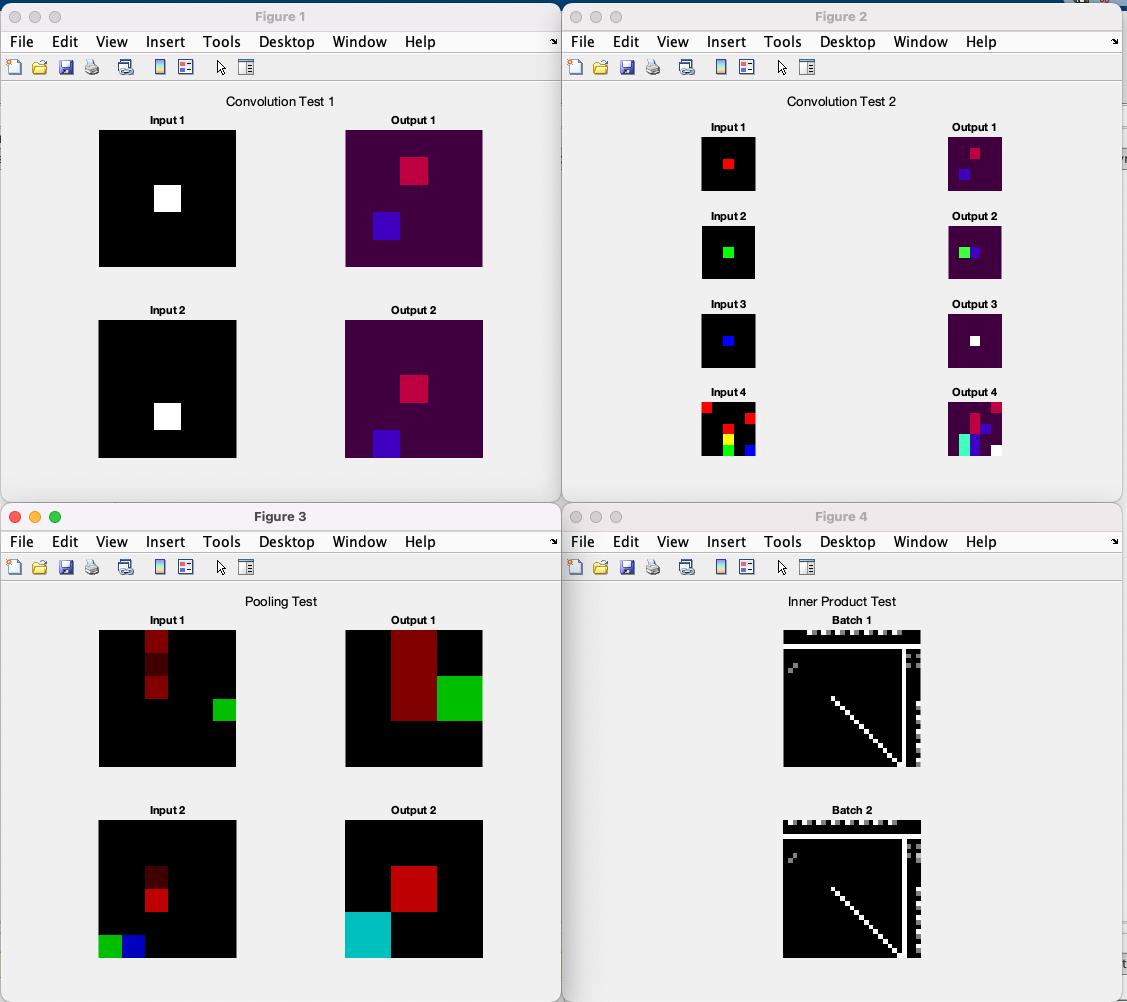
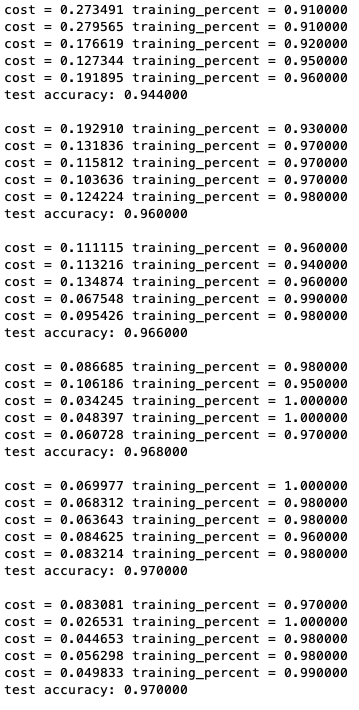
Part 1:

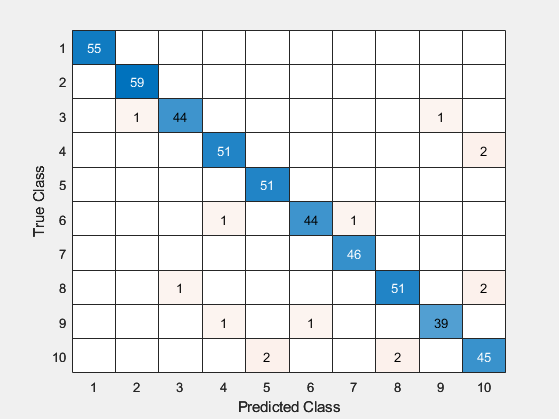


Part 3.1:

After the 3000 iterations, the test accuracy was 97% as can be seen below:

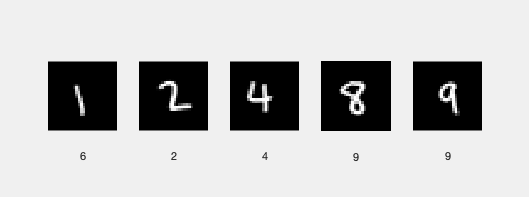


Part 3.2:

The confusion matrix is: 

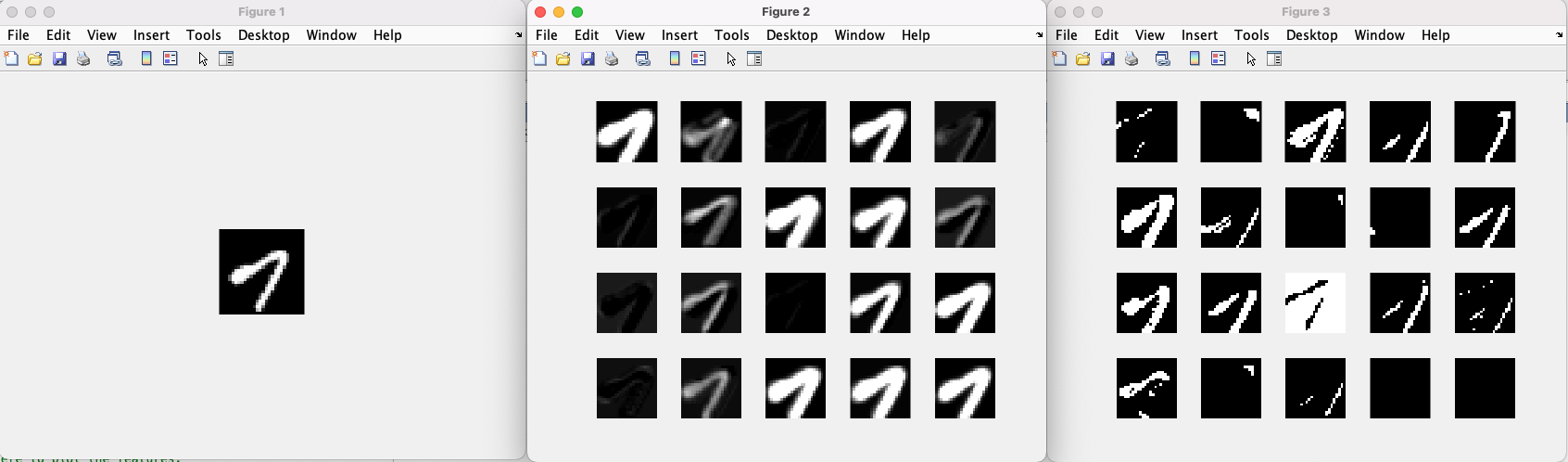
From this it can be seen that the numbers 9 and 7 were the most confused pairs. There were 2 instances where the actual number was 7 and the network predicted a 9 and there were 2 cases where the actual number was 9 and the predicted number was 7. The reason for this is quite clear as the dashed 7 () looks a lot like a , which can confuse the network and make it think that a 9 is a 7 and vice-versa.

Part 3.3:



Based on the samples I wrote and entered as input, the success rate was about 60%.

Part 4.1 and 4.2:

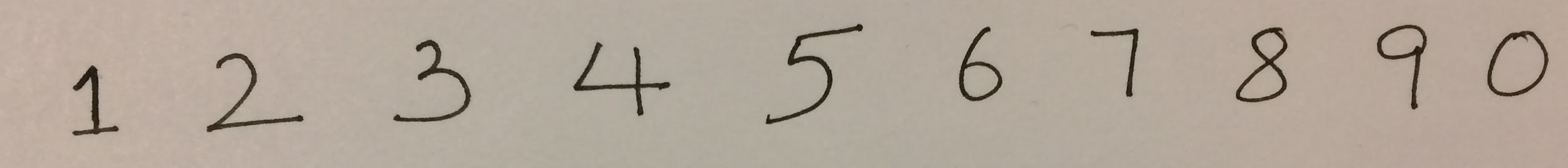


Input Conv\_layer RELU layer

The first convolution layer feature maps seems to dilate the image and make the numbers thicker. It looks like it is doing a sort of max pooling as it picks the max image in a neighbourhood. Since these are greyscale images, it looks like its is trying to further maximize the values and almost binarize the image. The Relu activation maps are taking the input from the convolution layer and making all the spots close to zero darker especially those from the convolution layer that were negative and its leaving the larger values as they are. In some cases the images values are being flipped over completely (higher values are lowered and lower values are made larger). Since pixels with negative values are automatically made zero, these layers look similar, except the Relu layer seems to take on extreme values compared to the original image and convolution layer.

Part 5

Image1 output:



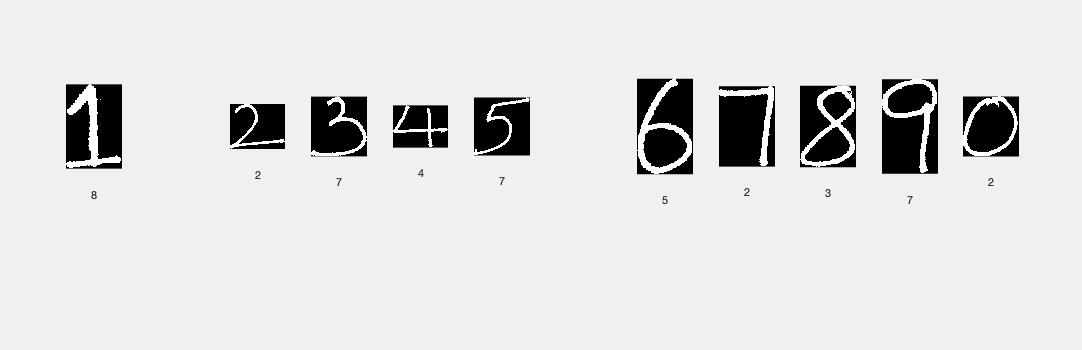
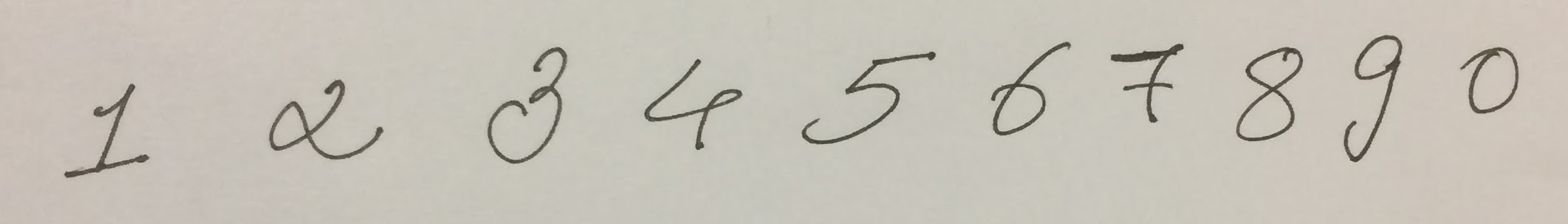


Image2 output:



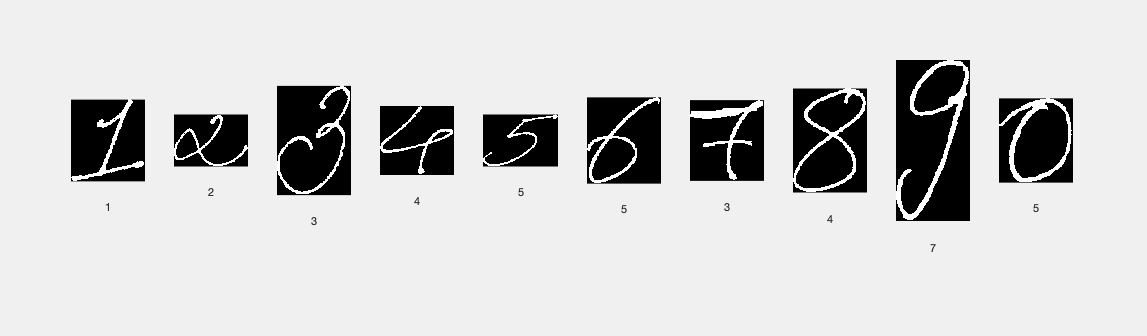
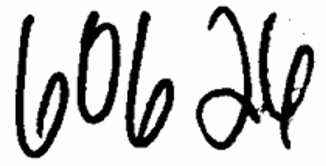


Image3 output:



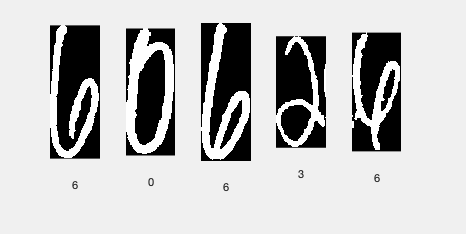


Image4 output:

