**EX:No.8 ARIMA FOR TIME SERIES FORECASTING**

**221501010**

**AIM :** To create an ARIMA model for time series forecasting.

**PROCEDURE:**

 Read the dataset and convert the date column to datetime format.

 Set the date column as the index for time series analysis.

 Aggregate the data to monthly mean (if not already monthly).

 Check for stationarity.

 Determine optimal ARIMA parameters (p, d, q).

 Fit the ARIMA model using the training data.

 Make predictions and compare with actual values.

**IMPLEMENTATION :**

**# Step 1: Import libraries**

import pandas as pd

import matplotlib.pyplot as plt

from statsmodels.tsa.arima.model import ARIMA

from sklearn.metrics import mean\_squared\_error

from math import sqrt

# Step 2: Load dataset

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

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

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

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

# Step 3: Plot original data

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

plt.plot(df['Passengers'], label='Airline Passengers')

plt.title('Original Time Series')

plt.xlabel('Date')

plt.ylabel('Passengers')

plt.legend()

plt.show()

# Step 4: Train-test split (80% train, 20% test)

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

train, test = df['Passengers'][:train\_size], df['Passengers'][train\_size:]

# Step 5: Fit ARIMA model

# ARIMA(p,d,q) -> p=AR lags, d=differencing, q=MA lags

model = ARIMA(train, order=(2, 1, 2))  # You can experiment with (p,d,q)

model\_fit = model.fit()

# Step 6: Forecast

forecast = model\_fit.forecast(steps=len(test))

forecast = pd.Series(forecast, index=test.index)

# Step 7: Plot actual vs forecast

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

plt.plot(train, label='Training Data')

plt.plot(test, label='Actual Test Data')

plt.plot(forecast, label='Forecast', color='red')

plt.title('ARIMA Forecast vs Actuals – Airline Passengers')

plt.xlabel('Date')

plt.ylabel('Passengers')

plt.legend()

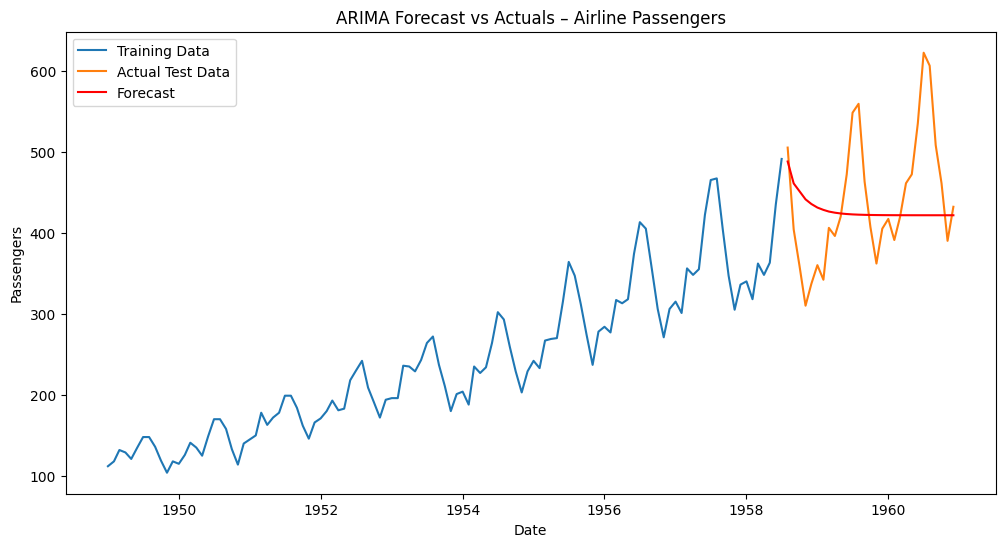
plt.show()

# Step 8: Evaluation

rmse = sqrt(mean\_squared\_error(test, forecast))

print(f'Root Mean Squared Error (RMSE): {rmse:.2f}')

**OUTPUT:**



**RESULT :** Thus ARIMA model has been successfully created for time series forecasting.