# Deep Learning Report

Abhinav

Roll No. - 21004
DL Assignment 1

## Answer 1:

Cross Entropy (CE) and Mean Squared Error (MSE) are two common loss functions used in machine learning tasks. CE is primarily utilized in classification problems, where the goal is to predict the probability distribution of classes. It measures the dissimilarity between the predicted probabilities and the actual class labels, optimizing the model to make more confident and accurate predictions for each class. On the other hand, MSE is typically employed in regression tasks, calculating the average squared difference between predicted and actual values. While both loss functions have their merits, the choice between them depends on the specific problem at hand.

In logistic regression, where the objective is to predict probabilities for binary classification, CE is preferred over MSE. This preference stems from CE's direct alignment with the model's goal of estimating probabilities. By optimizing the CE loss, logistic regression models are trained to output probabilities that closely match the true class labels, resulting in better-calibrated predictions. In contrast, MSE may not be as effective for logistic regression since it does not explicitly optimize for probability estimation, potentially leading to suboptimal performance in classification tasks. Therefore, for logistic regression models aiming to achieve accurate and well-calibrated probability predictions, CE emerges as the preferred choice over MSE.

## Answer 2:

Option C - both
In a binary classification task with a deep neural network (DNN) containing at least one hidden layer and equipped with linear activation functions, the Mean Squared Error (MSE) loss function guarantees a convex optimization problem , because :

1. **Linear Activation Functions:** Linear activation functions, such as the identity function $f(x) = x$, result in a linear relationship between the input and output of each neuron in the network. Since linear functions are convex, composing multiple linear activation functions in a deep neural network preserves convexity throughout the network.

2. **Mean Squared Error (MSE) Loss:** MSE computes the average of the squared differences between predicted and actual values. The MSE loss function itself is convex, as the squared error term is a convex function, and averaging preserves convexity.

3. **Convex Optimization:** Combining linear activation functions with the MSE loss function results in a convex optimization problem. Convex optimization problems have a unique global minimum, and gradient-based optimization algorithms (such as gradient descent) can efficiently find this minimum.

In summary, when using linear activation functions in a DNN for binary classification tasks, pairing them with the MSE loss function ensures convexity in the optimization problem, guaranteeing a unique global minimum that can be efficiently found through gradient-based optimization methods.

## Answer 3:

The code implements a neural network classifier for the MNIST dataset using TensorFlow and Keras. It preprocesses the data by flattening and normalizing the images. The model architecture consists of two hidden layers with ReLU activation and an output layer with softmax. Hyperparameter tuning is done manually, varying the number of units and activation functions. The best model is selected based on validation accuracy. Finally, the model is evaluated on the test set for generalization performance.

## Answer 4:

The suitability of a model for a particular dataset depends on various factors such as model complexity, dataset characteristics, and computational resources.

1. LeNet-5 is a simple and efficient convolutional neural network architecture. It can be used for the SVHN dataset due to its simplicity and efficiency. However, it may not achieve great performance compared to more complex models like VGG or ResNet. LeNet-5 typically achieves a test accuracy of around 66

2. AlexNet has a deeper architecture compared to LeNet-5, requiring more computational resources. It achieves better test and train scores than LeNet-5 due to its deeper and more complex architecture. AlexNet is suitable for the SVHN dataset, but its deeper architecture may require more computational resources.

3. VGG is well-suited for the SVHN dataset due to its strong performance on image classification tasks. It has the ability to capture intricate features from the data. However, its deeper architecture may require more computational resources compared to shallower models like LeNet-5.

4. ResNet is highly suitable for the SVHN dataset due to its state-of-the-art performance on image classification tasks. It is effective at capturing intricate features in the data. However, its deeper architecture may require significant computational resources for training.

In summary, while LeNet-5 is simpler and more efficient, it may not achieve the same level of performance as more complex models like AlexNet, VGG, or ResNet. VGG and ResNet, with their deeper architectures, have the potential to achieve higher accuracy but may require more computational resources for training. The choice of model ultimately depends on the trade-off between model performance and computational constraints.