COMPARISON OF THE PERFORMANCE OF TWO MULTI-LAYER PERCEPTRONS IN MNIST DATASET

Submitted by,

Alby Joseph
S5 CSE
Roll no: 9
TVE19CS009

SOURCE CODE

The Source code used to train the model can be found here (Github repository link)

I trained the MNIST dataset in two models. The model summary of the two models is as shown in the images below

Model 1

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 784)               0

 dense (Dense)               (None, 70)                54950

 dense_1 (Dense)             (None, 10)                710

=================================================================
Total params: 55,660
Trainable params: 55,660
Non-trainable params: 0
```

Model 2

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten_1 (Flatten)         (None, 784)               0

 dense_2 (Dense)             (None, 80)                62800

 dense_3 (Dense)             (None, 80)                6480

 dense_4 (Dense)             (None, 10)                810

=================================================================
Total params: 70,090
Trainable params: 70,090
Non-trainable params: 0
```

A comparison on both the model training is tabulated below:

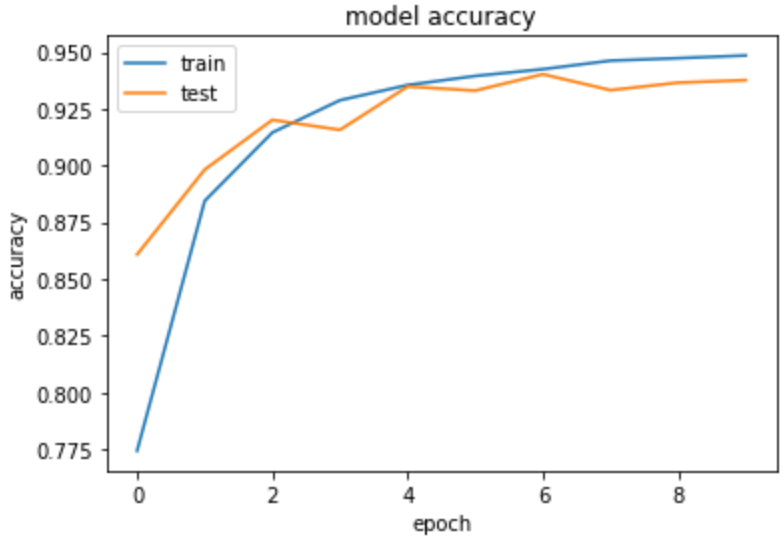
Criteria	Model 1	Model 2
# of Hidden layers	1	2
# of parameters	55,660	70,090
Total # of neurons	80	170

Both the models were trained with same Loss function (cross_entropy), same optimization function(Adam) and the same number of epochs(8).

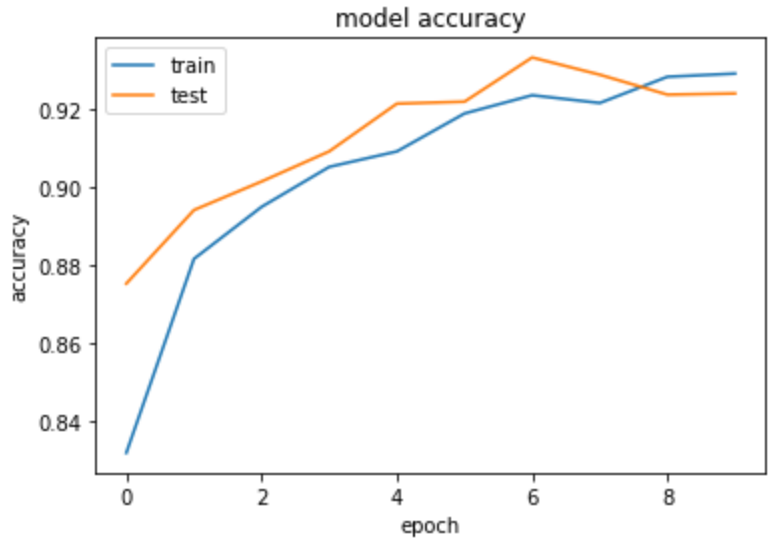
Validation Accuracy of Model 1 =93.76%
Validation Accuracy of Model 2 =92.39%

Accuracy vs Epoch Plots

<u>Model 1</u>



Model 2



Inference

Clearly the second model is a more complex model compared to the first one as it contains more hidden layers, more neurons, and hence more trainable parameters. From the plots, we can infer that the second model is getting overfitted towards the end epochs. Thus Model 1 shows slightly less accuracy than Model 2. The second model is getting overfitted because it is even learning the noise in the training samples and is losing its generalization ability.