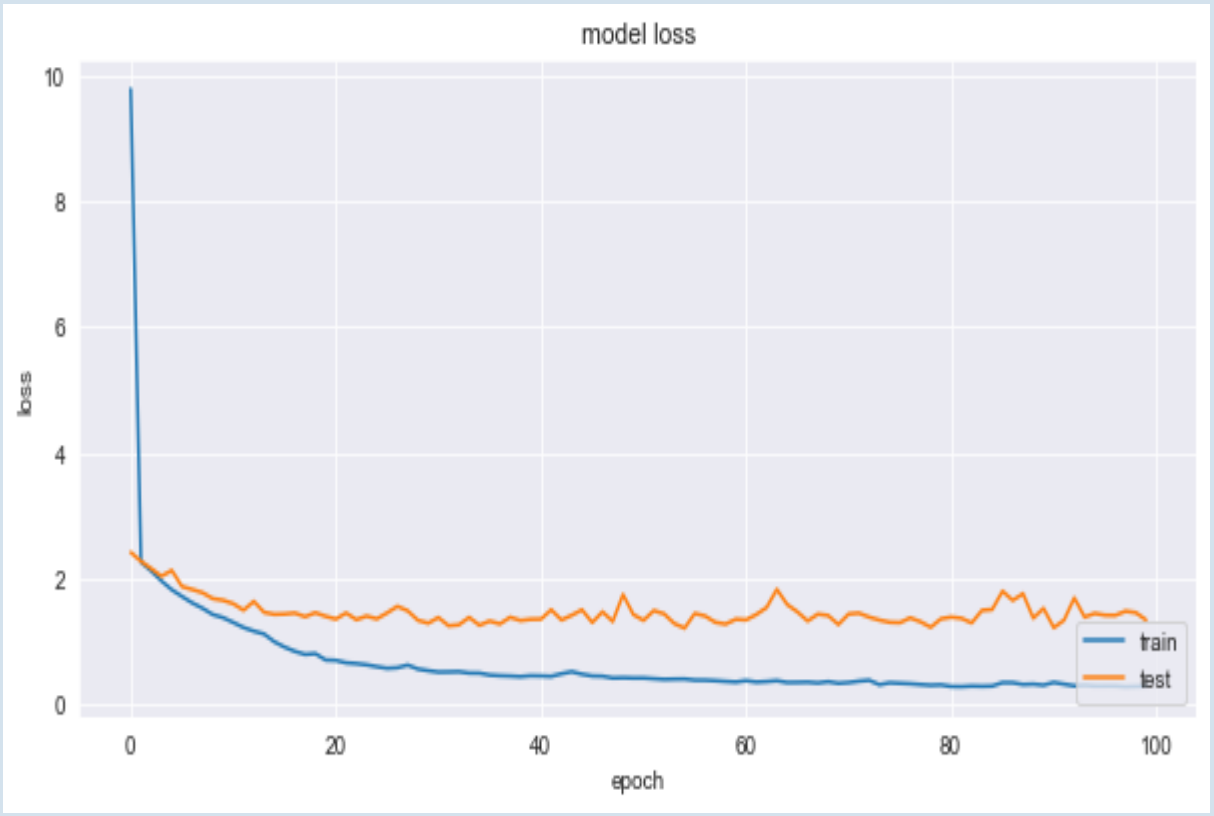
ASSIGNMENT 3
ELL 784
Intro to ML

ROHAN KUMAR BOHARA

2021AMA2095

Part 1 (a): Train neural network on given images

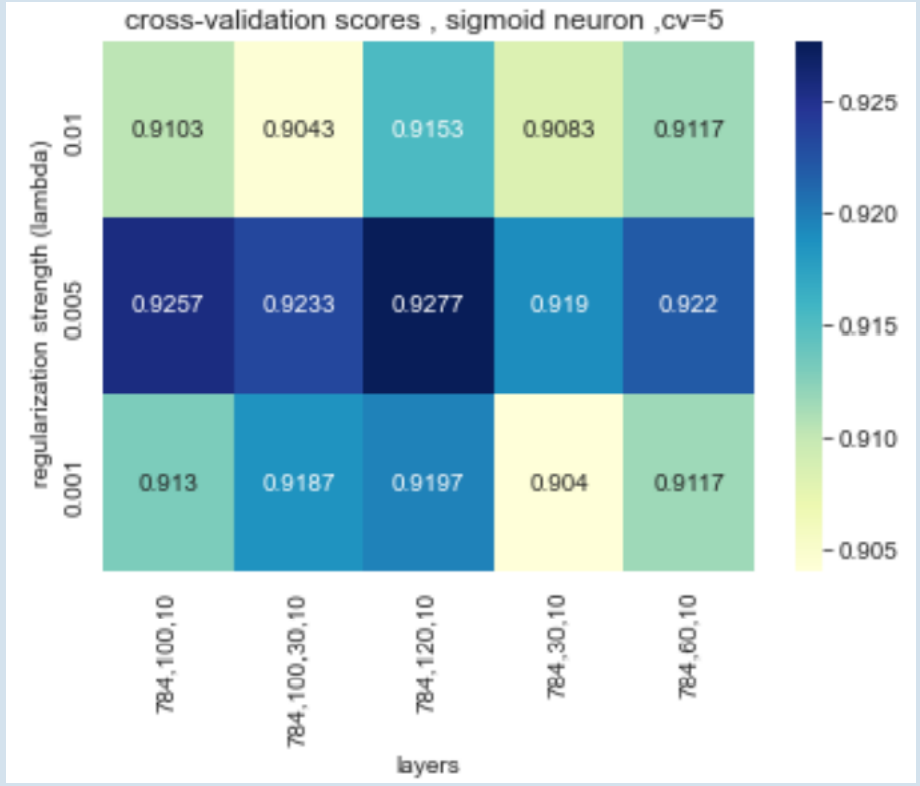
Using Tensorflow library



Using grid search cv from sk learn

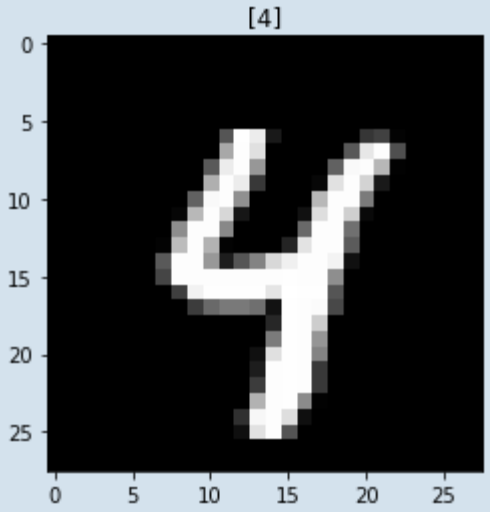
```
clf = GridSearchCV(MLPClassifier(),param_grid=param_grid,cv=StratifiedKFold(n_splits=4))
```

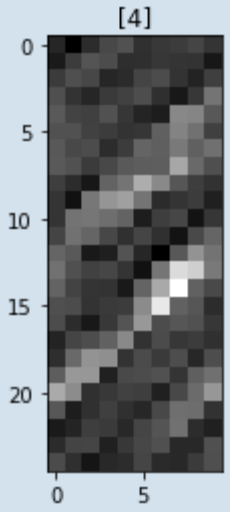
```
train accuracy 0.9955539088551315
test accuracy 0.8866666666666667
MLPClassifier(alpha=50, hidden_layer_sizes=(35, 30), max_iter=100)
```

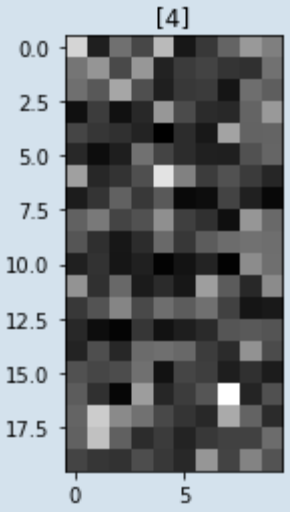


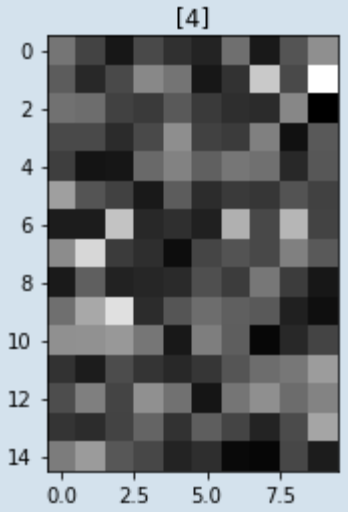

```
Best fit :
0.001 learning_rate_init
fiiting time 792.9344367980957

train accuracy 0.9855502037791775
test accuracy 0.9166666666666666
```

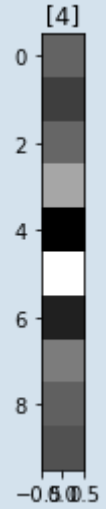
Correct classification Visualisation



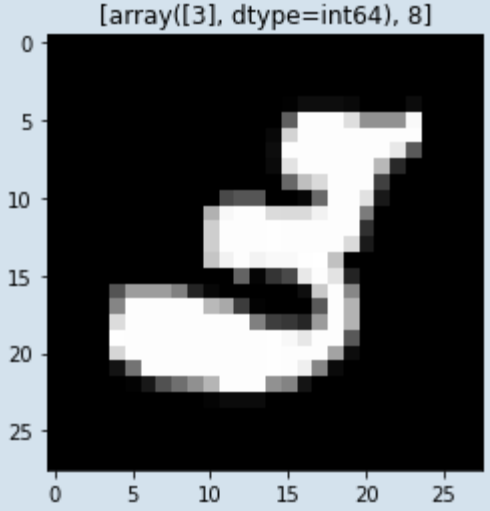


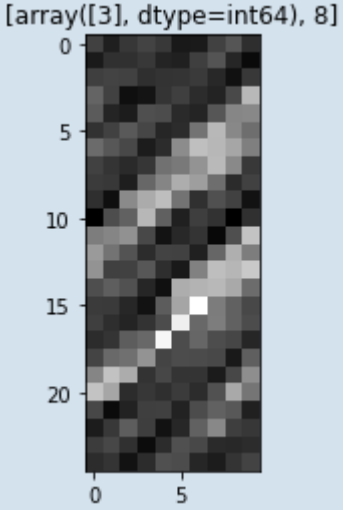


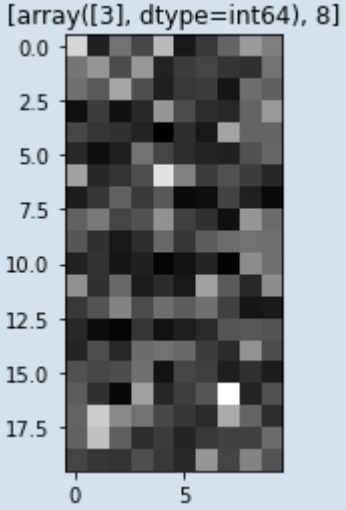


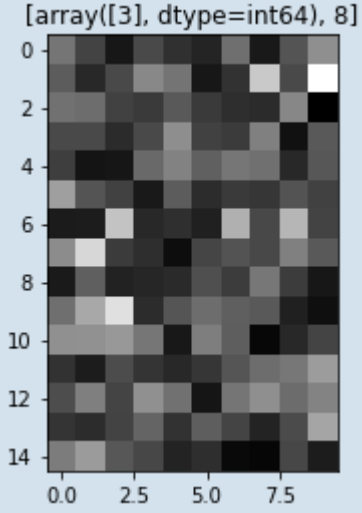


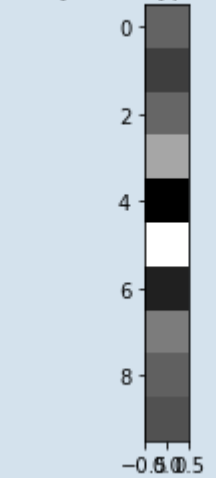
wrong classification visualization











Part 1(b): Comparison with PCA features

Dimensionality reduction in neural networks When number of hidden units

- When number of hidden units < number of input units.

The net performs dimensionality reduction of input image to low dimension representation.

- Each synthesized dimension (each hidden unit) is logistic function of inputs hk (x) = 1 /1+exp(w0+ PN i=1 wi xi) .
- Hence an N-input input layer when input to M unit hidden input layer , we get non-linear transformation of the N-dimesnion to M-dimension data.

PCA:
Given data points in d-dimensional space, project into lower dimensional space while preserving as much information as possible (linear transformation)

No hidden layer (Logistic regression)

Neural net 25 – 10

The PCA data contained 25 feature representation of the image data given to us in previous assignment . Training neural net with no hidden layer directly on this gives us model with best testing accuracy of 85 percentage.

1 hidden layer (17 hidden units)

Neural net 25 – 17 – 10

Best accuracy (testing)achieved is 88.05 percent , with 91 percent training accuracy. Adding one hidden layer help improve the accuracy of the model significantly , because neural net performs non-linear tranformation of this PCA data and learns best out of it.

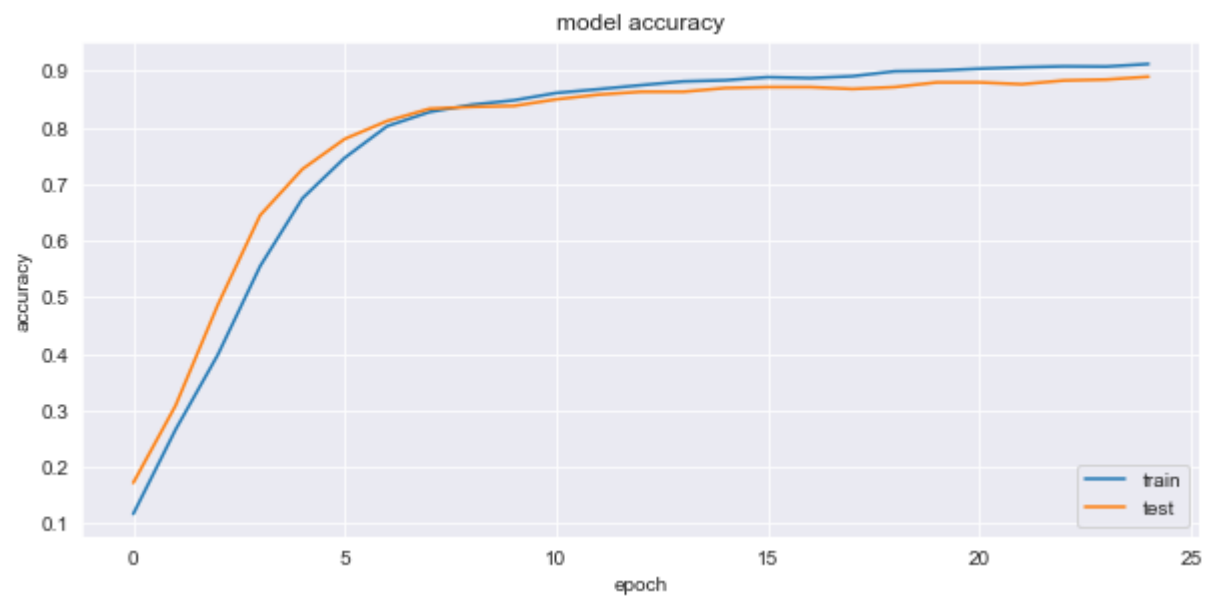
2 hidden layer (20 and 15 hidden units)

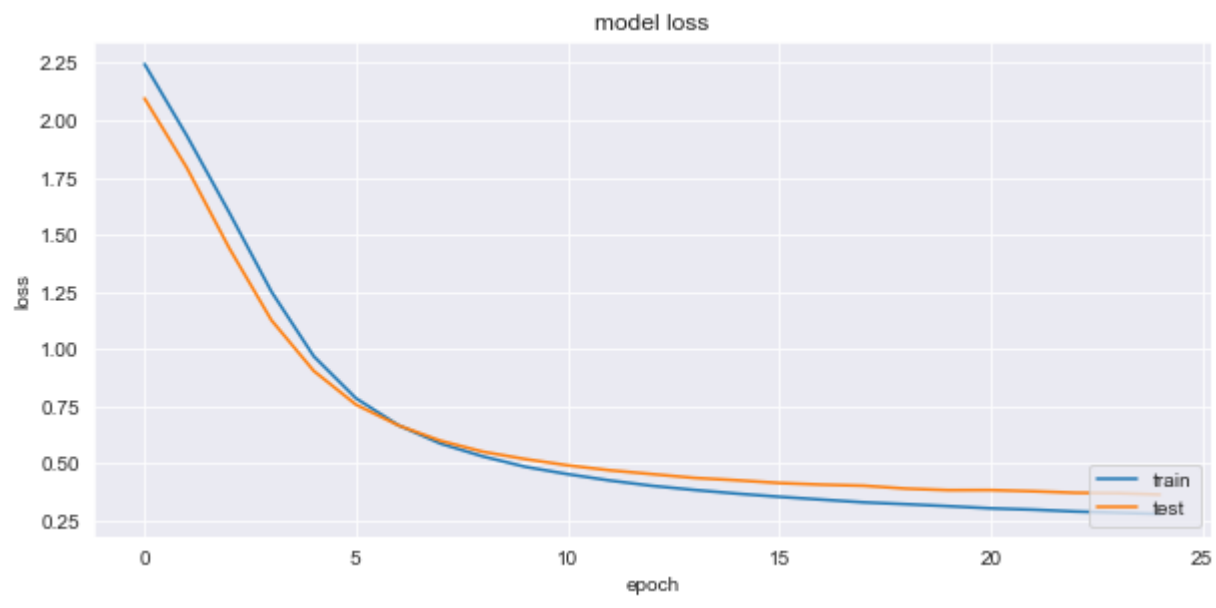
Neural net 25 – 20 – 15 – 10

Best accuracy (testing)achieved is 89 percent , with 92 percent training accuracy. Adding another hidden layer help improve the accuracy of the model slightly.

Comparison with raw pixel data

- Neural net trained on 784 pixel data (94 percent accuracy) performed way better than the neural net trained on PCA features (89 percent accuracy).
- Though PCA does not throw away every other pixel and it only transforms the data to have important features. Keeping maximum variance in the dataset.
- Reducing Dimensions in an image where pixels are the features, would mean downsampling the image





Part 2:

Using Convolutional networks

Convolution solves three important ideas that improves a machinelearning model:

- sparse interactions : In ANN every output unit interacts with every input unit . Convolution network uses sparse connections by taking kernel smaller than the input.
- parameter sharing
- equivariant representations

Stages in Convolutional network

A typical layer of a convolutional network consists of three stages:
In the first stage, the layer performs several convolutions in parallel to produce aset of linear activations.
In the second stage, each linear activation is run through a nonlinear activation function, such as the reLU.
In the third stage, we use a pooling function to modify the output of the layer further

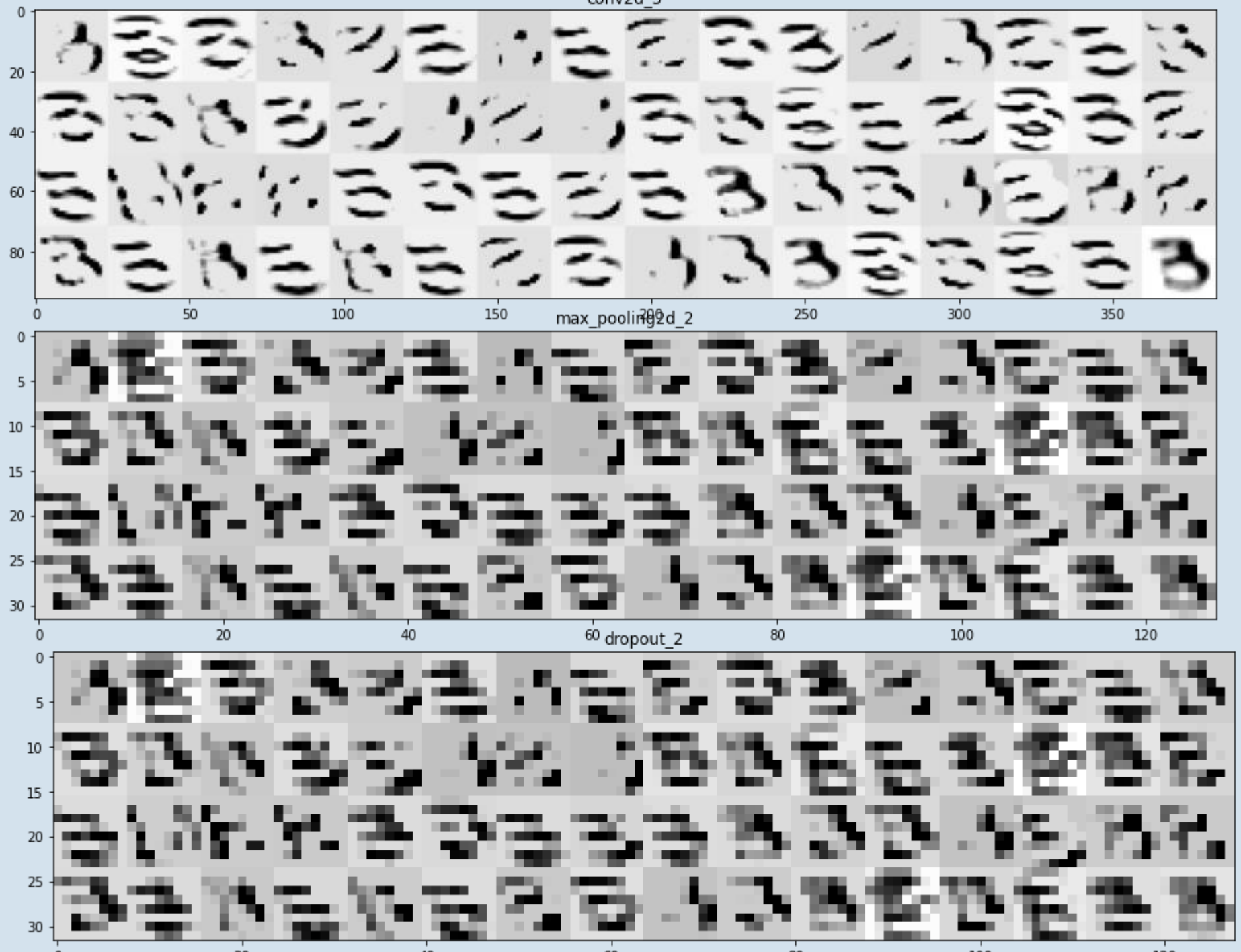
For regularization we can use Dropout after the pooling layer

We used 1 layer of convolution followed by pooling .
Then we use Dropout for regularization purposes .
After 3 of above layers , we then perform our old ANN on the activation function. The code below nicely explains our neural net structure.

```
loss= 0.02745866030454356
accuracy= 0.9904000163078308
```

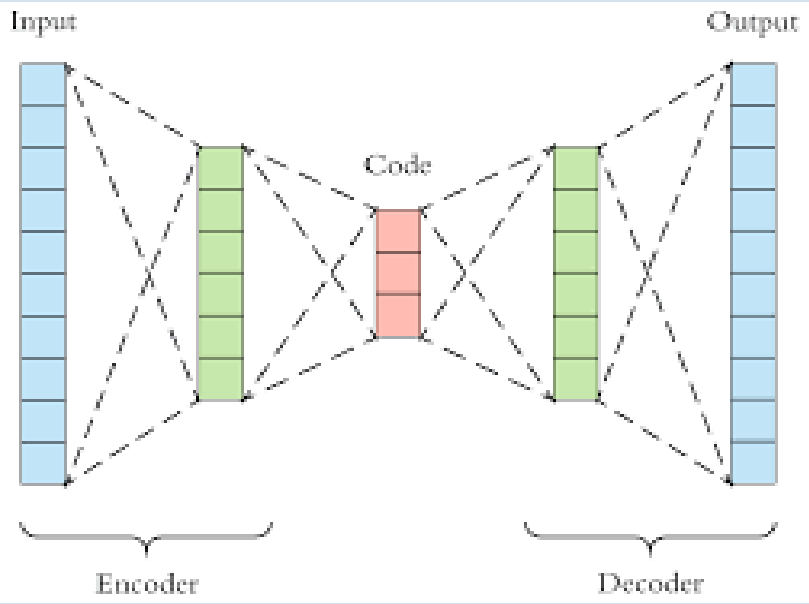
Observations :

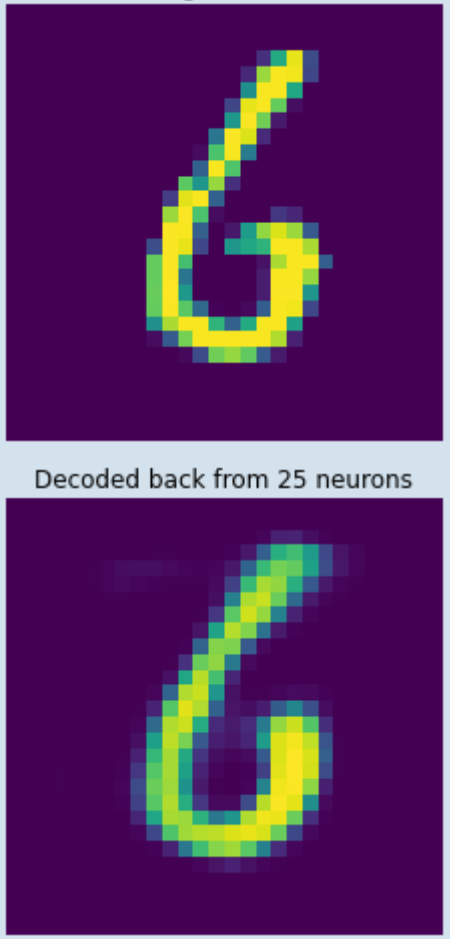
- This first layer retains almost the full shape of the image, and also most of the information present in the image
- As we go deeper into the network we can see that the activations become more complex and abstract. It starts encoding high-level features such as edges, curves and angles.


conv2d_3
max_pooling2d_2
dropout_2

Using Autoencoder :

- Auto-encoders are neural nets used for feature extraction .
- It is unsupervised algorithms , i.e does not used labels. The Key idea is to transform image into low dimension called encoder network ,the reconstructing the image back from the decoder network . The model is trained so that it does this task with least loss.
- Hence we get non-linear transformation of image to lower dimensions.





observation :

- Simple autoencoder neural net i.e just 3 layers with 784,15,784 . Then performed Training on it. Still with so simple autoencoder network we are able to reconstruct the digit images .
- Now after training we encode the training images and get the encoded outputs Train another ANN for the encoded-outputs vs labels (Supervised)
- Got 96 percent validation accuracy with so simple network.