**Average Testing Accuracy**

96.42%, 96.27%, 95.97%, 96.33%, 96.12%. For the feedforward neural network which is above the threshold

96.54%, 96.57%, 95.07%, 96.81%, 96.27% which is again above the threshold.

**What I learned from the project**

I have tried to implement both models myself which was both tiresome and rewarding experience. Although in the end for some reason I could not make the CNN implementation work because it was way too slow and did not learn appropriately, however while implementing the forward pass and back propagation I learnt how important the chain rule was regarding the time complexity of the models. Also, I comprehend why we need derivable functions as activation functions, basically I believe I have a great grasp as how back propagation works. At first, I tried to implement CNN with 2 conv layer and 1 pooling layer which did not seem to work because I could not reduce the pixel amount reasonably and the model did not complete for an hour. Thus, playing with the hyperparameters like neuron amount, pooling size, kernel size and stride size. I learnt how each of these affect the network. I am aware that this was not asked of me, but to completely understand how neural network works, I tried to implement the CNN from scratch and tried to back propagate back from both the NN and convolutional & pooling layers that also gave me intuition as to how I should setup the CNN based on what I have as data. I will have the python code of my implementation of CNN though my final answer would be the one with built in functions.