Programming Assignment 2

**Software**

The models were trained and tested using Google Colab: <https://colab.research.google.com/drive/1T4wIEyLqyz9qE8lVItroBPBU_QHs1YN0?usp=sharing\>

The output file *output.txt* was generated using Putty. All files are located in the PA2 zip file

**Model 1**

The MNIST dataset alters the resolution of each image to be 28x28x1, meaning each image is a grayscale image of size 28x28. For this implementation, the models accept batch sizes of 10 images. Model 1 accepts one 28x28 image and flattens the pixel features into a dimensional space that can be accepted by the fully connected layer (**fcl1**). Then, the flattened features are fed into fcl1 which is comprised of 100 neurons, each using the sigmoid function for activation. It was trained with a learning rate of .1 for 60 epochs.

Average accuracy: 85.95%

Average loss: 3.6577

**Model 2**

Model 2 adds two convolutional layers (**conv1** and **conv2**) to the architecture of model 1. **Conv1** accepts a gray scale image and convolves it using 40 5x5 kernels. **Conv2** accepts the 40 channel output from **conv1** and performs convolution with 40 5x5 kernels. Both convolutional layers utilize max pooling with a 2x2 kernel and sigmoid activation function. The output is then fed through a fully connected layer (**fcl2**).

Average accuracy: 11.35%

Average loss: 3.7637

**Model 3**

Model 3 changes all activation functions from sigmoid to RelU activation. The learning rate for gradient descent was also changed to .03.

Average accuracy: 99.30%

Average loss: 0.0394

**Model 4**

All layers and hyperparameters remained the same as model 3. An additional fully connected layer (**fcl3**) was added after **fcl2** from the previous model.

Average accuracy: 99.37%

Average loss: 0.0396

**Model 5**

New fully connected layers (**fcl4** and **fcl5**) were implemented with 1000 neurons each, as well as RelU as for activation. This model was then trained for 40 epochs using a dropout rate of .5.

Average accuracy: 99.61%

Average loss: 0.0227

**Results**

Model 5 had the best performance on both average accuracy as well as average loss across training samples. Conversely, model 2 had the worst performance of all the models. The low accuracy is most likely attributed to using sigmoid as the activation function instead of RelU. A large learning rate might have also contributed to the model’s poor performance by preventing the learner from converging to the global minimum.