**EX:No.10 VAR MODEL FOR MULTIVARIATE TIME SERIES**

**221501010**

**AIM :** To create a variable autoregression model for multivariate time series data.

**PROCEDURE:**

#### **1. Prepare the Data**

#### **2. Make the Data Stationary**

#### **3. Select the Lag Order (p)**

#### **4. Fit the VAR Model**

#### **5. Diagnose the Model**

#### **6. Forecast Future Values**

**IMPLEMENTATION :**

# Step 1: Import libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from statsmodels.tsa.api import VAR

from sklearn.metrics import mean\_squared\_error

from math import sqrt

# Step 2: Load the dataset

df = pd.read\_csv('/content/ch3\_airline\_passengers.csv')

df.columns = ['Month', 'Passengers']

df['Month'] = pd.to\_datetime(df['Month'])

df.set\_index('Month', inplace=True)

# Step 3: Create synthetic second variable (e.g., rolling mean or shifted)

df['Passengers\_shifted'] = df['Passengers'].shift(1).fillna(method='bfill')

# Step 4: Plot the variables

df[['Passengers', 'Passengers\_shifted']].plot(figsize=(12, 5), title='Multivariate Time Series - Airline Passengers')

plt.show()

# Step 5: Split into train and test sets

train\_size = int(len(df) \* 0.8)

train, test = df.iloc[:train\_size], df.iloc[train\_size:]

# Step 6: Fit VAR model

model = VAR(train)

fitted\_model = model.fit(maxlags=15, ic='aic')  # auto-select best lag using AIC

# Step 7: Forecast

forecast\_input = train.values[-fitted\_model.k\_ar:]

forecast\_steps = len(test)

forecast = fitted\_model.forecast(y=forecast\_input, steps=forecast\_steps)

# Step 8: Convert forecast to DataFrame

forecast\_df = pd.DataFrame(forecast, index=test.index, columns=['Passengers\_forecast', 'Passengers\_shifted\_forecast'])

# Step 9: Plot actual vs forecast

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

plt.plot(test['Passengers'], label='Actual Passengers')

plt.plot(forecast\_df['Passengers\_forecast'], label='VAR Forecast', color='red')

plt.title('VAR Forecast vs Actual – Airline Passengers')

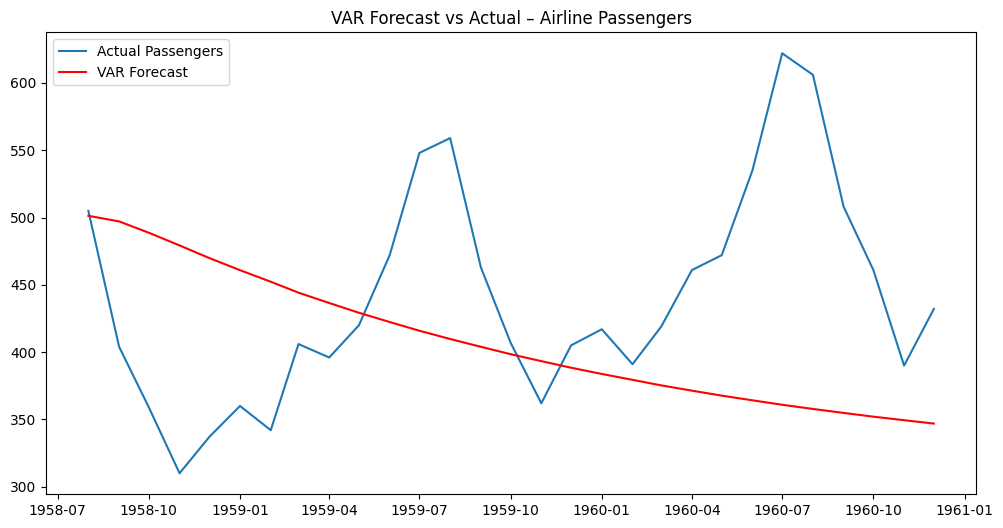
plt.legend()

plt.show()

# Step 10: Evaluate

rmse = sqrt(mean\_squared\_error(test['Passengers'], forecast\_df['Passengers\_forecast']))

print(f'RMSE (Passengers Forecast): {rmse:.2f}')

**OUTPUT:**

**RESULT :** Thus variable autoregression has been successfully implemented for multivariate time series data.