Deep Learning Assignment Report

MANISH GAYEN
21161
March 27, 2024

Question 1

In logistic regression, Cross Entropy loss is preferred over Mean Squared Error (MSE) because:

1. **Probabilistic Interpretation**: Cross Entropy loss aligns with logistic regression's probabilistic interpretation, measuring the difference between predicted probabilities and actual class labels.
2. **Single Best Answer**: Cross Entropy loss encourages the model to output probabilities close to true class labels, ensuring convergence towards a single best answer.
3. **Impact on Training**: Cross Entropy loss facilitates faster convergence by providing stable optimization, especially in cases of imbalanced class distributions, as it adjusts parameters sensitively based on prediction importance.

Question 2

For a binary classification task with linear activation functions, let y_i be the true label and $\hat{y}_i$ be the predicted output of the neural network for input x_i. The Mean Squared Error (MSE) loss for a single sample is given by:

$$L_{\text{MSE}}(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$$

For a dataset with N samples, the total MSE loss is:

$$L_{\text{total}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

Considering linear activation functions, $\hat{y}_i$ can be written as a linear function of x_i and the parameters θ of the neural network:

$$\hat{y}_i = f(x_i, \theta) = \theta^T x_i + b$$

Where θ^T denotes the transpose of the weight vector, x_i denotes the input vector, and b is the bias term.

Substituting $\hat{y}_i$ in the MSE loss equation:

$$L_{\text{MSE}}(y_i, \theta^T x_i + b) = (y_i - (\theta^T x_i + b))^2$$

This is a quadratic function of the parameters θ, and thus, it is convex. Since the total MSE loss is a sum of convex functions (one for each sample), it is also convex.

On the other hand, Cross-Entropy (CE) loss does not guarantee convexity when using linear activation functions. The CE loss function involves a logarithm term and can lead to non-convexity, especially when combined with linear activation functions.

Therefore, the correct answer is (b) MSE.

Question 3

Neural Network Architecture

Number of Hidden Layers: 2
Neurons per Layer: 256
Activation Functions: ReLU for hidden layers, Softmax for output layer

Preprocessing

- Convert images to tensors.
- Normalize pixel values to the range [0, 1].

Hyperparameter Tuning Strategies

- Use grid search or random search to tune hyperparameters such as learning rate, batch size, and number of epochs.
- Consider cross-validation to evaluate model performance on different subsets of the data.
- Monitor metrics such as accuracy, loss and validation performance during hyperparameter tuning.

Question 4

The accuracy curve for each model are as follows:

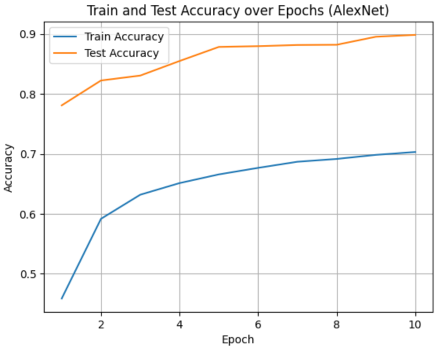

Figure 1: AlexNet

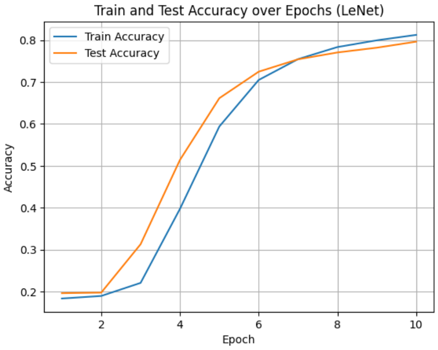

Figure 2: LeNet

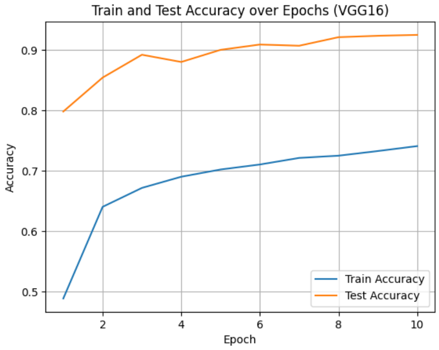

Figure 3: VGG-16

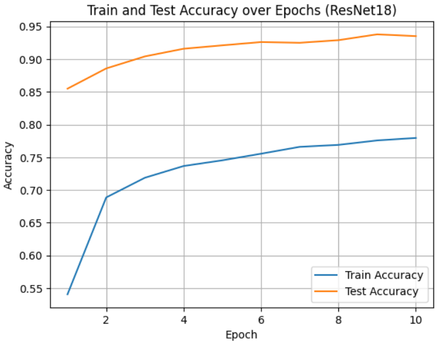

Figure 4: ResNet-18

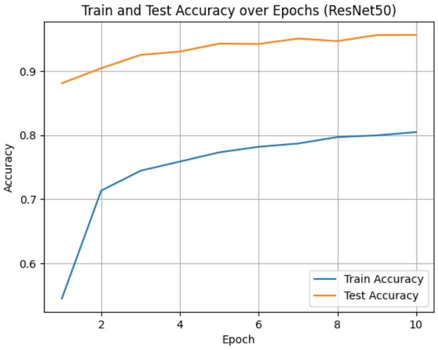

Figure 5: ResNet-50

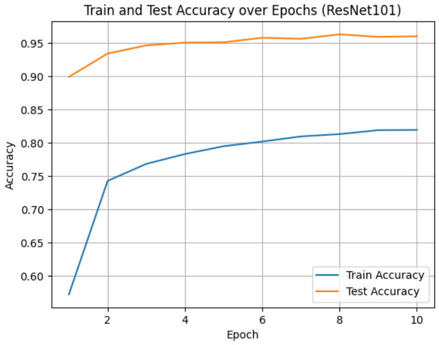

Figure 6: ResNet-101

The accuracy of the various models after training 10 epochs on 25% of the SVHN dataset are given below:

From the accuracy scores, it can be observed that ResNet50 and ResNet101 perform the best out of the rest. This can be explained by the following reasons:

1. **Deeper Architecture**: ResNet50 and ResNet101 are deeper networks compared to ResNet18, AlexNet, LeNet5, VGG-16 allowing them to capture more intricate patterns and features from the SVHN dataset.
2. **Increased Parameters**: With more number of parameters, ResNet50 and ResNet101 have greater flexibility to learn complex patterns, which is important for the varied digit styles and backgrounds present in SVHN images.
3. **Residual Connections**: ResNet50 and ResNet101 utilize residual connections to ease training of deep networks, mitigating issues like vanishing gradients and facilitating better feature reuse throughout the network.

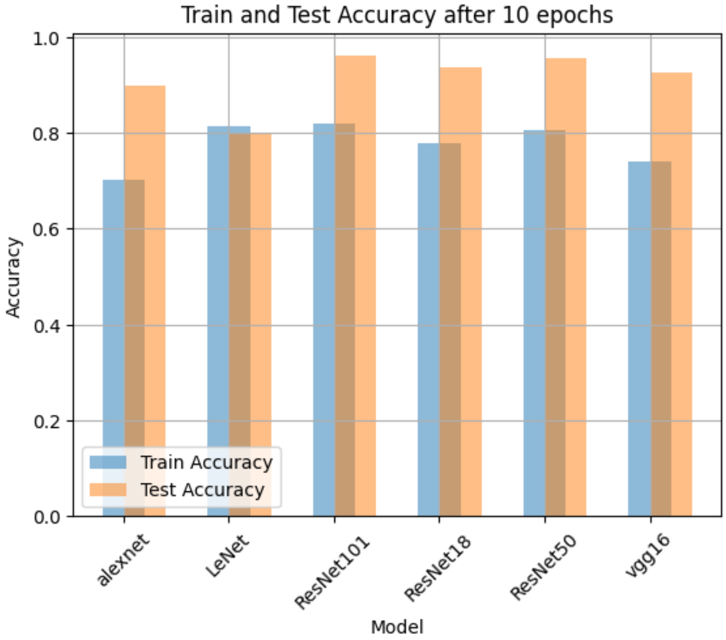

Figure 7: Train and Test accuracy

4. **Feature Reuse**: Deeper layers in ResNet50 and ResNet101 can effectively reuse features learned in earlier layers through skip connections, aiding in preserving and utilizing low-level features essential for recognizing digits in varied conditions.

5. **Model Capacity**: The higher capacity of ResNet50 and ResNet101 enables them to capture more complex relationships and patterns in the data, which is beneficial for datasets like SVHN with diverse digit variations.

6. **Handling Data Variability**: SVHN dataset includes varied street view house numbers captured under different conditions. Deeper networks are better equipped to handle this variability due to their increased capacity and ability to learn diverse features from the data.