**Implement programs for estimating & eliminating trend in time series data – aggregation, smoothing.**

**EX:No.4 DATE:1/02/25**

# AIM:

To Implement programs for estimating & eliminating trend in time series data – aggregation, smoothing..

## OBJECTIVE:

To estimate and remove trends in time-series air pollution data using aggregation and smoothing techniques.

## BACKGROUND:

* Time series data often has trends that affect analysis.
* **Aggregation** (e.g., monthly/yearly averaging) helps identify patterns.
* **Smoothing** (e.g., moving average, exponential smoothing) removes fluctuations.
* Trend elimination improves forecasting and stationarity.

## SCOPE OF THE PROGRAM:

* Load and clean air pollution data (2012-2021).
* Apply **aggregation** (monthly/yearly averages) to estimate trends.
* Use **moving average smoothing** to reduce noise.
* Apply **exponential smoothing** to highlight trends

**CODE:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from statsmodels.tsa.api import SimpleExpSmoothing

def load\_data(filename):

    df = pd.read\_csv("/content/MSFT.csv")

    # Set the 'Date' column as the index and convert it to DatetimeIndex

    df['Date'] = pd.to\_datetime(df['Date'])  # Convert 'Date' column to datetime objects

    df.set\_index('Date', inplace=True)  # Set 'Date' column as index

    return df

def aggregate\_data(df, freq='W'):

    return df.resample(freq).mean()

def moving\_average\_smoothing(df, window=10):

    return df.rolling(window=window).mean()

def exponential\_smoothing(df, alpha=0.2):

    model = SimpleExpSmoothing(df).fit(smoothing\_level=alpha, optimized=False)

    return model.fittedvalues

def plot\_results(original, smoothed, title):

    plt.figure(figsize=(12, 6))

    plt.plot(original, label='Original', alpha=0.6)

    plt.plot(smoothed, label='Smoothed', linewidth=2)

    plt.legend()

    plt.title(title)

    plt.show()

# Example usage

data\_file = 'stock\_prices.csv'  # Replace with your dataset

df = load\_data(data\_file)

closing\_prices = df['Close']

# Aggregation

weekly\_data = aggregate\_data(df, 'W')

monthly\_data = aggregate\_data(df, 'M')

# Smoothing

ma\_smoothed = moving\_average\_smoothing(closing\_prices, window=10)

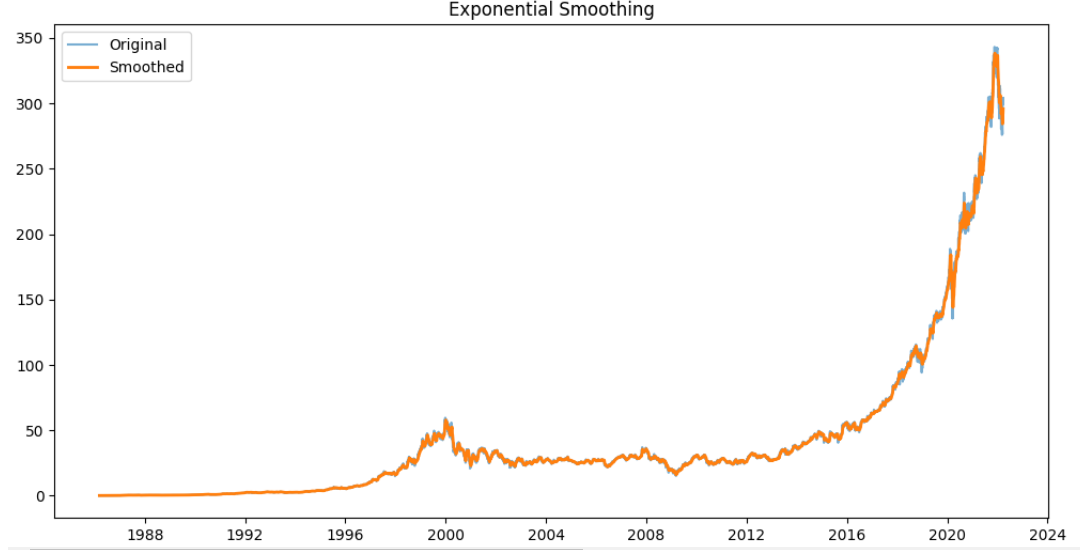
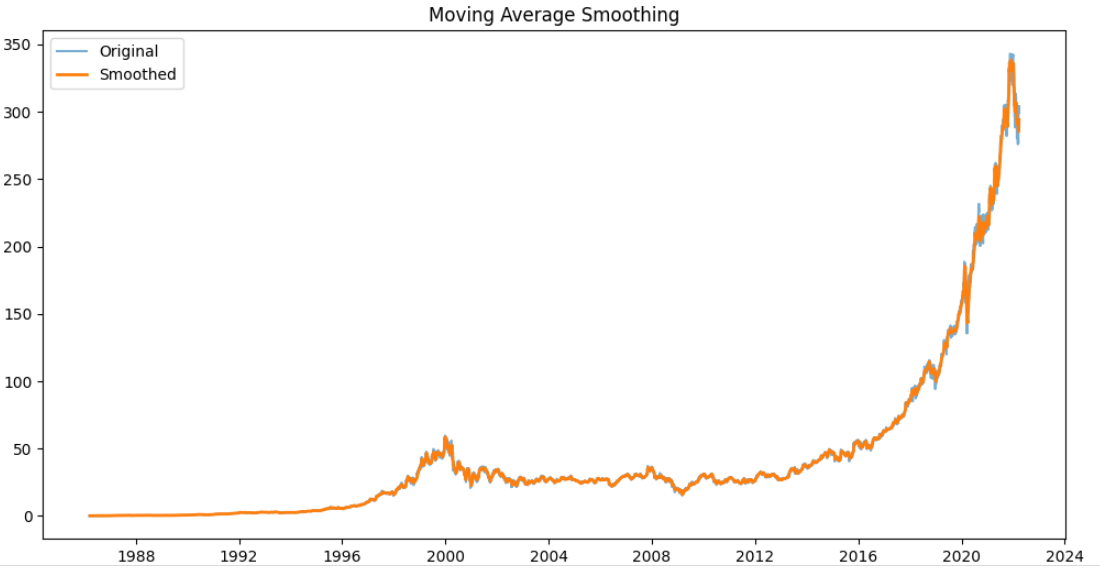
exp\_smoothed = exponential\_smoothing(closing\_prices, alpha=0.2)

# Plot results

plot\_results(closing\_prices, ma\_smoothed, 'Moving Average Smoothing')

plot\_results(closing\_prices, exp\_smoothed, 'Exponential Smoothing')

# OUTPUT:

****

**RESULT:**

Thus, the program using the time series data implementation has been done successfully.