

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** ($X = \text{Time}$) and **dependent variable** ($y = \text{Monthly Beer Production}$).

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")

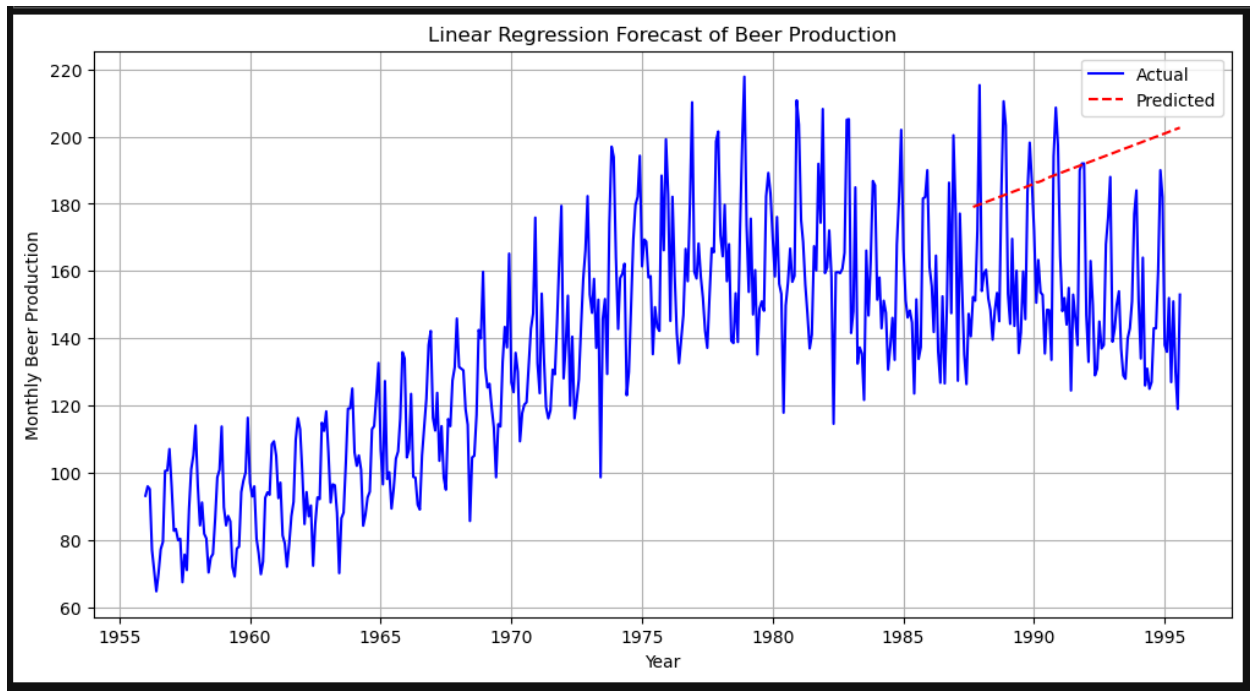
plt.title("Linear Regression Forecast of Beer Production")

plt.legend()

plt.grid(True)

plt.show()
```

OUTPUT :



RESULT :

The above program has been successfully written and executed .