- Name: Prerna Sunil Jadhav
- Sap ID: 60004220127
- Batch: C22
- Branch: Computer Engineering
- Course: Machine Learning
- Experiment 5: BackPropogation Algorithm

```
import numpy as np
import matplotlib.pyplot as plt

x1 = float(input("Enter x1: "))
x2 = float(input("Enter x2: "))

bias1 = float(input("Enter bias1: "))
bias2 = float(input("Enter bias2: "))
bias3 = float(input("Enter bias3: "))

learning_rate_alpha = float(input("Enter learning rate: "))

epochs = int(input("Enter number of epochs: "))

w11 = 0.6
w21 = -0.1
w12 = -0.3
w22 = 0.4
w31 = 0.4
w32 = 0.1

errors = []

def sigmoid(x : int):
    return 1 / (1 + np.exp(-x))

for epoch in range(epochs):
    # Forward pass
    z1_input = x1 * w11 + x2 * w21 + bias1
    z1 = sigmoid(z1_input)

    z2_in = x1 * w12 + x2 * w22 + bias2
    z2 = sigmoid(z2_in)

    y_input = z1 * w31 + z2 * w32 + bias3
    y = sigmoid(y_input)

    target_y = 1
    error_y = (target_y - y)
    errors.append(error_y)

    # Backward pass
    delta_y = y * (1 - y) * (target_y - y)
    delta_z1 = z1 * (1 - z1) * w31 * delta_y
    delta_z2 = z2 * (1 - z2) * w32 * delta_y

    w31 += learning_rate_alpha * z1 * delta_y
    w32 += learning_rate_alpha * z2 * delta_y

    w11 += learning_rate_alpha * x1 * delta_z1
    w21 += learning_rate_alpha * x2 * delta_z1
    w12 += learning_rate_alpha * x1 * delta_z2
    w22 += learning_rate_alpha * x2 * delta_z2

    if (epoch == 0 or epoch == epochs-1):
      print(f"\nEpoch Number: {epoch + 1}(Forward pass):")
      print(f"Hidden layer:, z1 = {z1:.4f}, z2 = {z2:.4f}")
      print(f"Error at y: {error_y:.4f}")
      print("Updated weights after Backward propogation of error:")
      print(f"w11 = {w11:.4f}")
      print(f"w12 = {w12:.4f}")
      print(f"w21 = {w21:.4f}")
      print(f"w22 = {w22:.4f}")
      print(f"w31 = {w31:.4f}")
      print(f"w32 = {w32:.4f}")
      print(f"Output: y = {y:.4f}\n")
```

```
Enter x1: 0
Enter x2: 1
Enter bias1: 0.3
Enter bias2: 0.5
Enter bias3: -0.2
```

```
Enter learning rate: 1
Enter number of epochs: 650

Epoch Number: 1(Forward pass):
Hidden layer:, z1 = 0.5498, z2 = 0.7109
Error at y: 0.4773
Updated weights after Backward propogation of error:
w11 = 0.6000
w12 = -0.3000
w21 = -0.0882
w22 = 0.4024
w31 = 0.4655
w32 = 0.1847
Output: y = 0.5227


Epoch Number: 650(Forward pass):
Hidden layer:, z1 = 0.7565, z2 = 0.8296
Error at y: 0.0241
Updated weights after Backward propogation of error:
w11 = 0.6000
w12 = -0.3000
w21 = 0.8337
w22 = 1.0831
w31 = 2.4151
w32 = 2.5012
Output: y = 0.9759
```

```
plt.plot(range(1, epochs + 1), errors)
plt.xlabel('Epoch')
plt.ylabel('Error')
plt.title('Graph of Error vs. Epochs')
plt.show()
```