Estimating & eliminating trend in time series data-aggregation, smoothing

Aim

To implement programs for estimating & eliminating trend in time series data-aggregation, smoothing.

Procedure

1. Import libraries like `pandas` and `matplotlib`.
2. Load the supermarket sales dataset using `pandas.read_csv()`.
3. Convert the Date column to datetime format and sort the data.
4. Set the Date column as index to work with time series operations.
5. Aggregate the data (e.g., daily to monthly sales) using `resample()`.
6. Smooth the data using a rolling mean (e.g., 7-day window) to estimate the trend.
7. Detrend the data by subtracting the rolling mean from the original values and plot the results.

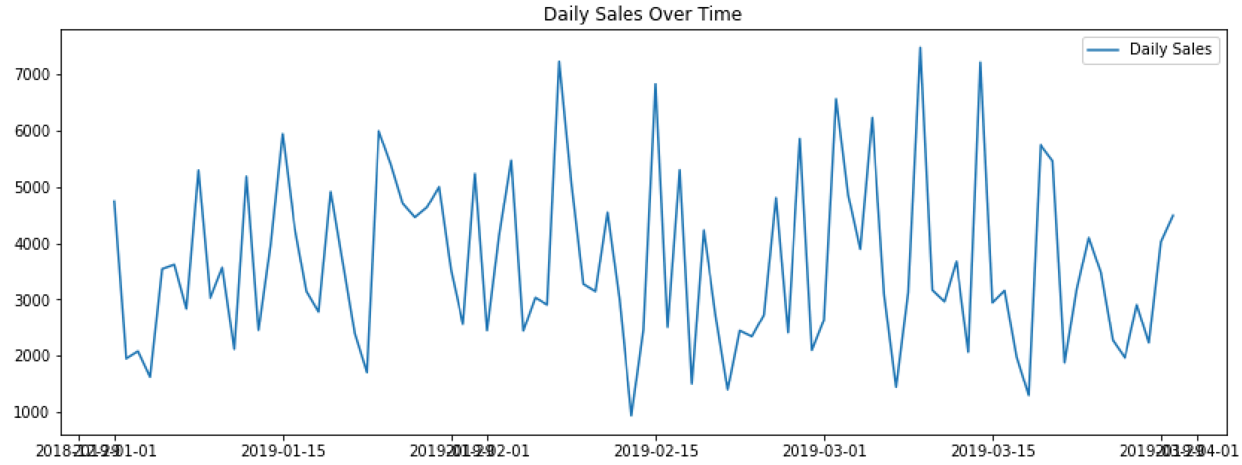
Code

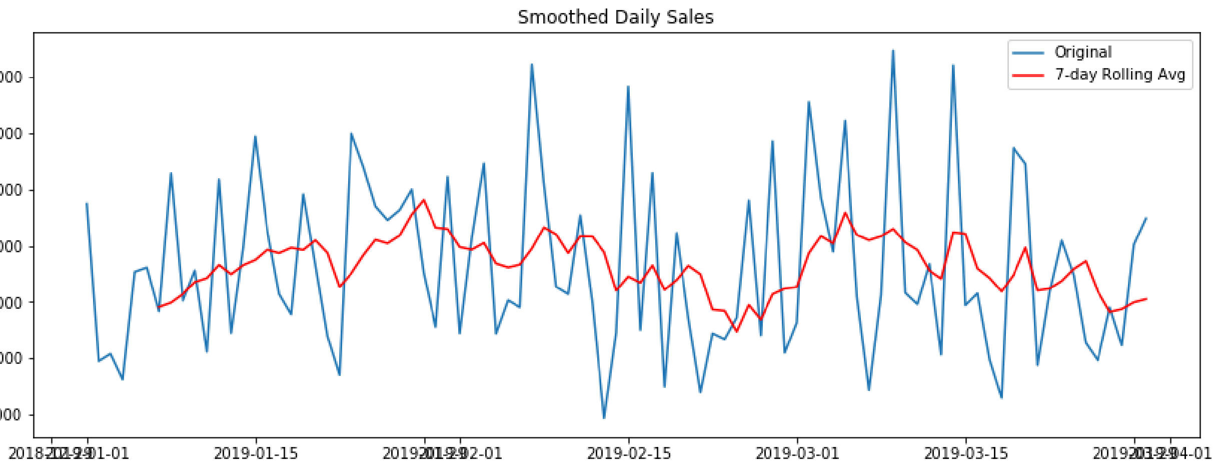
```
 import pandas as pd

import matplotlib.pyplot as plt

# Load dataset

df = pd.read_csv(r"C:\Users\Poong\Downloads\supermarket_sales - Sheet1.csv")

# Parse date and sort

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

df = df.sort_values('Date')

# Set date as index
```

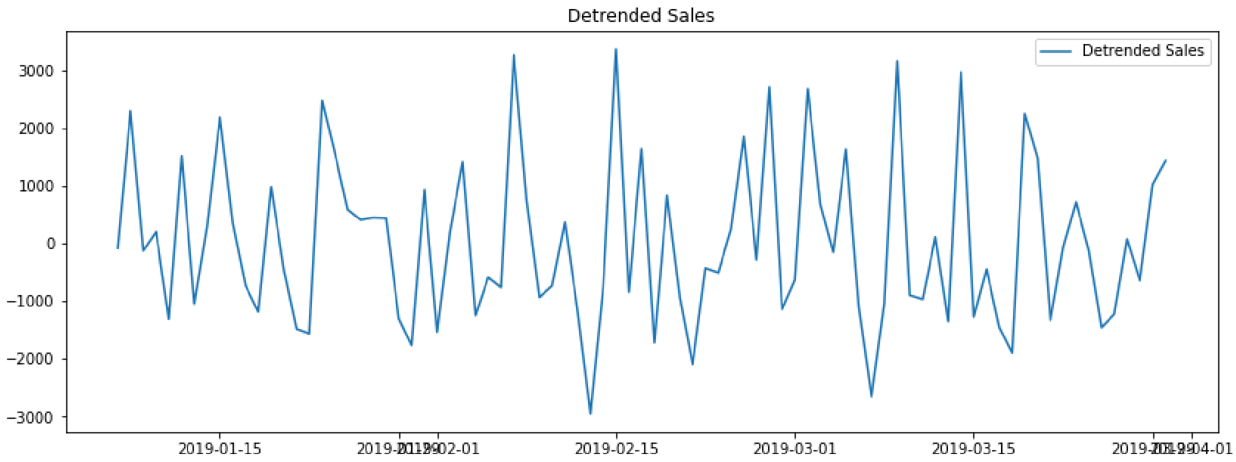
```
df.set_index('Date', inplace=True)
# Aggregate sales per day
daily_sales = df.resample('D').sum()['Total']
# Aggregate monthly sales
monthly_sales = daily_sales.resample('M').sum()
# Plot original daily sales
plt.figure(figsize=(14, 5))
plt.plot(daily_sales, label='Daily Sales')
plt.title('Daily Sales Over Time')
plt.legend()
plt.show()
# Smooth trend with rolling average (7-day window)
rolling_avg = daily_sales.rolling(window=7).mean()
# Plot smoothed sales
plt.figure(figsize=(14, 5))
plt.plot(daily_sales, label='Original')
plt.plot(rolling_avg, label='7-day Rolling Avg', color='red')
plt.title('Smoothed Daily Sales')
plt.legend()
plt.show()
# Detrend: Original - Rolling Average
```

```
detrended = daily_sales - rolling_avg

# Plot detrended data

plt.figure(figsize=(14, 5))

plt.plot(detrended, label='Detrended Sales')

plt.title('Detrended Sales')

plt.legend()

plt.show()
```

Output







Result

The above program has been successfully executed.