Continuous Evaluation: 70%, Viva: 30%

Assignment 4:

i. Download and install TensorFlow from https://www.tensorflow.org/install/install_sources or using command `sudo pip install tensorflow` (Alternatively the Keras library can be used).

ii. Download the MNIST dataset (contains class labels for digits $0-9$), using the command:

```
import tensorflow as tf
data = tf.contrib.learn.datasets.mnist.load_mnist()
```

or

```
from keras.datasets import mnist
(x_train, y_train), (x_test, y_test) = mnist.load_data()
```

iii. Reduce the training size by 1/10 if computation resources are limited. Define Radial Basis Function (RBF) as

```
def RBF(x, c, s):
   return np.exp(-np.sum(x - c) ** 2, axis = 1)/(2 * s ** 2))
```

where `x` is the actual value, `c` is center (assumed as mean) and `s` is the standard deviation.

Converted 28×28 image into 32×32 using rbf and store the new dataset with the labels. Split the dataset as 80% training and 10% validation and 10% test.

iv. Now run the fully connected network after flattening the data by changing the number of hyper-parameters:

a. Use Gradient Descent Optimizer (learning rate = 0.001) and Squared Error Loss

b. Use Adam Optimizer (learning rate = 0.001) and Categorical Cross Entropy Loss

Hidden Layers	Activation Function	Hidden Neurons
1	Sigmoid	[16]
2	Sigmoid	[16, 32]
3	Sigmoid	[16, 32, 64]

Try all the possible combinations.

v. For the following few tasks, use Adam optimizer (learning rate = 0.001) and Categorical Cross Entropy Loss. Run the network by changing the activation function hyper-parameter:

Hidden Layers	**Activation Function**	**Hidden Neurons**
3	Sigmoid	[16, 32, 64]
3	Tanh	[16, 32, 64]
3	ReLU	[16, 32, 64]

vi. Run the network by changing the number of Dropout hyper-parameters:

Hidden Layers	Activation Function	Hidden Neurons	Dropout
3	ReLU	[16, 32, 64]	0.9
3	ReLU	[16, 32, 64]	0.75
3	ReLU	[16, 32, 64]	0.5
3	ReLU	[16, 32, 64]	0.25
3	ReLU	[16, 32, 64]	0.10

vii. Plot the graph for loss vs epoch and accuracy (train and validation accuracy) vs epoch for all the above cases. Point out the logic in the report.

viii. With the best set hyper-parameters from above run vary the Adam Optimizer learning rate $\{0.01, 0.001, 0.005, 0.0001, 0.0005\}$. Print the time to achieve the best validation accuracy (as reported before from all runs) from all these five run.

ix. Create five images (of size 28×28) containing a digit of your own handwriting and test whether your trained classifier can predict it or not.

Submit a report with the result.