| **EXP:4**  **20/02/2025** | **Checking Time Series Stationarity using Statistical Tests** |
| --- | --- |

**AIM:**

To determine whether a time series dataset is stationary using the Augmented Dickey-Fuller (ADF) Test and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test in Python.

**PROCEDURE:**

**1) Load Data & Preprocess**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from statsmodels.tsa.stattools import adfuller, kpss

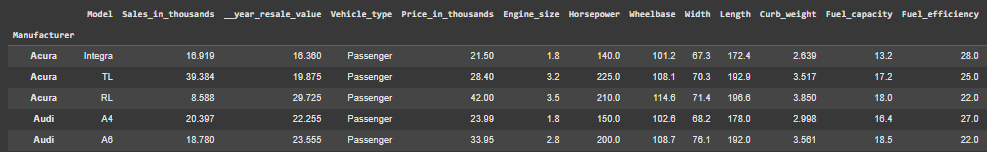
data = pd.read\_csv("/content/sample\_data/Car\_sales.csv", header=0, index\_col=0)

data = data.dropna()

sales\_data = data['Sales\_in\_thousands']

data.head()

**OUTPUT**

****

**2)Visual Inspection - Rolling Mean & Standard Deviation**

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

**rolling\_mean = sales\_data.rolling(window=12).mean()**

**rolling\_std = sales\_data.rolling(window=12).std()**

**plt.plot(sales\_data, label='Original Data')**

**plt.plot(rolling\_mean, label='Rolling Mean', color='red')**

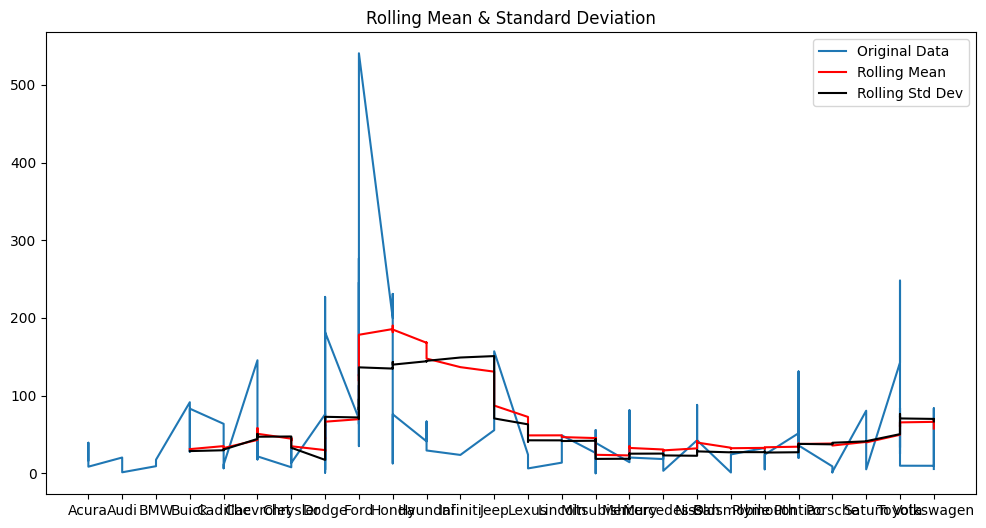
**plt.plot(rolling\_std, label='Rolling Std Dev', color='black')**

**plt.legend()**

**plt.title('Rolling Mean & Standard Deviation')**

**plt.show()**

**OUTPUT**

****

**3) Augmented Dickey-Fuller (ADF) Test**

print("\n### Augmented Dickey-Fuller (ADF) Test ###")

adf\_test = adfuller(sales\_data)

print(f"ADF Statistic: {adf\_test[0]}")

print(f"p-value: {adf\_test[1]}")

print("Critical Values:")

for key, value in adf\_test[4].items():

print(f"{key}: {value}")

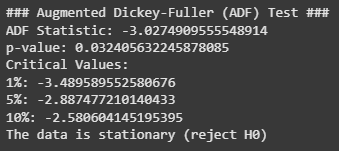
if adf\_test[1] <= 0.05:

print("The data is stationary (reject H0)")

else:

print("The data is non-stationary (fail to reject H0)")

**OUTPUT**

****

**4)KPSS Test**

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

**5)KPSS Test (Checks if Data is Stationary)**

from statsmodels.tsa.stattools import kpss

print("\n### KPSS Test ###")

kpss\_test = kpss(sales\_data, regression='c', nlags='auto')

print(f"KPSS Statistic: {kpss\_test[0]}")

print(f"p-value: {kpss\_test[1]}")

print("Critical Values:")

for key, value in kpss\_test[3].items():

print(f"{key}: {value}")

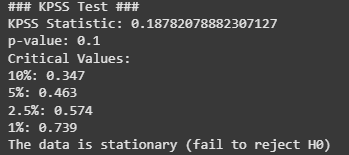
if kpss\_test[1] > 0.05:

print("The data is stationary (fail to reject H0)")

else:

print("The data is non-stationary (reject H0)")

**OUTPUT**

****

**RESULT:**

Thus the program has been executed successfully