

\*\*Objectives:\*\*

The objective of this experiment is to analyze stock market time series data using smoothing techniques and rolling statistics. The experiment applies data cleaning, seasonal differencing, smoothing using SES and HWES, and moving average calculations to reveal trends and enhance forecasting insights.

\*\*Background/Scope:\*\*

Time series analysis is crucial in financial data analytics. By using smoothing techniques like Simple Exponential Smoothing (SES) and Holt-Winters Exponential Smoothing (HWES), trends in data can be visualized with reduced noise. Additionally, rolling averages help capture short- and long-term movements, aiding traders and analysts in decision-making.

\*\*Steps for Time Series Smoothing and Trend Analysis:\*\*

\*\*Step 1: Load the Dataset\*\*

import pandas as pd

# Load dataset

df = pd.read\_csv("market.csv")

df.head()

\*\*Step 2: Data Cleaning and Preprocessing\*\*

# Convert 'Date' to datetime

df['Date'] = pd.to\_datetime(df['Date'])

# Remove missing and duplicate values

df.dropna(subset=['Date', 'Close', 'Adj Close', 'Open', 'High', 'Low', 'Volume'], inplace=True)

df.drop\_duplicates(subset=['Date', 'Ticker'], inplace=True)

# Outlier removal using IQR

Q1 = df['Close'].quantile(0.25)

Q3 = df['Close'].quantile(0.75)

IQR = Q3 - Q1

df = df[(df['Close'] >= (Q1 - 1.5 \* IQR)) & (df['Close'] <= (Q3 + 1.5 \* IQR))]

\*\*Step 3: Visualize Stock Trends by Ticker\*\*

import matplotlib.pyplot as plt

cols = ['Close', 'Adj Close', 'Open', 'High', 'Low', 'Volume']

fig, axs = plt.subplots(3, 2, figsize=(15, 16))

fig.suptitle('Data by ticker type')

for i, col in enumerate(cols):

row = i // 2

col\_idx = i % 2

for ticker, data in df.groupby('Ticker'):

axs[row, col\_idx].plot(data['Date'], data[col], label=ticker)

axs[row, col\_idx].set\_title(col)

axs[row, col\_idx].set\_xlabel('Date')

axs[row, col\_idx].set\_ylabel('Price')

axs[row, col\_idx].legend(loc='right')

axs[row, col\_idx].tick\_params(axis='x', rotation=45)

plt.tight\_layout()

plt.subplots\_adjust(wspace=0.3, hspace=0.8)

plt.show()

\*\*Step 4: Apply Seasonal Differencing\*\*

df['seasonal\_first\_difference'] = df['Volume'] - df['Volume'].shift(12)

df['seasonal\_first\_difference'].fillna(0, inplace=True)

\*\*Step 5: Apply Smoothing Techniques\*\*

from statsmodels.tsa.holtwinters import SimpleExpSmoothing, ExponentialSmoothing

# Simple Exponential Smoothing

df['SES'] = SimpleExpSmoothing(df['seasonal\_first\_difference']).fit(smoothing\_level=0.5, optimized=False).fittedvalues

# Holt-Winters Exponential Smoothing

df['HWES'] = ExponentialSmoothing(df['seasonal\_first\_difference'], trend='add', seasonal='add', seasonal\_periods=7).fit().fittedvalues

\*\*Step 6: Visualize Smoothing Techniques\*\*

plt.figure(figsize=(12, 6))

plt.plot(df['Volume'], label='Original')

plt.title('Smoothing Techniques Comparison')

plt.legend()

plt.show()

plt.figure(figsize=(12, 6))

plt.plot(df['SES'], label='Simple Exponential Smoothing', color='red')

plt.title('Simple Exponential Smoothing')

plt.legend()

plt.show()

plt.figure(figsize=(12, 6))

plt.plot(df['HWES'], label='Holt-Winters Exponential Smoothing', color='green')

plt.title('Holt-Winters Exponential Smoothing')

plt.legend()

plt.show()

\*\*Step 7: Resample Data by Week\*\*

weekly\_data = df.resample('W', on='Date').mean(numeric\_only=True)

weekly\_data.head()

\*\*Step 8: Calculate and Plot Rolling Averages (SMA 20 and SMA 50)\*\*

sma\_20 = df['High'].rolling(window=20).mean()

sma\_50 = df['High'].rolling(window=50).mean()

priceSma\_df = pd.DataFrame({

'High': df['High'],

'SMA 20': sma\_20,

'SMA 50': sma\_50

})

priceSma\_df.plot(figsize=(12, 6))

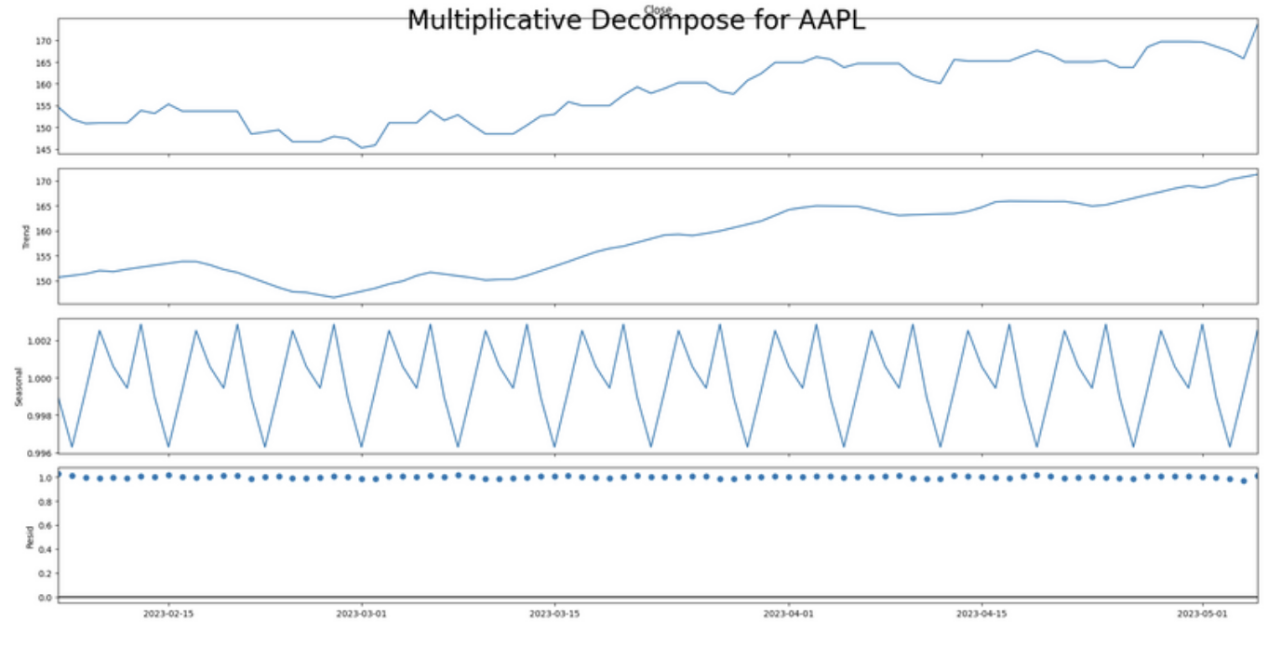
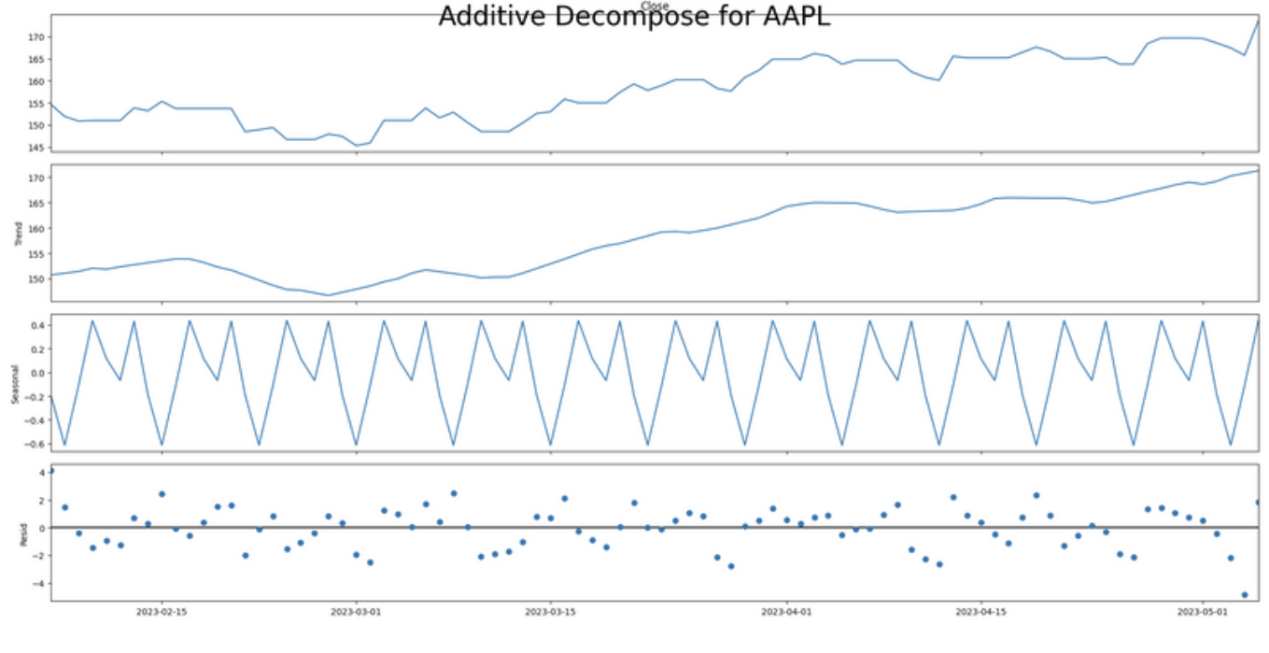
plt.title('SMA 20 vs SMA 50')

plt.xlabel('Date')

plt.ylabel('High Price')

plt.grid(True)

plt.show()



\*\*Result:\*\*

Thus, the stock market data was successfully analyzed using smoothing and rolling statistics techniques. SES and HWES revealed short-term and seasonal trends, while SMA highlighted long- and short-term market behaviors for better insight and forecasting.