EXP. NO: 2

Date: 9/8/25.

A PYTHON PROGRAM TO IMPLEMENT SIMPLE LINEAR REGRESSION USING LEAST SQUARE METHOD

Aim:

To implement a python program for constructing a simple linear regression using least square method.

Algorithm:

Step 1: Import necessary libraries:

* Pandas for data manipulation and matplotlib.pyplot for plotting

Step 2: Read the dataset:

* Use the pandas 'read_csv' function to read the dataset.

Step 3: prepare the data:

* Extract the independent variable (x) and dependent variable (y) from the dataset.
* Reshape x and y to 2D arrays if needed.

Step 4: Calculate the mean:

* Calculate the mean of x and y.

Step 5: Calculate the coefficients.

* Calculate the slope (m) using the formula:

$$m = \frac{\sum_{i=1}^{m} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - x)^2}$$

* Calculate the intercept (b) using formula: $b = \bar{y} - m\bar{x}$.

Step 6: Make predictions:

* Use the calculated slope and intercept to make Predictions for each X value.

$$\bar{y} = mx + b.$$

Program:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
data = pd.read.csv('headbrain.csv')
x,y = np.array(list(data['Head size(cm^3)']),
      np.array(list(data['Brain Weight(grams)']))
Print(x[:5], y[:5])
[4512 3738 4261 3777 4177] [1530 1297 1335 1282 159
def get_lines(x,y):
x_m, y_m = np.mean(x), np.mean(y)
Print(x_m, y_m)
x_d, y_d = x-x_m, y-y_m.
m = np.sum(x_d * y_d) / np.sum(x_d ** 2)
c = y_m - (m* x_m)
Print(m, c)
Return lambda x : m * x +c
lin = get_line(x,y)
X = np.linspace(np.min(x)-100, np.max(x)+100, 1000)
Y = np.array([lin(x) for x in X])
plt.plot(X,Y, color='red', label='Regression line').
plt.scatter(x,y, color='green', label='Scatter plot').
plt.xlabel('Head size (cm^3)')
plt.ylabel('Brain Weight (grams)') plt.legend()
plt.show().
```

Step 7 : Plot the regression line :

* Plot the original data points (x, y) as a scatter plot.
* Plot the regression line (x, predicted_y) as line plot.

Step 8 : Calculate the R-squared value :

* Calculate the total sum of squares $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$.
* Calculate the residual sum of squares (RSS) using the formula $RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$.
* R-squared value $R^2 = 1 - \frac{RSS}{TSS}$

Step 9 : Display the results.

* Print the slope, intercept, & R-squared value.

Step 10 : Compile the program :

* Combine all the steps into a Python program.

Result :

Thus the Python program to implement simple linear regression using least square method for the given head brain dataset is analyzed and the linear regression line is constructed successfully.

```
def get_error(line_func, x, y):
    y_m = np.mean(y)
    y_pred = np.array([line_func(i) for i in x])
    ss_t = np.sum((y - y_m) ** 2)
    ss_r = np.sum((y - y_pred) ** 2)
    return 1 - (ss_r / ss_t)
print(get_error(line, x, y))
```