# Part A: MNIST with a side of MLP

1. The test set accuracy is 92.8%, after removing all hidden layer.
2. The confusion matrix after normalization, display with float precision of 0.02:

```
    0     1     2     3     4     5     6     7     8     9 Predicted
0 [ 0.98  0.    0.    0.    0.    0.01  0.01  0.    0.    0.  ]
1 [ 0.    0.98  0.01  0.    0.    0.    0.    0.    0.01  0.  ]
2 [ 0.    0.01  0.92  0.01  0.01  0.    0.01  0.01  0.03  0.  ]
3 [ 0.    0.    0.02  0.9   0.    0.03  0.    0.01  0.02  0.01]
4 [ 0.    0.    0.01  0.    0.94  0.    0.01  0.    0.01  0.03]
5 [ 0.01  0.    0.    0.03  0.01  0.88  0.01  0.01  0.03  0.  ]
6 [ 0.01  0.    0.01  0.    0.01  0.02  0.95  0.    0.    0.  ]
7 [ 0.    0.01  0.03  0.    0.01  0.    0.    0.92  0.    0.03]
8 [ 0.01  0.01  0.01  0.02  0.01  0.03  0.01  0.01  0.9   0.01]
9 [ 0.01  0.01  0.    0.01  0.03  0.01  0.    0.02  0.01  0.91]
Actual
```

3. Along the diagonal, the digit 0 and 1 has the highest entry, the digit 5 has the lowest entry
4. The worst performing digit 5 is most often confused with digit 3 and 8
5. The reason is that digit 5 looks very similar to 3 and 8, so it is hard for the model to classify and very likely to perform bad. In contrast, the digit 6 is usually easy to distinguish with other digits.
6. 93.82%
7. 98.16%
8. The new confusion matrix:

```
    0     1     2     3     4     5     6     7     8     9 Predicted
0 [ 0.99  0.    0.    0.    0.    0.    0.    0.    0.    0.  ]
1 [ 0.    0.99  0.    0.    0.    0.    0.    0.    0.    0.  ]
2 [ 0.    0.    0.99  0.    0.    0.    0.    0.01  0.    0.  ]
3 [ 0.    0.    0.    0.99  0.    0.    0.    0.    0.    0.  ]
4 [ 0.    0.    0.    0.    0.98  0.    0.    0.    0.    0.01]
5 [ 0.    0.    0.    0.01  0.    0.98  0.    0.    0.    0.  ]
6 [ 0.    0.    0.    0.    0.    0.    0.98  0.    0.    0.  ]
7 [ 0.    0.    0.01  0.    0.    0.    0.    0.99  0.    0.  ]
8 [ 0.    0.    0.01  0.    0.    0.    0.    0.    0.98  0.  ]
```

```
9 [ 0.    0.    0.    0.    0.    0.    0.    0.01  0.    0.98]
Actual
```

9. The third network with two hidden layer of 256 neuron each has the best performance
10. The test loss decreased from 0.0811322005614 to 0.00274153511821, while the accuracy also decreased from 0.9853 to 0.9834. Therefore, the L2 loss function did not improve the result.

# Part B: MNIST garnished with a CNN

11. Comparing the three architectures with 12 epoches:

| | test accuracy | training time |
|---|---|---|
| A | 97.45% | 38s |
| B | 98.22% | 228s |
| C | 97.86% | 227s |

# Part C: Finely-tuned Cats and Dog

1. After lowering the number of training and validation samples by a factor of 20, the accuracy decrease to 43.75%. The reason is that insufficient data is provided to the model to give good training result.
2. If I fine-tune for 1 or 2 epochs using the original number of training and validation samples, the accuracy gets to 88% in first epoch, 91% in second epoch. The reason is that the significantly increased amount of data improve the neural network, and this certainly help improve the accuracy of testing utility.