Correlation and regression for data analysis

Ex.no:03

Date:

Name:Praisy nishitha

Register no:24900090

Aim :

To analyse given data using coeffificient of correlation and regression line

X	25	28	35	32	31	36	29	38	34	32
Y	43	46	49	41	36	32	31	30	33	39

Software required :

Python

Theory:

Correlation describes the strength of an association between two variables, and is completely symmetrical, the correlation between A and B is the same as the correlation between B and A. However, if the two variables are related it means that when one changes by a certain amount the other changes on an average by a certain amount.
If y represents the dependent variable and x the independent variable, this relationship is described as the regression of y on x. The relationship can be represented by a simple equation called the regression equation. The regression equation representing how much y changes with any given change of x can be used to construct a regression line on a scatter diagram, and in the simplest case this is assumed to be a straight line.

Procedure :

1. Compute $\sum X, \sum Y, \sum X^2, \sum Y^2$ and $\sum XY$.
2. Calculate correlation coefficient by

$$\rho = \frac{N\sum XY - \sum X \sum Y}{\sqrt{N\sum X^2 - (\sum X)^2}\sqrt{N\sum Y^2 - (\sum Y)^2}}$$

3. Compute $\bar{X} = \frac{\sum X}{N}$ and $\bar{Y} = \frac{\sum Y}{N}$
4. Calculate regression coefficient by

$$b_{YX} = \frac{N\sum XY - \sum X \sum Y}{N\sum X^2 - (\sum X)^2}$$

5. The regression line Y on X is given by

$$Y = b_{YX}(X - \bar{X}) + \bar{Y}$$

6. Plot the given data and the Regression line in a graph.

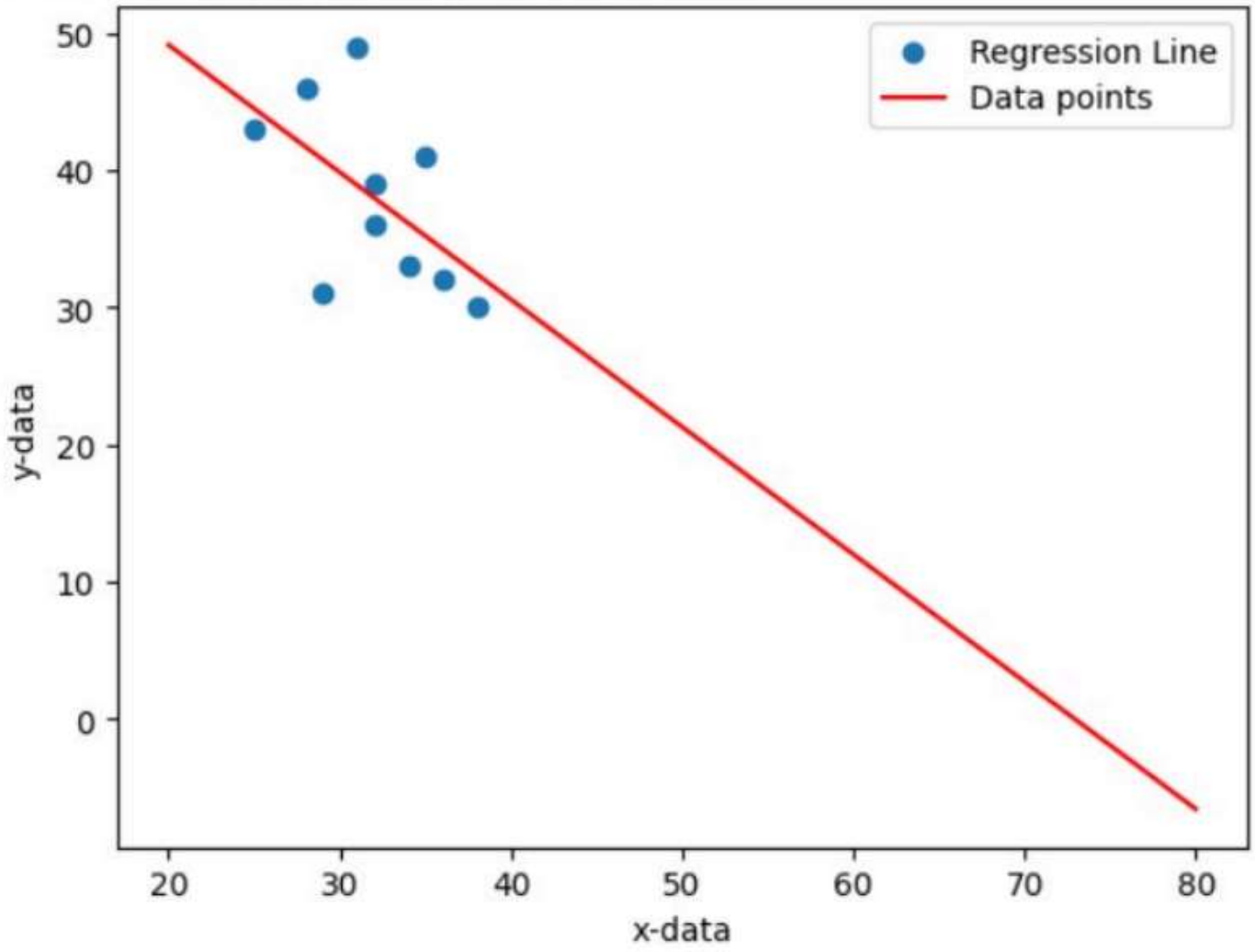
Program :

Developed by:Praisy nishitha Register no:24900090

```
import matplotlib.pyplot as plt
x=[int(i) for i in input().split()]
y=[int(i) for i in input().split()]
N=len(x)
Sx=0
Sy=0
Sxy=0
Sx2=0
Sy2=0
for i in range(0,N):
  Sx=Sx+x[i]
  Sy=Sy+y[i]
  Sxy=Sxy+x[i]*y[i]
  Sx2=Sx2+x[i]**2
  Sy2=Sy2+y[i]**2
r=(N*Sxy-Sx*Sy)/(math.sqrt(N*Sx2-Sx**2)*math.sqrt(N*Sy2-Sy**2))
print("The Correlation coefficient is %0.3f"%r)
byx=(N*Sxy-Sx*Sy)/(N*Sx2-Sx**2)
xmean=Sx/N
ymean=Sy/N
print("THe Regression line Y on X is ::: y = %0.3f + %0.3f (x-%0.3f)"%(ymean,byx,xmean))
plt.scatter(x,y)
def Reg(x):
  return ymean + byx*(x-xmean)
x=np.linspace(0,80,51)
y1=Reg(x)
plt.plot(x,y1,'r')
plt.xlabel('x-data')
plt.ylabel('y-data')
plt.legend(['Regression Line','Data points'])
```

output

```
25 28 35 31 32 36 29 38 34 32
43 46 41 49 36 32 31 30 33 39
The Correlation coefficient is -0.551
The Regression line Y on X is ::: y = 38.000 + -0.929 (x-32.000)
<matplotlib.legend.Legend at 0x78a0ac383a30>
```



Result

The correlation and regression for data analysis of objects from feeder using probability distribution are calculated