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

**AIM:**

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

**PROCEDURE:**

**1.**Import the necessary libraries:

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from statsmodels.tsa.api import SimpleExpSmoothing

**2.** Load the dataset :

file\_path = "/content/autism\_screening.csv" *# Update with actual file path*

df = pd.read\_csv(file\_path)

**3.**Ensure relevant columns exist.

required\_columns = {'age', 'class/asd'}

if not required\_columns.issubset(df.columns):

raise ValueError(f"Dataset is missing required columns: {required\_columns - set(df.columns)}")

**4.** Aggregate ASD classification per age → Compute rolling mean (moving average) → Apply exponential smoothing for trend estimation.

df\_agg = df.groupby("age")["class/asd"].mean().reset\_index()

df\_agg["Moving\_Avg"] = df\_agg["class/asd"].rolling(window=5, center=True).mean()

exp\_smooth = SimpleExpSmoothing(df\_agg["class/asd"]).fit(smoothing\_level=0.2, optimized=False)

df\_agg["Exp\_Smooth"] = exp\_smooth.fittedvalues

**5.** Create subplots for visualization:Original Data (Aggregated);Moving Average Trend;ExponentialSmoothing;Detrended Data

fig, axes = plt.subplots(2, 2, figsize=(12, 10))

*# Original Data (Aggregated)*

axes[0, 0].plot(df\_agg["age"], df\_agg["class/asd"], marker="o", linestyle="-", alpha=0.6, label="Aggregated Data")

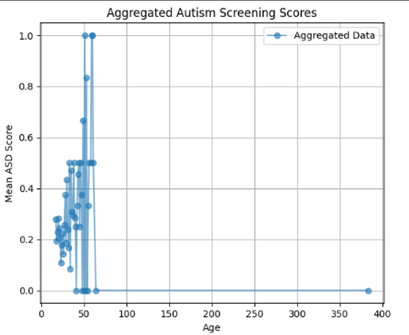
axes[0, 0].set\_title("Aggregated Autism Screening Scores")

axes[0, 0].set\_xlabel("Age")

axes[0, 0].set\_ylabel("Mean ASD Score")

axes[0, 0].legend()

axes[0, 0].grid(True)



*# Moving Average Trend*

axes[0, 1].plot(df\_agg["age"], df\_agg["Moving\_Avg"], color="blue", linewidth=2, label="Moving Average")

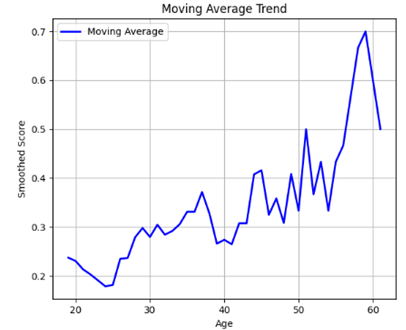
axes[0, 1].set\_title("Moving Average Trend")

axes[0, 1].set\_xlabel("Age")

axes[0, 1].set\_ylabel("Smoothed Score")

axes[0, 1].legend()

axes[0, 1].grid(True)



***# Exponential Smoothing***

axes[1, 0].plot(df\_agg["age"], df\_agg["Exp\_Smooth"], color="green", linestyle="--", linewidth=2, label="Exponential Smoothing")

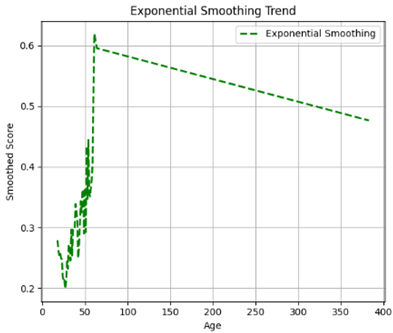
axes[1, 0].set\_title("Exponential Smoothing Trend")

axes[1, 0].set\_xlabel("Age")

axes[1, 0].set\_ylabel("Smoothed Score")

axes[1, 0].legend()

axes[1, 0].grid(True)



*# Detrended Data*

axes[1, 1].plot(df\_agg["age"], df\_agg["Detrended"], color="red", linestyle="dotted", alpha=0.8, label="Detrended Data")

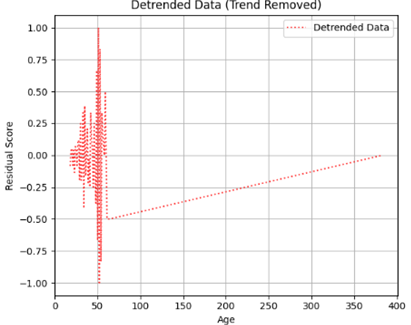
axes[1, 1].set\_title("Detrended Data (Trend Removed)")

axes[1, 1].set\_xlabel("Age")

axes[1, 1].set\_ylabel("Residual Score")

axes[1, 1].legend()

axes[1, 1].grid(True)



*# Adjust layout*

plt.tight\_layout()

plt.show()

**RESULT:**

The program to check stationary on the autism screening dataset has been implemented successfully.