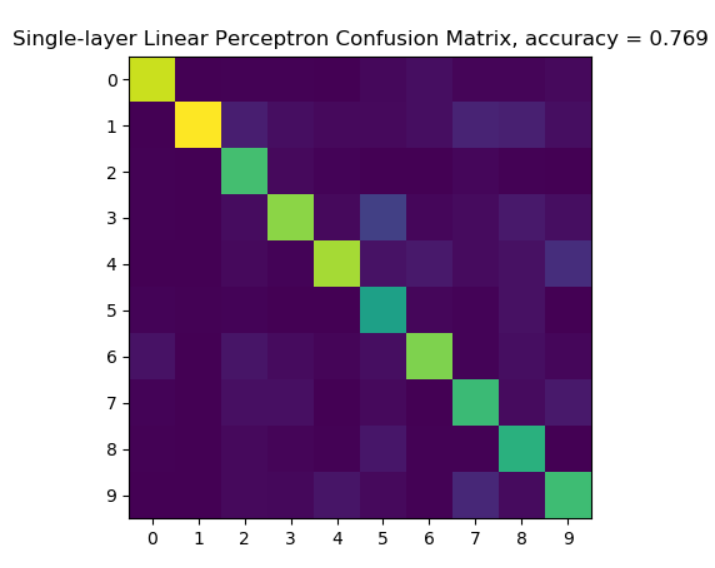
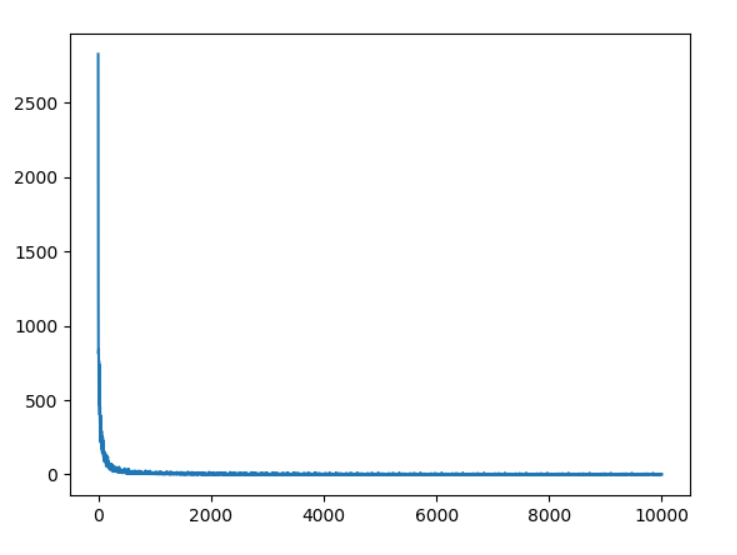
**Assignment 4 – Convolutional Neural Network**

We implemented 4 neural networks to classify images of handwritten digits into 10 classes, 0 to 9. The dataset used was MNIST dataset. The models were trained on 12000 images.

**Part 1: SLP Linear**

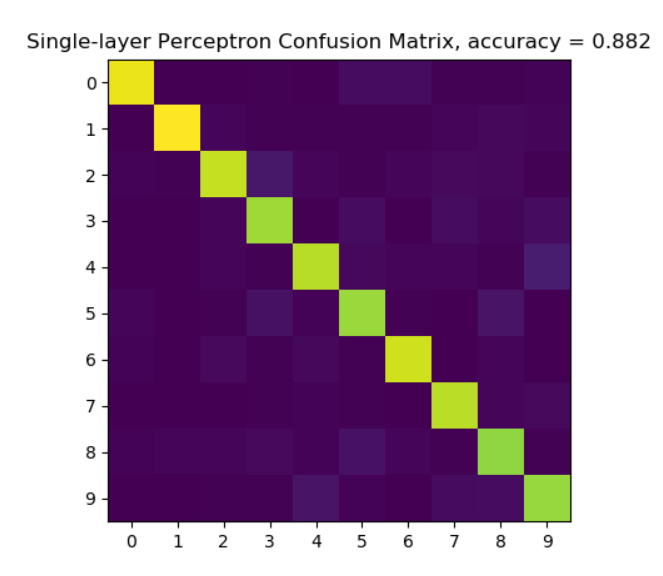
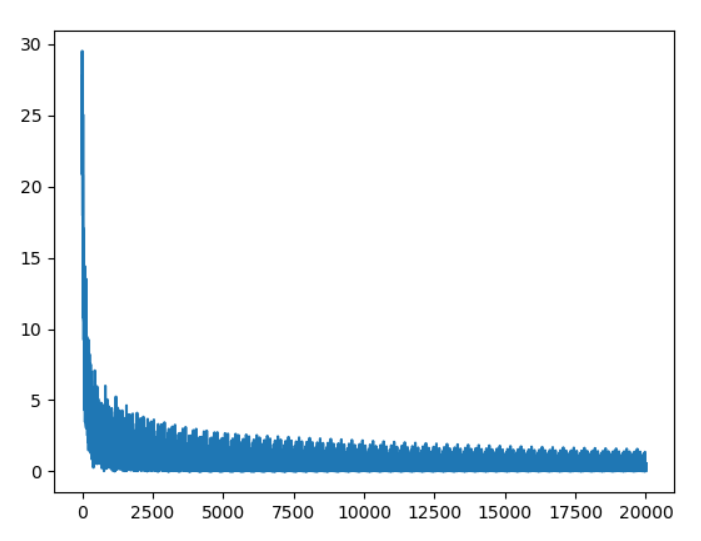
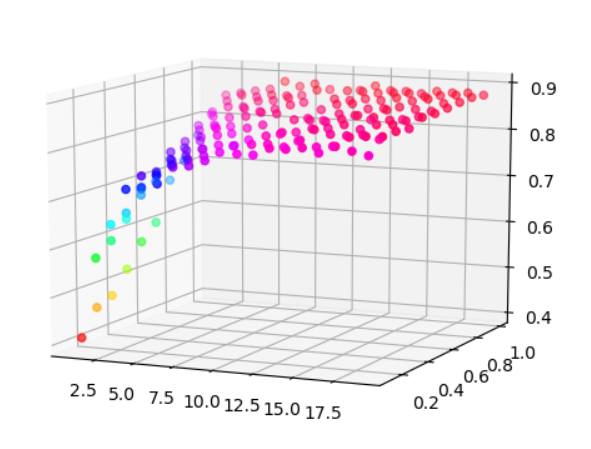
In this part of the assignment, we have implemented a single layer perceptron to classify MNIST dataset. Didn’t use any nonlinear activation in this model. Got **accuracy** of **76.9%** with parameters **learning rate = 0.8** and **decay rate = 0.9**, ran for **10000 iterations.**



Loss over iterations

**Part 2: SLP**

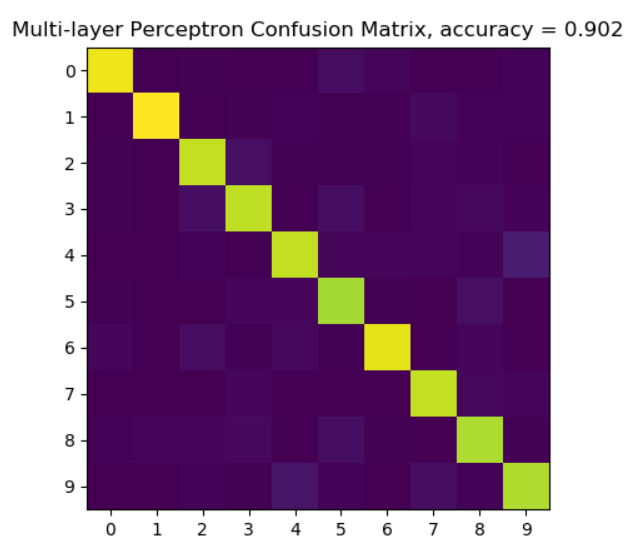
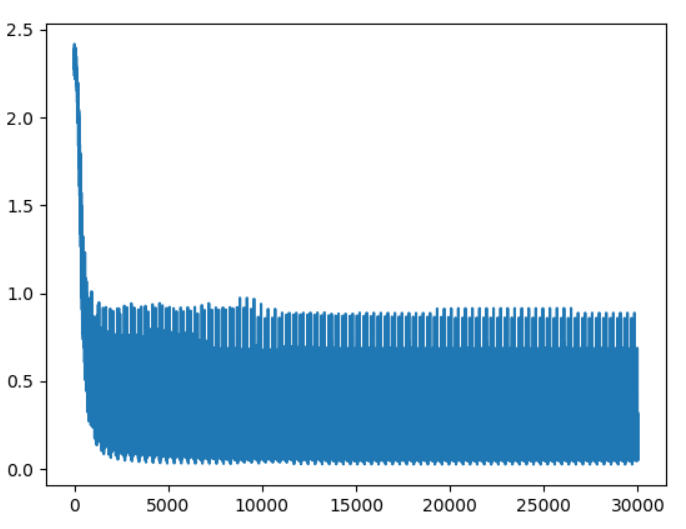
In this part, we modified SLP Linear by changing cost function from euclidean to soft-max cross-entropy. **Accuracy of 88.2%** is achieved with parameters **learning rate = 12** and **decay = 1**, ran for **20000 iterations.** I tried a wide range of learning rates from 0 to 20 and got best accuracy for the above hyper parameters. The accuracy graph depicts accuracy on vertical axis and learning rate and decay on horizontal axes.



Accuracy over hyper params Loss over iterations

**Part 3: MLP**

In this, we implemented a multi-layer perceptron with two fully connected layers, and out of which one was passed through Relu activation layer. We used soft-max cross-entropy as the loss function. **Accuracy of 90.2%** was produced with parameters **learning rate = 0.5** and **decay rate = 0.4,** ran for **30000 iterations.**



**Part 4: CNN**

In this, we implemented a convolutional neural network with one set of convolution, relu and 2x2 max pooling layer. These layers were followed by a flattening layer which was then connected to a fully connected soft max loss layer to generate classes. **Accuracy of 87.6%** is achieved with parameters **learning rate = 0.3** and **decay rate = 0.9**, ran for **iterations 25000**.

