

EX:No.4

DATE:1/02/25

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

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 supermarket sales dataset(2012-2021).
- Apply **aggregation** (monthly/yearly averages) to estimate trends.
- Use a **moving average dataset**.
- Apply **exponential smoothing** to highlight trends

CODE:

```
import pandas as pd
import matplotlib.pyplot as plt
```

```
# Load the dataset
```

```
df = pd.read_csv('supermarket_sales - Sheet1.csv')
```

```
# Step 1: Convert 'Date' column to datetime format
```

```
df['Date'] = pd.to_datetime(df['Date'])
```

```
# Step 2: Aggregate total sales by date
```

```
daily_sales = df.groupby('Date')['Total'].sum().reset_index()
```

```
# Step 3: Apply smoothing using Moving Averages
```

```
daily_sales['MA_7'] = daily_sales['Total'].rolling(window=7).mean() # 7-day moving average
```

```
daily_sales['MA_14'] = daily_sales['Total'].rolling(window=14).mean() # 14-day moving average
```

```
# Step 4: Plot the original and smoothed data
```

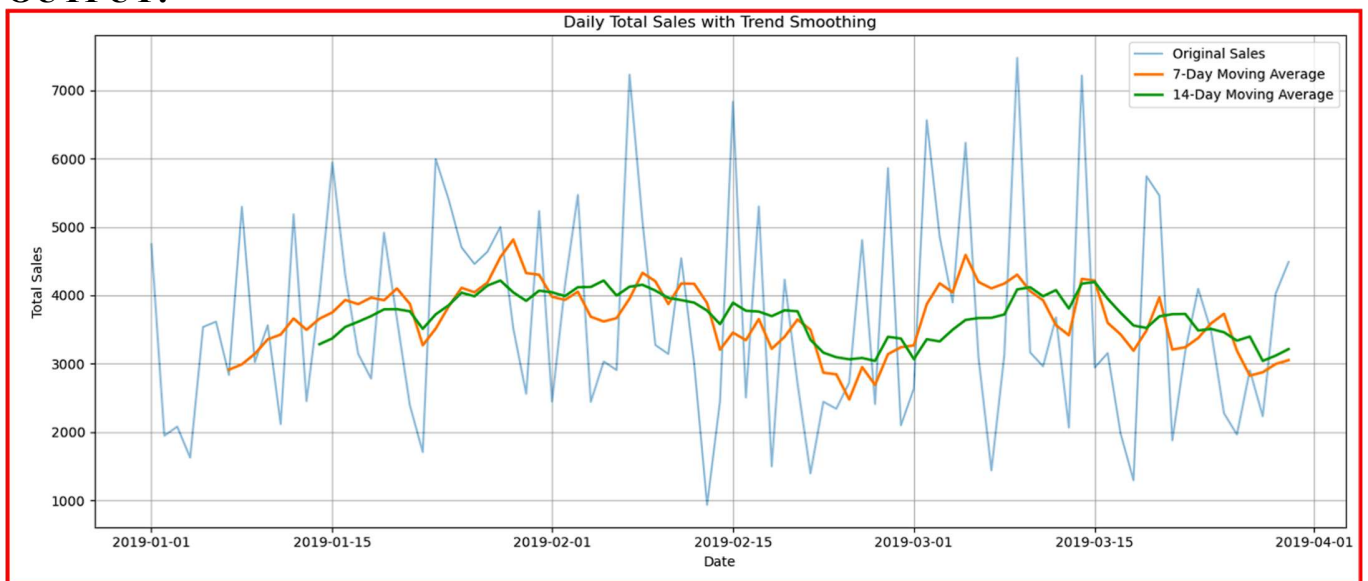
```
plt.figure(figsize=(14, 6))
```

```
plt.plot(daily_sales['Date'], daily_sales['Total'], label='Original Sales', alpha=0.5)
```

```
plt.plot(daily_sales['Date'], daily_sales['MA_7'], label='7-Day Moving Average', linewidth=2)
```

```
plt.plot(daily_sales['Date'], daily_sales['MA_14'], label='14-Day Moving Average', linewidth=2)
plt.title('Daily Total Sales with Trend Smoothing')
plt.xlabel('Date')
plt.ylabel('Total Sales')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

OUTPUT:



RESULT:

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