

**Task:**

Building a Neural Network from Scratch

**Objective:**

In this assignment, you are required to implement a neural network from scratch in Python. Build a feedforward neural network and implement backpropagation for training. By the end of this assignment, you should have a working neural network that can be trained on a simple dataset for multi-class classification.

**Dataset:**

Use the MNIST dataset for the 10 class classification task. You can use any library to import the dataset.

**Network Architecture:**

1. You have to make a NN using at least 2 fully connected layers (No need for convolutional layers). Other than that you are free to choose the structure of the network.
2. Initialize the weights randomly using seed value as the last three digits of your roll number. For example, your roll number is M24CSE099, then your seed value should be 99. Set bias = 1.
3. Use Train-test splits as randomized 70:30, 80:20 and 90:10.
4. Set batch size as your year of admission. For example, your roll number is M24CSE099, then your batch size should be 24.
5. Use an appropriate loss function.
6. Train for 25 epochs. Plot accuracy and loss per epoch.
7. Prepare a Confusion matrix for all the combinations of the network. You may use an in-built function for this purpose.
8. Report total trainable and non-trainable parameters.

**Bonus: Use lates initialization strategies. Explore different activation functions. Use regularization/early stopping to avoid overfitting, if it occurs.**

**Note: Do not use in-built functions, unless mentioned.**

**Grading Rubrics:**

- Correctness and completeness of the neural network implementation (20 points)
- Implementation of feedforward and backpropagation (15 points)
- Proper handling of training and evaluation (10 points)
- Documentation and clarity of code (5 points)
- Bonus (15 points)