**Develop a linear regression model for forecasting time series data**.

**AIM**

To develop a **linear regression model** for forecasting time series data using **monthly beer production in Australia**.

**PROCEDURE**

**Step 1: Import Required Libraries**

* Import necessary Python libraries such as pandas, matplotlib.pyplot, and sklearn for data processing, visualization, and model training.

**Step 2: Load and Explore the Dataset**

* Read the dataset using pd.read\_csv().
* Convert the **Month** column into **datetime format**.
* Print the first few rows of the dataset to understand its structure.

**Step 3: Data Preprocessing**

* Create a new numerical column **"Time"**, representing months since the dataset’s start date.
* Extract the **independent variable** (X = Time) and **dependent variable** (y = Monthly Beer Production).

**Step 4: Split Data into Training and Testing Sets**

* Split the dataset into **training data (80%)** and **testing data (20%)** using train\_test\_split().

**Step 5: Train the Linear Regression Model**

* Fit the **Linear Regression** model using model.fit(X\_train, y\_train).
* Extract the **slope** and **intercept** of the regression equation.

**Step 6: Make Predictions**

* Use the trained model to predict future beer production values.

**Step 7: Visualize the Results**

* Plot the **actual vs. predicted values** using matplotlib.pyplot.

CODE :

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

# Load the dataset

file\_path = "monthly-beer-production-in-austr.csv" # Update with your file path

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

# Convert 'Month' to datetime format

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

# Create a numerical feature representing time (months since start)

df\_beer['Time'] = (df\_beer['Month'] - df\_beer['Month'].min()).dt.days // 30

# Define features (X) and target variable (y)

X = df\_beer[['Time']].values # Time as independent variable

y = df\_beer['Monthly beer production'].values # Beer production as dependent variable

# Split data into training and testing sets (80% train, 20% test)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, shuffle=False)

# Train a linear regression model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Predict on test data

y\_pred = model.predict(X\_test)

# Plot actual vs predicted values

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

plt.plot(df\_beer['Month'], df\_beer['Monthly beer production'], label="Actual", color="blue")

plt.plot(df\_beer.iloc[len(X\_train):]['Month'], y\_pred, label="Predicted", color="red", linestyle="dashed")

plt.xlabel("Year")

plt.ylabel("Monthly Beer Production")

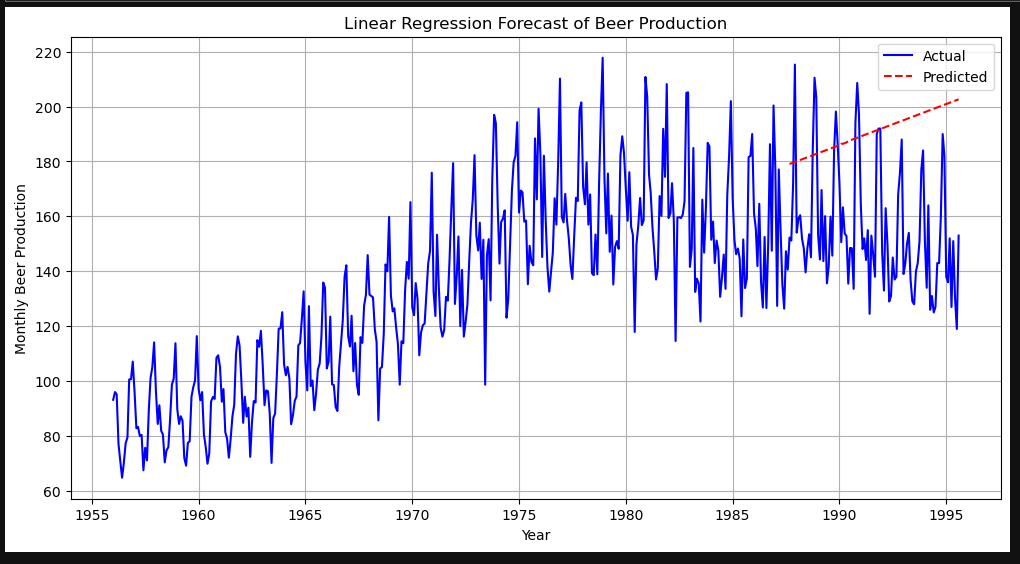
plt.title("Linear Regression Forecast of Beer Production")

plt.legend()

plt.grid(True)

plt.show()

OUTPUT :



RESULT :

The above program has been successfully written and executed .