

# The changes you made

## Hyperparameters

```
layers of the network : 3
number of neurons in each layer : (784, 32, 10)
epochs = 10
mini_batch_size = 10
eta(learning rate) = 2
```

## Result

```
Initial performance : 861 / 10000
Epoch 0 : 6923 / 10000
Epoch 1 : 7901 / 10000
Epoch 2 : 8020 / 10000
Epoch 3 : 8873 / 10000
Epoch 4 : 8949 / 10000
Epoch 5 : 8980 / 10000
Epoch 6 : 8953 / 10000
Epoch 7 : 8977 / 10000
Epoch 8 : 9005 / 10000
Epoch 9 : 9001 / 10000

the time required to train the net : 95.85s
```

## how/why they improve the net's performance

Original network is a two layers network, I add a layer that makes the network more complex,  
so that it can fit more complex functions.  
Then I change learning rate, to make network convergence quickly.

## Even though the datasets are similar, it is much harder to get a high accuracy for notMNIST, Why?

Since two different datasets have different styles, it is difficult for a network that performs well on this datasets to have the same performance on another datasets.

In MNIST, the image features are relatively simple, while the image features of notMNIST are more complex, so it is necessary to adjust the hyperparameters of the neural network for these two different styles.