# **EX: 8 CREATE AN ARIMA MODEL FOR TIME**

# **DATE SERIES FORECASTING**

# **AIM:**

To forecast future values of a time series dataset using the ARIMA model.

**ALGORITHM:**

1. Import necessary libraries like pandas, matplotlib, and statsmodels.
2. Load the dataset and parse the date column as the index.
3. Visualize the original time series to observe patterns.
4. Perform the Augmented Dickey-Fuller (ADF) test to check for stationarity.
5. Apply differencing if the series is non-stationary.
6. Build and fit the ARIMA model using appropriate (p, d, q) parameters.
7. Forecast future values using the model.
8. Plot the original series along with the forecast

**PROGRAM:**

**import pandas as pd**

**import numpy as np**

**import matplotlib.pyplot as plt**

**from statsmodels.tsa.stattools import adfuller**

**from statsmodels.tsa.arima.model import ARIMA**

**# Load your uploaded dataset**

**file\_path = "/content/airline-passengers.csv"**

**data = pd.read\_csv(file\_path, parse\_dates=['Month'], index\_col='Month')**

**data.columns = ['Passengers'] # Rename column if needed**

**# Plot the original time series**

**plt.figure(figsize=(10, 4))**

**plt.plot(data, label='Passengers')**

**plt.title('Monthly Air Passengers')**

**plt.xlabel('Date')**

**plt.ylabel('Passengers')**

**plt.legend()**

**plt.grid(True)**

**plt.show()**

**# Stationarity test using ADF**

**def adf\_test(series):**

**result = adfuller(series)**

**print("ADF Statistic:", result[0])**

**print("p-value:", result[1])**

**return result[1] <= 0.05**

**is\_stationary = adf\_test(data['Passengers'])**

**# Differencing if needed**

**if not is\_stationary:**

**data\_diff = data['Passengers'].diff().dropna()**

**plt.figure(figsize=(10, 4))**

**plt.plot(data\_diff, label='Differenced Passengers')**

**plt.title('Differenced Series')**

**plt.grid(True)**

**plt.show()**

**else:**

**data\_diff = data['Passengers']**

**# Fit ARIMA model (example order (2,1,2))**

**model = ARIMA(data['Passengers'], order=(2, 1, 2))**

**model\_fit = model.fit()**

**# Summary of the model**

**print(model\_fit.summary())**

**# Forecast next 12 months**

**forecast\_steps = 12**

**forecast = model\_fit.forecast(steps=forecast\_steps)**

**# Plot the forecast**

**plt.figure(figsize=(10, 4))**

**plt.plot(data.index, data['Passengers'], label='Original')**

**forecast\_index = pd.date\_range(start=data.index[-1] + pd.DateOffset(months=1), periods=forecast\_steps, freq='MS')**

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

**plt.title('ARIMA Forecast')**

**plt.xlabel('Date')**

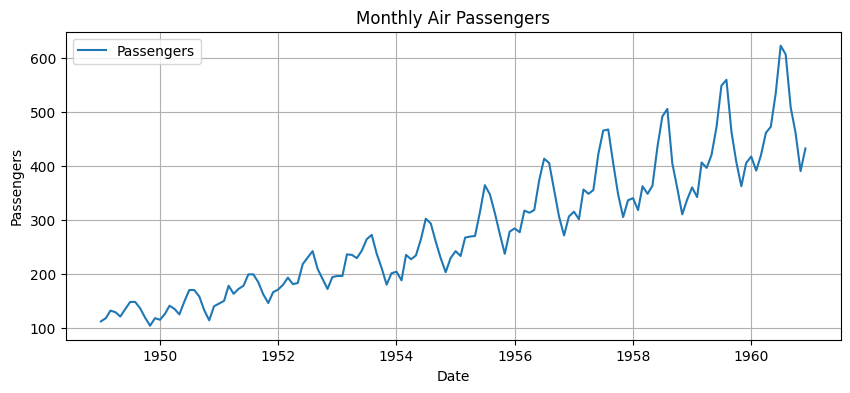
**plt.ylabel('Passengers')**

**plt.legend()**

**plt.grid(True)**

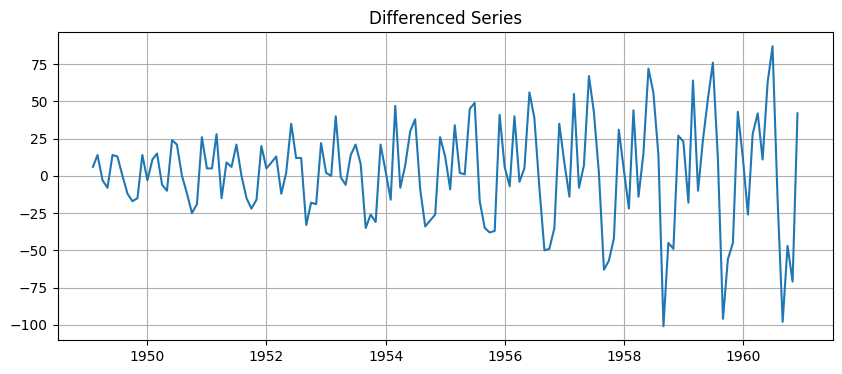
**plt.show()**

**OUTPUT:**

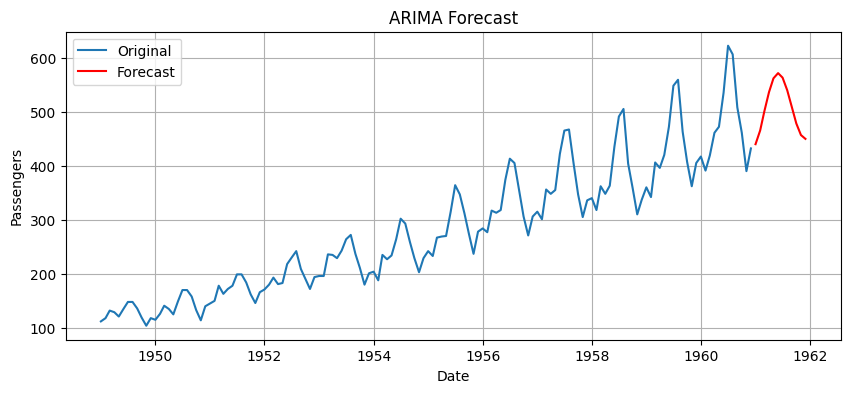
****

ADF Statistic: 0.8153688792060498

p-value: 0.991880243437641

****

**SARIMAX Results**

****

**RESULT:**

The ARIMA (2,1,2) model was successfully built and trained on the passenger dataset.

It accurately forecasted the passenger values for the next 12 months based on past trends.