Group Assignment

CS 3120

HAND WRITTEN DIGIT RECOGNIZER

Contents

ABSTRACT

This document presents an Artificial Neural Network to tackle the recognition of human handwritten digits from 0 to 9. A dataset of 42000 sample were obtained from MNIST. The data was trained using gradient decent back-propagation algorithm and further tested feed forward algorithm.

INTRODUCTION

Machine learning, a type of artificial intelligence that "learns" as it identifies new patterns in data, enables data scientists to effectively pinpoint revenue opportunities and create strategies to improve customer experiences using information hidden in huge data.

Any technology user today has benefitted from machine learning. Facial recognition technology allows social media platforms. Optical character recognition technology converts images of text into movable type.

Digit recognizer has been a major area of research in the field of Optical Character Recognition. Based on the input to the system, handwritten digit recognition. There are two approaches for the pattern recognition such as statistical and structural. The structural description of the object is based on the interconnections and interrelationship of features of input data.

Delta Rule

$$\Delta W = \eta \delta x$$

Notice that;

ΔW = $W_{new} - W_{old}$

η : Learning Rate Parameter

δx : Error [= Target ~ Network output]

x : Input

In a feed forward back propagation network net value is calculated in forward direction and the error is propagated backwards from the output nodes to the input nodes. This procedure is done until the error becomes minimum.

Input for the Neural Network

We have used 60 images from each class, 50 for training & 10 for testing. If the training is a success, testing images should be recognized and classified into their respective classes. When we create the input, the bit pattern of each image is taken into consideration. And also preprocessing is extremely important since it eliminates useless pixels.

Training the Network

Matlab command 'train' is used to train the Feed Forward Back Propagation network. The 'train' command trains a network 'net' according to net.trainParam.goal. net = train(net,Input,Target); Here net refers to the network returned by 'newff' function and Input and Target is same as the previous one.

METHODOLOGY

The data set from Kaggle has been used for this project from the below link

https://www.kaggle.com/c/digit-recognizer website. where we obtain two sets of data one for training and the other for testing purpose. The data csv files where each row represent a vectorized form of handwritten digit image, the images are 28x28 matrices so there 784 columns, and the training dataset each row is labeled with what digit does the row represent.

Matlab was used to for the project.

Data preparation

Initially the data are prepared to a form in order to feed the data into the neural network.

```
tr = csvread('train.csv',1,0);
tst = csvread('test.csv',1,0);

tr(tr==0)=10;
Tr = dummyvar(tr(:,1));
Tr = Tr';
trn = tr(:,2:end);
trn = trn';
```

The training data excluding the label column which is the input to the neural network was obtained and transposed.

A target matrix was created such that each column of the matrix represent the expected output for the corresponding column of input data by using label column of the training data with the help of dummyvar function. Here digit 0 was considered 10.

Creating and Training Neural Network

Then the neural network was created and trained with the following code or we can use the nprtool,

```
hiddenLayerSize = 250;
net = patternnet(hiddenLayerSize);

net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
net.trainParam.epochs=1000;
net.trainParam.goal = 0;
net.trainParam.min_grad=0;

[net,tr] = train(net,trn,Tr);
```

Here no. of neurons in the hidden layer is set to 250 and 70% of the train data are taken for training , 15% each for validation and testing.

Testing

A GUI was created in order to test the neural network where any image from the test dataset can be selected and detected for which digit it is.

For this all the rows in the test dataset were converted into matrices and saved as images.

How to run:

Run the create_network.m script and then assign net as my_net.mat and save it.

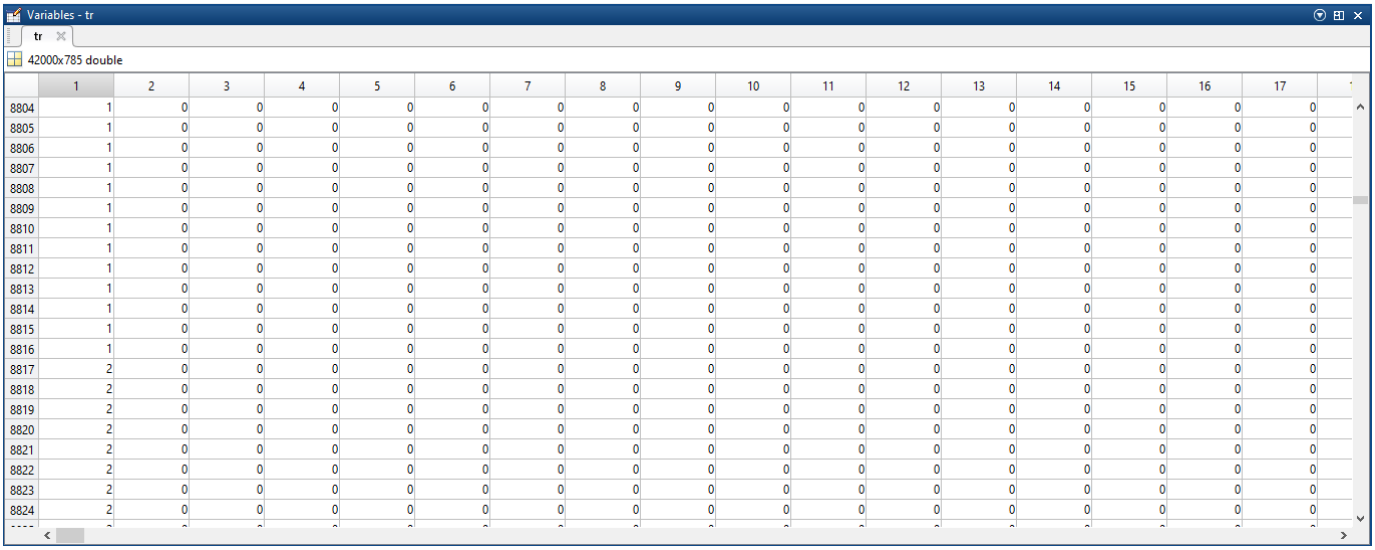
Next run the GUI.m script

Click Browse the Digit button and select the image from image created folder.

Click predict the digit button to get the predicted result.

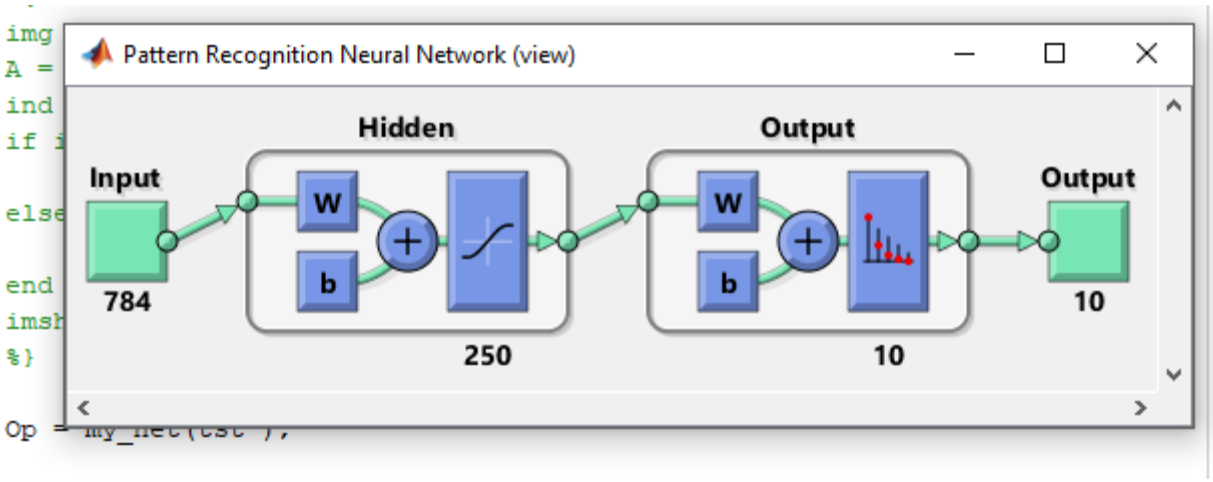
ANALYSIS

The input data which is given was ordered and using csvread() the input matrix with 42000x785 was created.

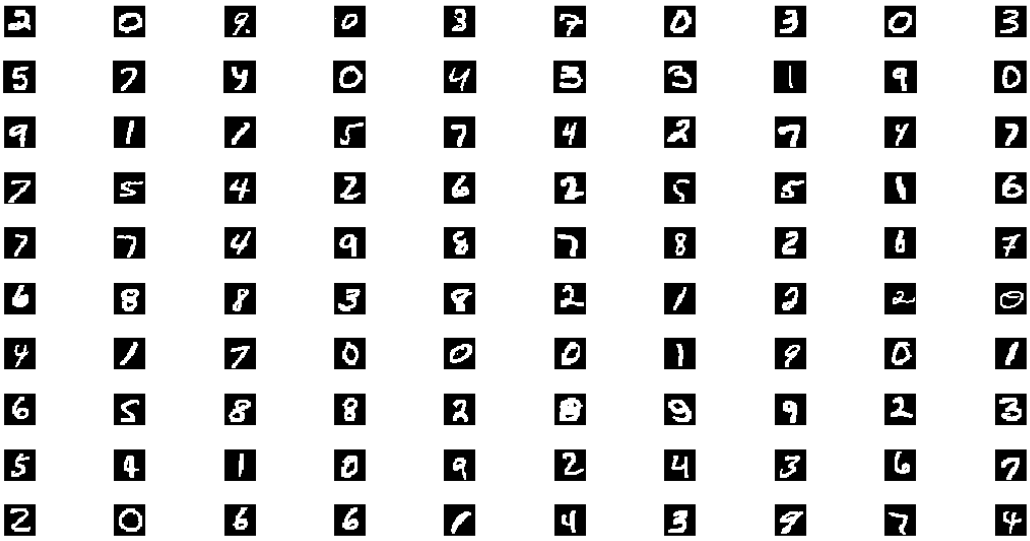


	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
8804	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8805	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8806	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8807	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8808	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8809	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8810	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8811	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8812	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8813	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8814	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8815	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8816	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8817	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8818	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8819	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8820	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8821	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8822	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8823	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8824	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

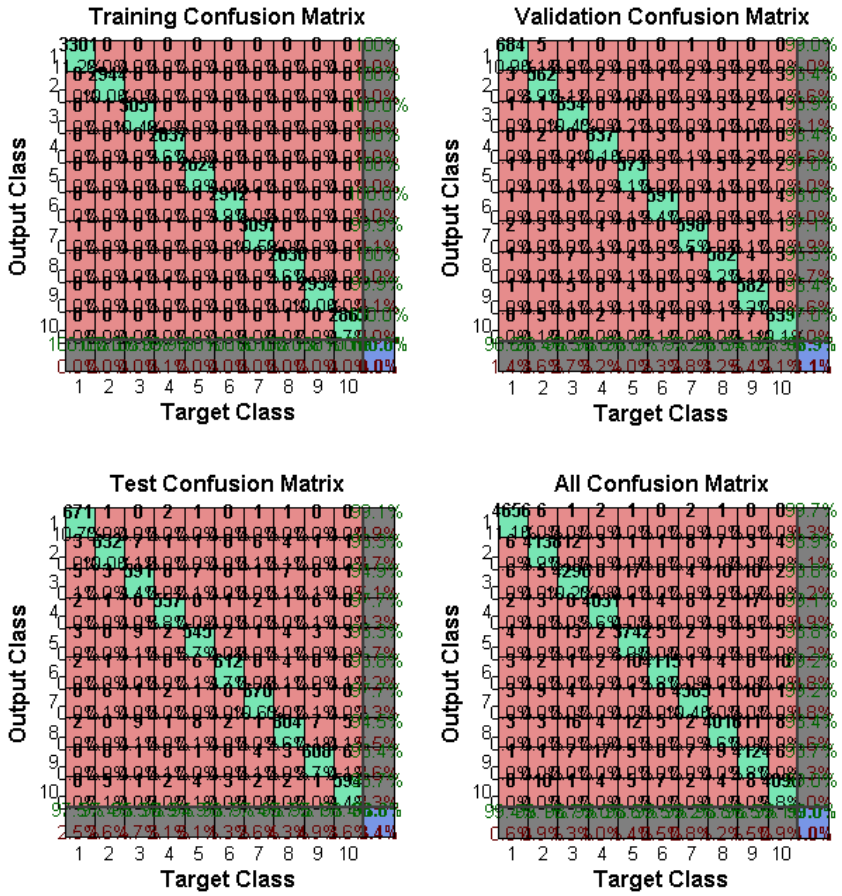
view(net) command can be used visualize the created network to verify the architecture and the type of transfer functions used in each layer.



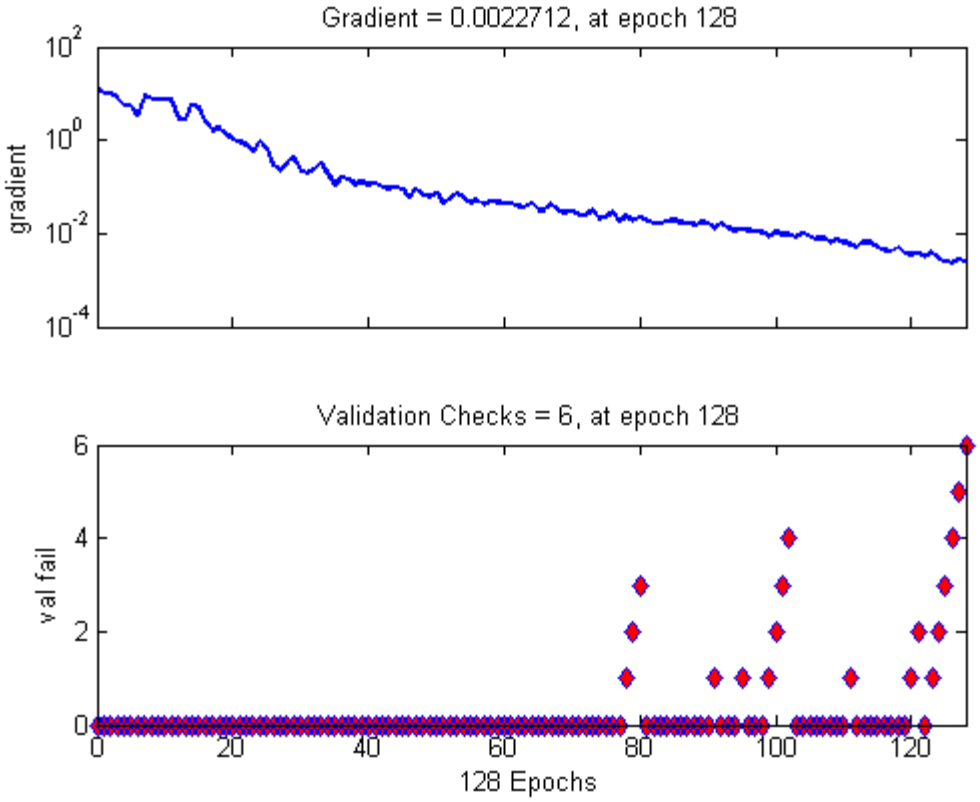
The following figure shows the first 100 test dataset. It shows random collection of from 0 to 9 digits.



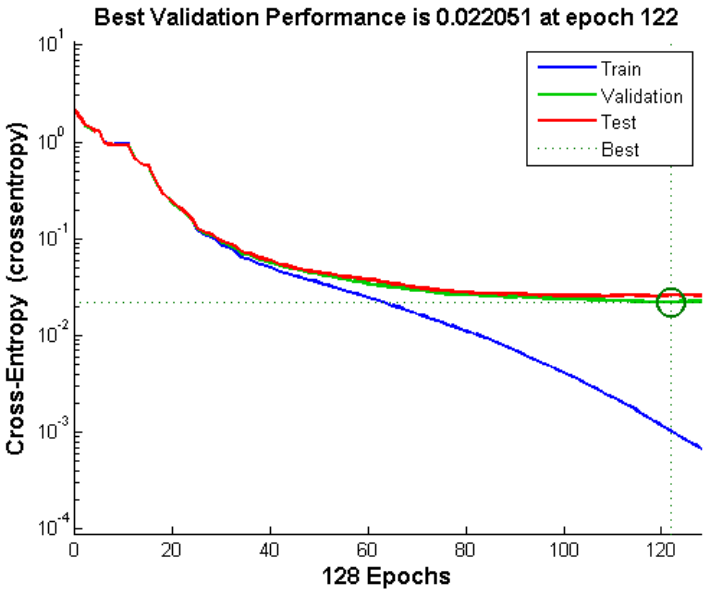
Confusion matrix, specially one that is tells the accuracy of the classifier. It is like an diagonal matrix that can obtained nearly lower values in upper triangular and lower triangular and diagonal will present high values if the neural network is accurate.



Training State

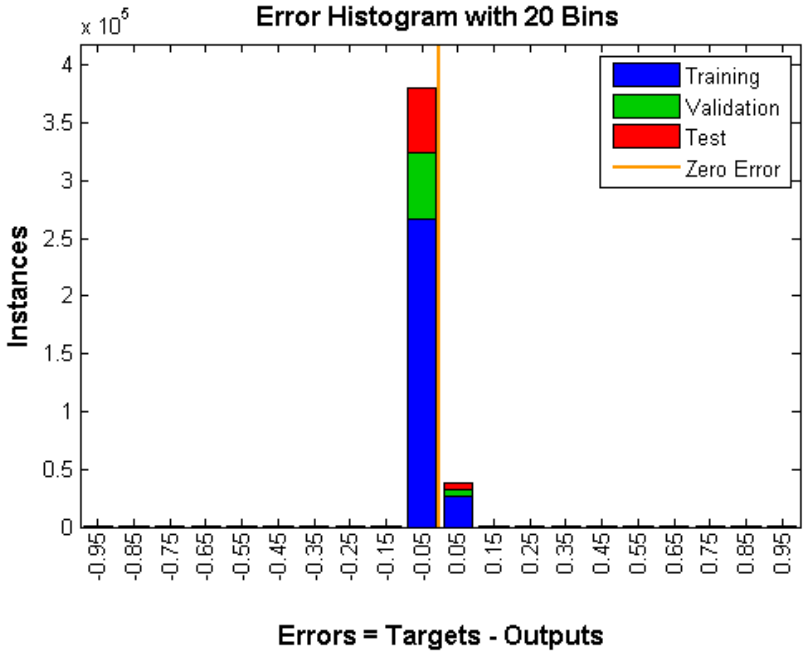


Performance

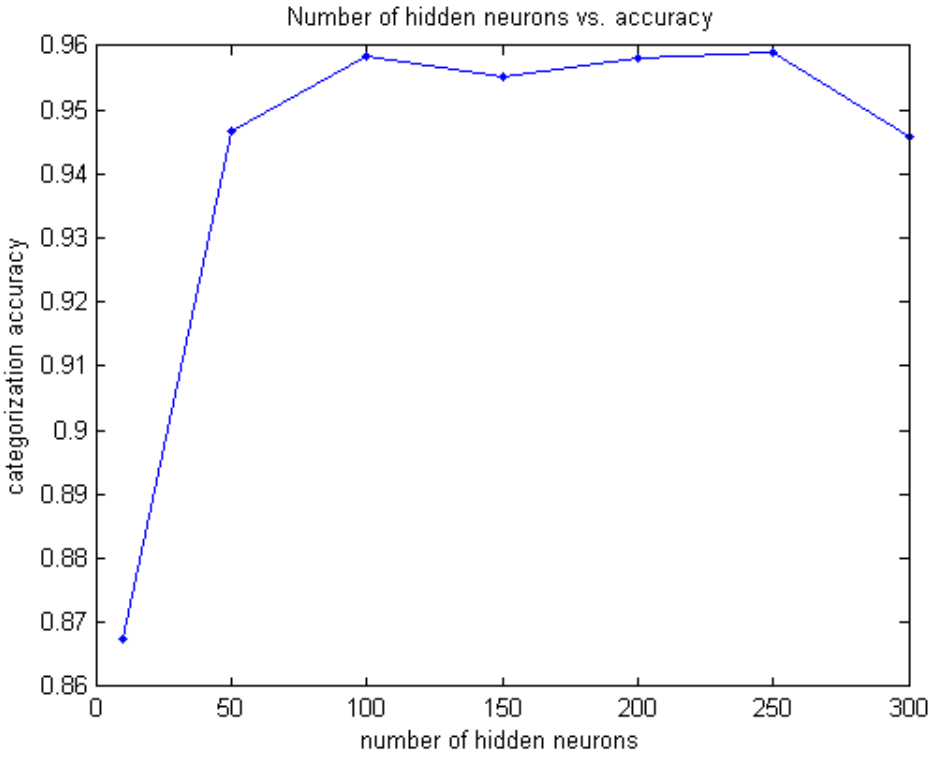


It can be observed that the training ran for 128 epochs. All through the epochs, the error decreased non-linearly, until at the 122th epoch, for the validation and testing data, the error did not decrease. Best validation performance is 0.022051.

Error histogram



Accuracy



Input matrix :- 42000x784

corresponds to the number of digits :- 42000

corresponds to the number of pixel inputs per digit :- 784

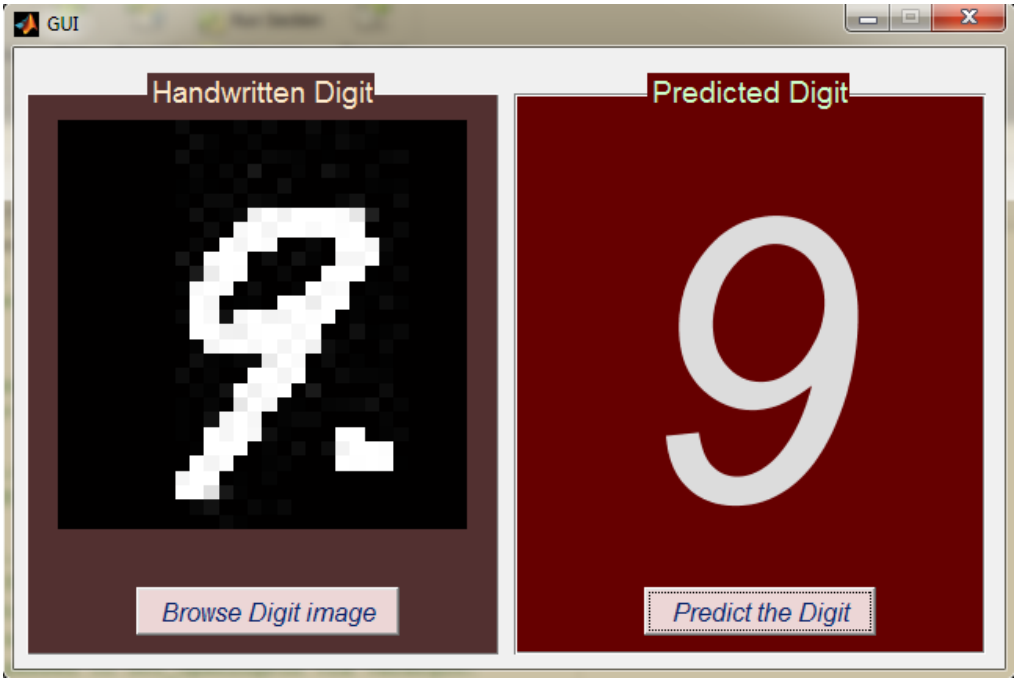
original train data for training :- $2/3^{rds}$

for testing :- $1/3^{rds}$

All of the data is not used for training to make sure the neural network does not over-fit to the data. If over fitting occurs the network may predict digits rather incorrectly. This is prevented by checking the accuracy of the network through a portion of the training data itself.

Output matrix is a 10x1 matrix. 10 corresponds to the 10 individual probabilities per digit, while 1 corresponds to the digit.

Output in GUI : Handwritten digits Recognizer



Above figure presents the graphical user interface of the digit recognizer. It can predict the digit according to the handwritten digit.

CONCLUSION

This particular network is trained to recognize ten digits 1, 2, 3, 4, 5, 6, 7, 8, 9 & 0. For testing set this shows a high accuracy which tests the input digits by means of the training set. Therefore at the moment this network is in a good stance to be used for handwritten digits recognition.

Name	Reg. No	Index no
M.R Mohammed Hasan	2015s15151	S13068
M.H.F.M Haflal	2015s15079	S12990
M.G.Priyadarshani	2015s15235	S12978
R.A.T.P.Ranasinghe	2015s15241	S13029

Contribution

All the group members were get together and did the neural network coding, report, result analysis,etc.