Implementation of Univariate Linear Regression

Aim:

To implement univariate Linear Regression to fit a straight line using least squares.

Equipment's required:

1. Hardware – PCs
2. Anaconda – Python 3.7 Installation / Moodle-Code Runner

Algorithm:

1. Get the independent variable X and dependent variable Y.
2. Calculate the mean of the X -values and the mean of the Y -values.
3. Find the slope m of the line of best fit using the formula.

$$m = \frac{\sum_{i=1}^{n}(x_i-\overline{X})(y_i-\overline{Y})}{\sum_{i=1}^{n}(x_i-\overline{X})^2}$$

4. Compute the y -intercept of the line by using the formula: $b = \overline{Y} - m\overline{X}$
5. Use the slope m and the y -intercept to form the equation of the line.
6. Obtain the straight line equation Y=mX+b and plot the scatterplot.

Program

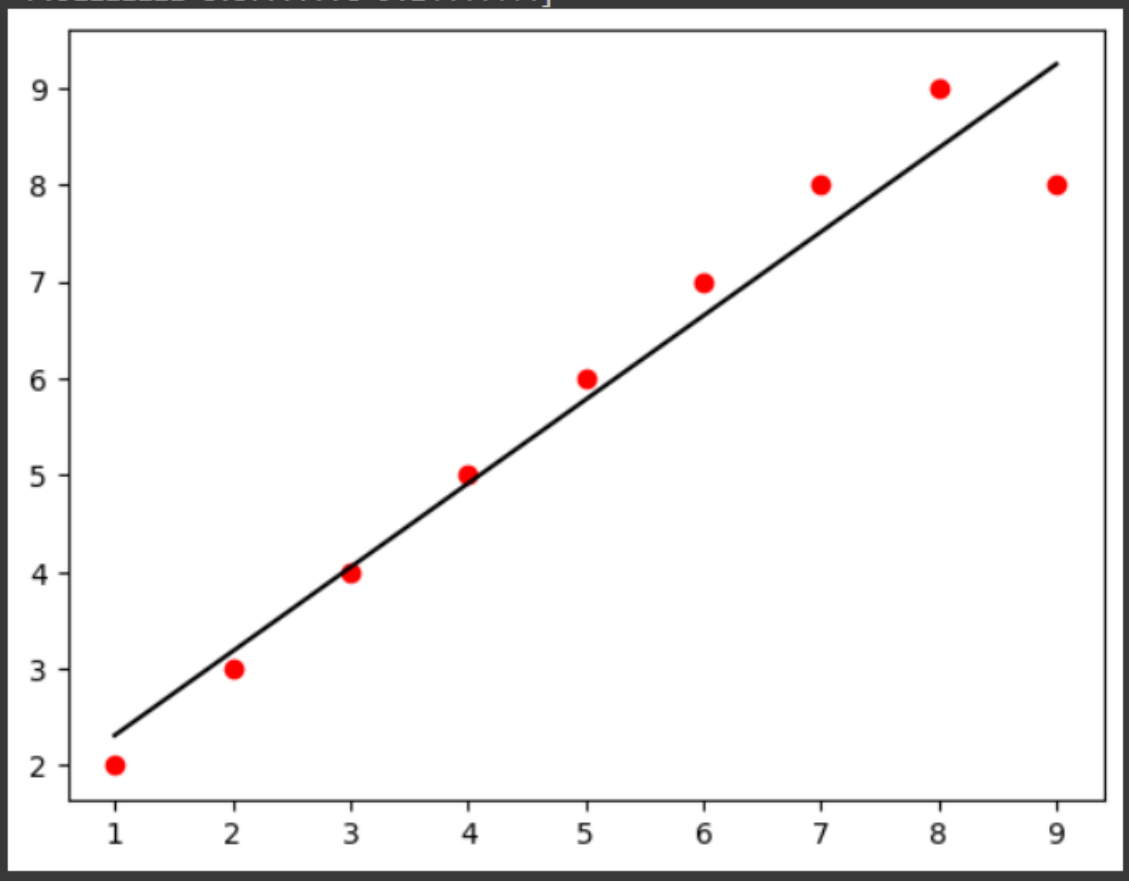
```
import numpy as np
X = np.array(eval(input()))
Y = np.array(eval(input()))
X_mean=np.mean(X)
Y_mean=np.mean(Y)
num=0
denom=0
for i in range(len(X)):
    num+=(X[i]-X_mean)*(Y[i]-Y_mean)
    denom+=(X[i]-X_mean)**2
    m=num/denom
    c=Y_mean-m*X_mean
print(m,c)
```

```
Y_=m*X+c
print(Y_pred)
import matplotlib.pyplot as plt
plt.scatter(X,Y,color='red')
plt.plot(X,Y_,color='black')
plt.show()
```

Output

```
import numpy as np
X = np.array(eval(input()))
Y = np.array(eval(input()))
X_mean=np.mean(X)
Y_mean=np.mean(Y)
num=0
denom=0
for i in range(len(X)):
    num+=(X[i]-X_mean)*(Y[i]-Y_mean)
    denom+=(X[i]-X_mean)**2
    m=num/denom
    c=Y_mean-m*X_mean
print(m,c)
Y_=m*X+c
print(Y_)
import matplotlib.pyplot as plt
plt.scatter(X,Y,color='red')
plt.plot(X,Y_,color='black')
plt.show()
```

```
[1,2,3,4,5,6,7,8,9]
[2,3,4,5,6,7,8,9,8]
0.8666666666666667 1.4444444444444438
[2.31111111 3.17777778 4.04444444 4.91111111 5.77777778 6.64444444
 7.51111111 8.37777778 9.24444444]
```



Result

Thus the univariate Linear Regression was implemented to fit a straight line using least squares.