**AIM:**Write a program in Python to implement Adaline Neural Network.

**CODE:**

Import numpy as np

x1=np.array([[1,1,-1,-1]])

x2=np.array([[1,-1,1,-1]])

t=np.array([[1],[1],[1],[-1]])

w11=0.1

w21=0.1

w01=0.1

alpha=0.1

i=0

bias=1

w1=np.zeros((4,1))

w2=np.zeros((4,1))

w0=np.zeros((4,1))

Yin=np.zeros((4,1))

y=np.zeros((4,1))

error=np.zeros((4,1))

count=0

while(count!=3):

i=0

if(count!=0):

w11=w1[3]

w21=w2[3]

w01=w0[3]

while(i!=4):

if(i==0):

Yin[i]= (x1[0][i]\*w11)+(x2[0][i]\*w21)+(bias\*w01)

y[i]=t[i][0]-Yin[i]

w1[i]=w11+(alpha\*y[i]\*x1[0][i])

w2[i]=w21+(alpha\*y[i]\*x2[0][i])

w0[i]=w01+(alpha\*y[i]\*bias)

else:

if(i>0 & i<=4):

Yin[i]= (x1[0][i]\*w1[i-1])+(x2[0][i]\*w2[i-1])+(bias\*w0[i-1])

y[i]=t[i][0]-Yin[i]

w1[i]=w1[i-1]+(alpha\*y[i]\*x1[0][i])

w2[i]=w2[i-1]+(alpha\*y[i]\*x2[0][i])

w0[i]=w0[i-1]+(alpha\*y[i]\*bias)

error[i]=(y[i])\*\*2

i=i+1

print('EPOCH',(count+1),':')

print('\n')

print('w1:',w1)

print('\n')

print('w2:',w2)

print('\n')

print('w0:',w0)

print('\n')

print('error',error)

print('\n\n')

count=count+1