Assignment Report

Azeem Waqar

Fast University, Islamabad

1 Question 1: Implementing Rosenblatt's Perceptron

1.1 Approach

- **Dataset Generation:** Created a synthetic dataset with 500 points, two features, and binary labels.
- **Visualization:** Plotted the dataset to analyze class separation.
- **Perceptron Implementation:**
  - Forward pass computes weighted sum and applies a step function.
  - Backward pass updates weights using the perceptron learning rule.
- **Evaluation:** Plotted decision boundary and calculated test accuracy.

2 Question 2: Implementing Convolution from Scratch

2.1 Approach

- **Dataset:** Used a grayscale image for processing.
- **Manual 2D Convolution:** Implemented convolution using nested loops.
- **Applied Filters:**
  - Edge detection to highlight sharp changes.
  - Blurring for noise reduction.
  - Sharpening to enhance details.
- **Comparison:** Evaluated convolution vs correlation, impact of stride and padding.
- **Visualization:** Displayed original and processed images.

3 Question 3: Implementing a CNN for CIFAR-10

3.1 Approach

- **Dataset:** Loaded CIFAR-10, normalized pixel values.
- **Model:** Convolutional layers (feature detection), pooling layers (downsampling), fully connected layers, softmax output.
- **Training:** Categorical cross-entropy loss, Adam optimizer.
- **Evaluation:** Used accuracy, confusion matrix, loss curves.
- **Enhancement:** Applied data augmentation (flipping, rotation) to improve generalization.

4 Question 4: Implementing a Vanilla RNN for Next-Word Prediction

4.1 Approach

- **Dataset:** Loaded Shakespeare text, tokenized words.
- **Model:** Embedding layer (word representation), SimpleRNN layer, dense softmax output.
- **Training:** Used categorical cross-entropy loss, Adam optimizer.
- **Prediction:** Given a phrase, the model predicts the next 10 words.
- **Evaluation:** Measured perplexity, accuracy, and compared with Word2Vec/GloVe embeddings.

5 Question 5: Hyperparameter Search for CNN and RNN

5.1 Approach

- **Hyperparameters:** Learning rate, layers, neurons, dropout, optimizer.
- **Method:** Used random search to automate testing.
- **Comparison:** Evaluated performance metrics (accuracy, loss).
- **Outcome:** Identified the best hyperparameter combination.