Project Report: Image Classification using a Neural Network

Name: Atreyee Das

Enrollment no.: 12023052002045

Department: Computer Science Engineering (CSE)

This report outlines the development and evaluation of a neural network model for image classification, focusing on a dataset of images. Here’s the link to the github repository for my code and datasets:

# Dataset

There are two csv files with the train and test data information. Each csv file has 785 Columns. Column 1 contains the Class Label and Column 2-784 contains Pixel Values from 28x28 Image arranged in row major format.

# Model Architecture

The model employed in this project is a convolutional neural network (CNN) consisting of the following layers:

1. **Input Layer:** This layer accepts grayscale images of size 28x28 pixels.
2. **Convolutional Layers:** Two convolutional layers are used with 32 and 64 filters, respectively, using a 3x3 kernel and ReLU activation. L2 regularization is applied to the convolutional layers to prevent overfitting.
3. **Max Pooling Layers:**Two max pooling layers with 2x2 pooling size to downsample the feature maps.
4. **Flatten Layer:** Flattens the output of the convolutional layers to a 1D vector.
5. **Dense Layers:** We have one dense layer with 128 neurons and ReLU activation. L2 regularization is applied to this dense layer.
6. **Dropout Layer:** A dropout layer with a 30% dropout rate is introduced to further prevent overfitting.
7. **Output Layer:** A dense layer with 10 neurons and softmax activation to output the probability of each class.

# Training and Validation

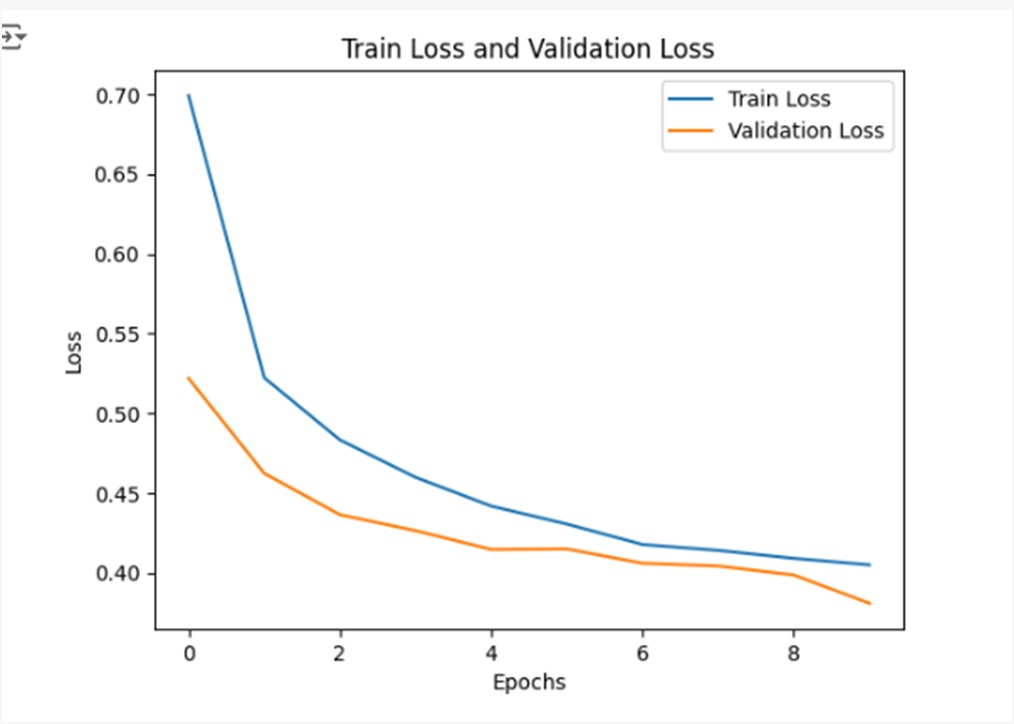
The model was trained using the Adam optimizer and the sparse categorical cross-entropy loss function. The training was performed for 10 epochs with a batch size of 32. A 20% split of the training data was used for validation.

#### Training and Validation Loss Graphs:

**Training Loss:** The training loss decreases consistently across the epochs, indicating the model is learning effectively from the training data.

**Validation Loss:** The validation loss also decreases but with slight fluctuations. The gap between training and validation loss remains relatively small, suggesting that the model is not overfitting significantly.

**Observations:** The decrease in both training and validation loss indicates that the model is learning the underlying patterns in the data and generalizing well to unseen examples.



# Performance Metrics

The model's performance of the test dataset was evaluated using the following metrics:

* **Test Accuracy:** 0.9000
* **Test Precision:** 0.9006
* **Test Recall:** 0.9000
* **Test F-Measure:** 0.8985

These metrics indicate that the model achieves a high level of accuracy and is effective in classifying the images. The precision and recall values are also close to 1, suggesting that the model has a balanced performance across different classes.

#### Confusion Matrix:

The confusion matrix provides a visual representation of the model's performance across the different classes. Each cell in the matrix indicates the number of predictions made for each class, allowing for an easy identification of which classes are being confused with one another. The diagonal values represent correct predictions, while off-diagonal values indicate misclassifications.

# Conclusion

In conclusion, the neural network model developed for image classification demonstrates strong performance on the provided dataset. The architecture effectively captures the features of the images, and the training process has resulted in a model that generalizes well to unseen data. Future work could involve experimenting with deeper architectures, data augmentation techniques, or transfer learning to further improve performance.