Experiment 1

Aim :- To train a neural network to implement a digit classifier using the MNIST dataset by only using the numpy library.

Theory:-

1. **Dataset:** The MNIST dataset consists of 60,000 training images and 10,000 test images of handwritten digits (0-9), each of size 28x28 pixels.
2. **Libraries Used:** NumPy
3. **Network Architecture:** A simple feedforward neural network with the following layers:
    a. Input Layer: 784 neurons (flattened 28x28 image pixels)
    b. Hidden Layer: 128 neurons (ReLU activation)
    c. Output Layer: 10 neurons (Softmax activation)
4. **Training Method:**
    a. Data Preprocessing: Normalize pixel values to range [0,1]
    b. Forward Propagation: Compute activations for each layer
    c. Loss Function: Categorical Cross-Entropy
    d. Backpropagation: Compute gradients using the chain rule
    e. Optimization: Stochastic Gradient Descent (SGD) with a learning rate of 0.01

Procedure:

1. **Data Preprocessing:**
    a. Load the MNIST dataset.
    b. Normalize the pixel values by dividing them by 255.
    c. Convert labels to one-hot encoding.
2. **Model Initialization:**
    a. Initialize weights and biases randomly.
3. **Training Loop:**
    a. For each epoch:

i. Perform forward propagation to compute predictions.
ii. Calculate the loss using categorical cross-entropy.
iii. Perform backpropagation to compute gradients.
iv. Update weights and biases using gradient descent.

4. **Testing the Model:**
    a. Perform forward propagation on test images.
    b. Compute accuracy by comparing predictions with actual labels.

Results:

- The model achieved an accuracy of approximately 85-90% on the test set after training for multiple epochs.
- Training time depended on computational resources but typically required several minutes per epoch.
- The performance of the model was observed to improve with increased hidden layer neurons and fine-tuning of learning rates.

Conclusion: The experiment successfully demonstrated that a neural network could be implemented and trained using only NumPy for digit classification on the MNIST dataset. The accuracy obtained is reasonable for a basic feedforward network without advanced optimizations. Further improvements, such as adding more hidden layers, using better weight initialization techniques, and implementing advanced optimizers like Adam, could enhance model performance.