VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Information Technology& Engineering

M.Tech Software Engineering

Subject – SWE2009 – Data Mining & Techniques

Slot : B2+TB2

CHARACTER RECOGNITION USING MACHINE LEARNING

Done By

K.KARTHIKEYAN – 16MIS0102
AKSHAY EBENEZAR.R – 17MIS0449

Faculty Incharge: Prof SUDHA M

APRIL 2019

ABSTRACT

The use of character recognition in automated data-entry applications is described. The processing of the documents on which the characters to be interpreted reside, starts with making electronic images of them. Neural networks are used to recognize the individual characters in the form images. The confidence of each recognition, which is provided by the neural network as part of the classification result, is one of the things used to customize the application to the demands of the client.

LIST OF CONTENTS:

CHAPTER-1

INTRODUCTION

we are use the neural Network to Recognize the character. In this paper it is developed 0ffline strategies for the isolated handwritten English character (A TO Z) and (0 to 9). This method improves the character recognition method. Preprocessing of the Character is used binarization ,thresolding and segmentation method .The proposed method is based on the use of feed forward back propagation method to classify the characters. The ANN is trained using the Back Propagation algorithm. In the proposed system ,English numerical letter is represented by binary numbers that are used as input then they are fed to an ANN. Neural network followed by the Back Propagation Algorithm which compromises Training.

Hand Gesture is habitually used in every day life style. It is so natural way to communicate. Hand gesture recognition method is widely used in the application area of Controlling mouse and/or keyboard functionality, mechanical system, 3D World, Manipulate virtual objects, Navigate in a Virtual Environment, Human/Robot Manipulation and Instruction Communicate at a distance. This paper introduces a real time hand gesture recognition system. This system consists of three stages: image acquisition, feature extraction, and recognition. In the first stage input image of hand gestures are acquiesced by digital camera in **4**

approximate frame rate. In second stage a rotation, translation, scaling and orientation invariant feature extraction method has been introduce to extract the feature of the input image based on moment feature extraction method. Finally, a neural network is used to recognize the hand gestures. The performance of the system tested on real data. Based on the experimental results, we noted that this system shows satisfactory performance in hand gesture recognition.

1.1 OBJECTIVE OF THE WORK

The recognition of handwritten characters has many applications such as automatic Postal sorting, automatic bank cheque processing etc. In the work on character recognition has been reviewed. The earlier systems known as Optical Character Recognition (OCR) systems that had been developed were confined to recognize only the printed or handwritten characters of fixed size and fonts. But, the present study aims at producing a system, which could recognize characters of any arbitrary size, shape and fonts. There are numerous approaches that address the problem and they vary in the features extracted from the graphical representation of the Characters.

1.2 SCOPE OF THE WORK:

The first phase in our character recognition process is converting the image to Binary image by thresholding the given character image. Binary images are images whose pixels have only two possible intensity values. They are normally displayed as Black and White. The converted Character image has pixel value zero for Black and one for white. Thus the color of the character is White and the background **i**s black

LITERATURE SURVEY:-

LINK FOR SURVEY MATERIALS:- https://karthikeyank2016.wixsite.com/mysite

Topics	published	Algorithim	Advantage	Disadvantage
Neural Network based Approach for Recognition of Text Images	January 2013	Multilayer perceptron network	As neural network is used here for recognition of offline English character images and it has been seen that recognition increases	There is sometimes result variation may be due to the number of character set used for training was reasonably low.
Character Recognition Using Neural Network	April 2013	Back propagation	Neural network based method gives the accuracy 85 %.	It can cause significant degradation in the feature extraction process
Handwritten English Character Recognition Using Neural Network	December 2010	Multilayer Perceptron Network	A lot of efforts have been made to get higher accuracy but stil there are tremendous scope of improving	English character recognition is giving high recognition accuracy and minimum training time

			recognition accuracy	
HAND WRITTEN CHARACTER RECOGNITION USING BACK PROPAGATION NETWORK	2005 - 2009	Back propagation Network	The BPN network designed proposed has the ability to recognize stimulus patterns without affecting by shift in position not by a small distortion in shape of input pattern.	To illustrate some problems that often arise when we are attempting to automate complex pattern-recognition
Character Recognition using Back Propagation Neural Network	September 2012	Back propagation network	The use of gradient decent back propagation algorithm has improved the performance of neural network.	Not have perfect accuracy
A Matlab Project in Optical Character Recognition (OCR)	2000	Feed forward network		More than one same character cannot be possible
HAND-WRITTEN CHARCTER RECOGNITION	2014	Neural network	The result which was got was correct up to more than 90% of the	In fact all the handwritten characters remain irregular.

			cases, but it would be improved at the end	
Character Recognition Using Matlab's Neural Network Toolbox	FEBRUARY 2013	Neural network (extraction,detection algorithim)	A simplistic approach for recognition of Optical characters using artificial neural networks has been described.	A poorly chosen set of features will yield poor classification rates by any neural network

A Survey on Optical Character Recognition System	Decembe r 2016	Neural network	During the early days, OCR has been used for mail sorting, bank cheque reading and signature verification	The time and space complexity of a post processor should some what very high
Literature Survey on Recognition and Evaluation of Optical Character Recognition (OCR)	Febraury 2018	Feed forward network	Our evaluation shows that LBP with SVM gives optimal results with accuracy of 96.5%.	Visual images are also subject to noise and therefore, there are issues particularly over edges.
A SURVEY ON HANDWRITTEN CHARACTER RECOGNITION (HCR) TECHNIQUES	March 2016	Radial base function	The sliding window scheme defends the left to right scripting	The existing HCR for handwritten has very less precision. We

FOR ENGLISH ALPHABETS			nature of the article as fine as the inconsistent alphabet extent characteristics.	require a proficient solution to resolve this difficulty so that overall performance can be amplified.
FUZZY LOGIC BASED HANDWRITTEN CHARACTER RECOGNITION	Not known	Fuzzy logic	The paper presents a fuzzy logic based approach for the recognition of isolated handwritten characters. The normalized angle approach (using fuzzy distance) gave the best rate of 83%.	However, some characters (such as 'E' and 'F', a','e','b')
Hand written character recognition using image fusion and fuzzy logic	2016	Image fusion and fuzzy logic	Fully automated	Not correctly recognize the letter "f"
HANDWRITTEN TEXT IMAGE AUTHENTICATION USING BACK PROPAGATION	September 2011	Back propagation algorithim	More accuracy	While training the network using two or more similar patterns which represent the same output can avoid restriction on width and height.

Multi-Layer Perceptron Network For English Character Recognition	2017	Multilayer perceptron network	recognize these characters with high performance accuracy and reduced training time as well as minimum classification time.	accuracy of recognitionat 95%.
HANDWRITTEN DIGIT RECOGNITION USING BACK PROPAGATION NEURAL NETWORK& K-NEAREST NEIGHBOUR CLASSIFIER	JULY 2013	k-NearestNeighbour algorithm and Back propagation Algorithm.	the classifier can be chosen as per the need and a trade-off between speed and accuracy is possible depending on the application.	It can be seen that the accuracy is above 90% for this trained neural network. The accuracy remains 88% on an average considering results of all the digits from 0 to 9.
Handwritten Character and Digit Recognition Using Artificial Neural Networks	APRIL 2013	Artificial neural network	drawing character will be 'X' in this 'X' will be recognize 83% high	Here error rate is 1.089 and Iterations will Be 78 and the time is 03 sec .
Noisy Character Recognition	2017	Single layer perceptron	The algorithm is also capable of the recognition	noisy characters with 17.45%, 33.19%, 87.66% noise

			of 82.5% noisy Characters having more noise (100%, 116.17%).	of the input pattern
CHARACTER RECOGNITION USING NEURAL NETWORKS	Not known	Neural networks	they are robust with respect to input noise, node failure, can adapt to input stimulus	Accuracy rate is hogh
A STUDY ON JAPANESE HISTORICAL CHARACTER RECOGNITION USING MODULAR NEURAL NETWORKS	October 2010	K-means algorithim	We confirmed this historical character recognition system has quite high recognition accuracy of 95.35% for 57 character categories, which contain more categories than the conventional research.	We confirmed to obtain quite high recognition accuracy of 95.35% for 57 character categories,
Pattern Recognition using Artificial Neural Network	2014	Artificial neural networks	With distortion the accuracy level of output is more in new defined network as compared to only BPN	As fault tolerance of HP is more than BPN so the error calculating capability is more in HP

Handwritten Farsi Character Recognition using Artificial Neural Network	2009	Multilayer perceptron network	We have collected 250 samples of handwritten Farsi characters written by ten different persons 25 each directly on screen. We have used 125 samples as a training data (training set) and remaining 125 samples as a test data (test set).	An experimental result shows that backpropagation network yields good recognition accuracy of 85%
OPTICAL CHARACTER RECOGNITION USING ARTIFICIAL NEURAL NETWORK	2016	Multilayer perceptron network	Recognized character will be store in file in the editable and accessible format.	.Only printed data recognition is not sufficient in today's scenario so the next step of this proposed system is to concentrate on handwritten text recognition with high accuracy.
Training Feed forward Neural Network With Backpropogation Algorithm	Januaray 2017	Back propagation network	BP Algorithm is known for its mathematical simplicity and accuracy	BP has its own limitations of slow convergence rate and local minima problem which is still a big

				problem when dealing with large complex problems.
A Review on Geometrical Analysis in Character Recognition	April 2015	Neural network	More accuracy	Improvement can be made in the existing geometrical character recognition techniques by finding new techniques for preprocessing.
Unicode Optical Character Recognition Using Neural Networks	October 2015	Some artificial neural network	Properly adjusted	Optical character recognition using neural networks involves various stages that needs to be implemented correctly else the whole process will fall back
Neuro chain:Handwritten recogntion	Not known	Artificial neural networks	The highest performance obtained is 70%	Even 30% not get it
Online Handwritten Character Recognition Using an Optical Backpropagation Neural Network	2005	Back propagation algorithim	The experimental results show that using the OBP can speed up convergence of the training process. Although small values	To obtain fair and independent results, two different architectures were used each with various values for the learning rate.

			for the learning rate were used,	
GradientBased Learning Applied to Document Recognition	1998	Multilayer feed forward network		No accuracy
Character Recognition Using Convolutional Neural Networks	2006-2007	Convolutional neural networks	. It was shown, that the concept of convolutional neural networks and weight sharing not only reduces the need for computation, but also offers a satisfying degree of noise resistance and invariance to various forms of distortion	With an error rate as low as 0.95%
Survey On Optical Character Recognition Using Neural Network	June 2017	Back propagation algorithim	artificial neural networks are commonly used to perform character recognition due to their high noise tolerance	it is unlikely to replace existing OCR methods, especially for English text
A Survey on Handwritten Character	Decembe r 2016	Multilayer perceptron Neural network	Strongly chosen feature set	Create computing algorithm with

Recognition using Multilayer Perceptron Neural Networks			provide good recognition rate and poorly chosen set provide poor recognition rate.	hundred percent accuracy is not possible

2.2 BACKGROUND

This project is based on Machine learning, We can provide a lot of data set as an Input to the software tool which will be recognized by the machine and similar pattern will be taken out from them.by use of some machine learning algorithim we bring out output. We can use Matlab tool for implementation .

2.3 CHALLENGES

We have faced some challenges in which the algorithim works perfectly in some epochs we take more time to identify which epoch suits better for the backprogation algorithim.

2.4 PROBLEM DEFINITION AND APPROACH:

This paper describe how handwritten English character Recognition (HCR) processed, trained and then recognized using Back propagation method. Size and fonts are different in training the data and testing the data. In the present paper, we have given a method to recognize a handwritten character using back propagation method. It is developed for isolated handwritten English Characters (A to Z). Preprocessing of Recognition is used binarization, thresolding and segmentation method. Image are first converted into gray scale then features are extracted which are in form of 0 and 1. 780 hand written characters are used in database for characters then test the data and find the Recognition accuracy.

CHAPTER – 3

EXPERIMENTAL DETAILS

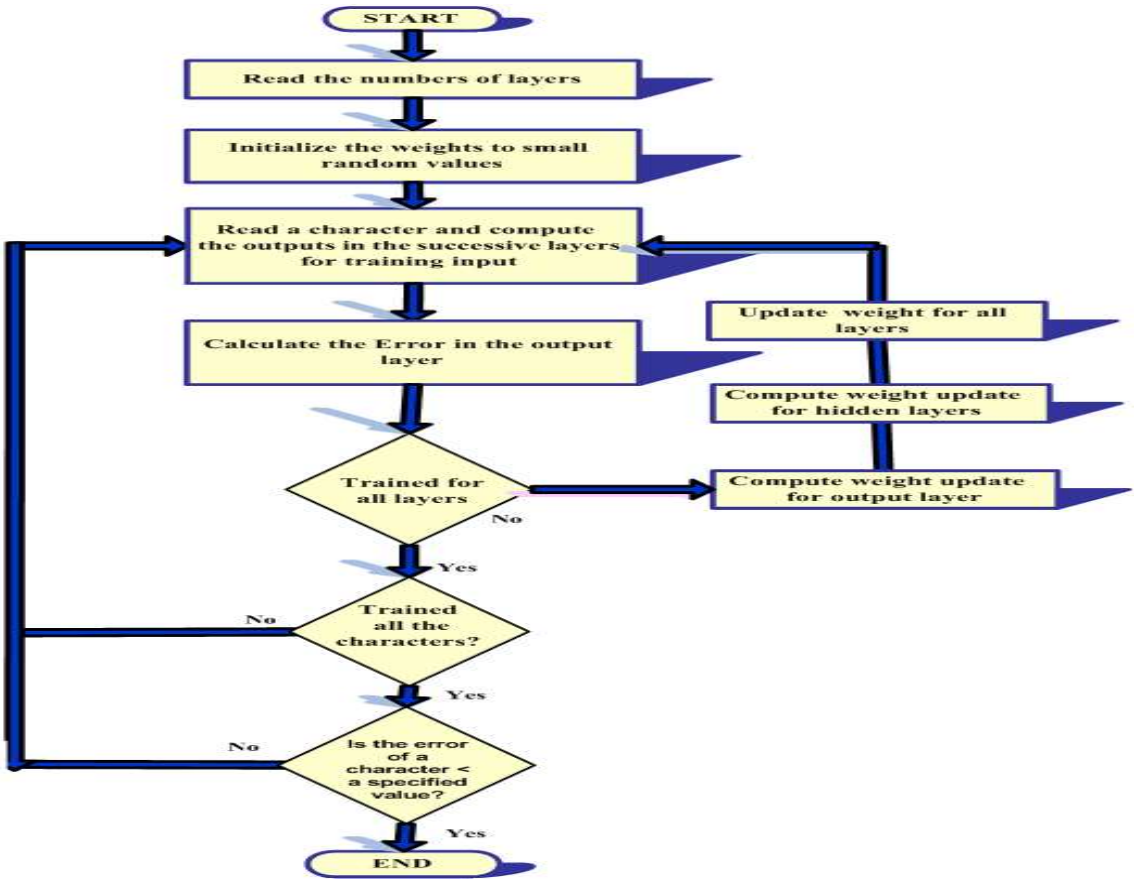
3.1 MACHINE LEARNING METHODS

Back propagation algorithim

The Back propagation Principle

Backpropagation is a special case of a more general technique called automatic differentiation. In the context of learning, **backpropagation** is commonly used by the gradient descent optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function.

- Comparing two set of datasets (A1-Z1) and (A2-Z2) first the train the network using back propagation algorithim then find error rate accordingly doing for some 100 epochs finally insert a input data set and finally the output value data set got .
- Here the for performance and to reduce error we using some function called erf function



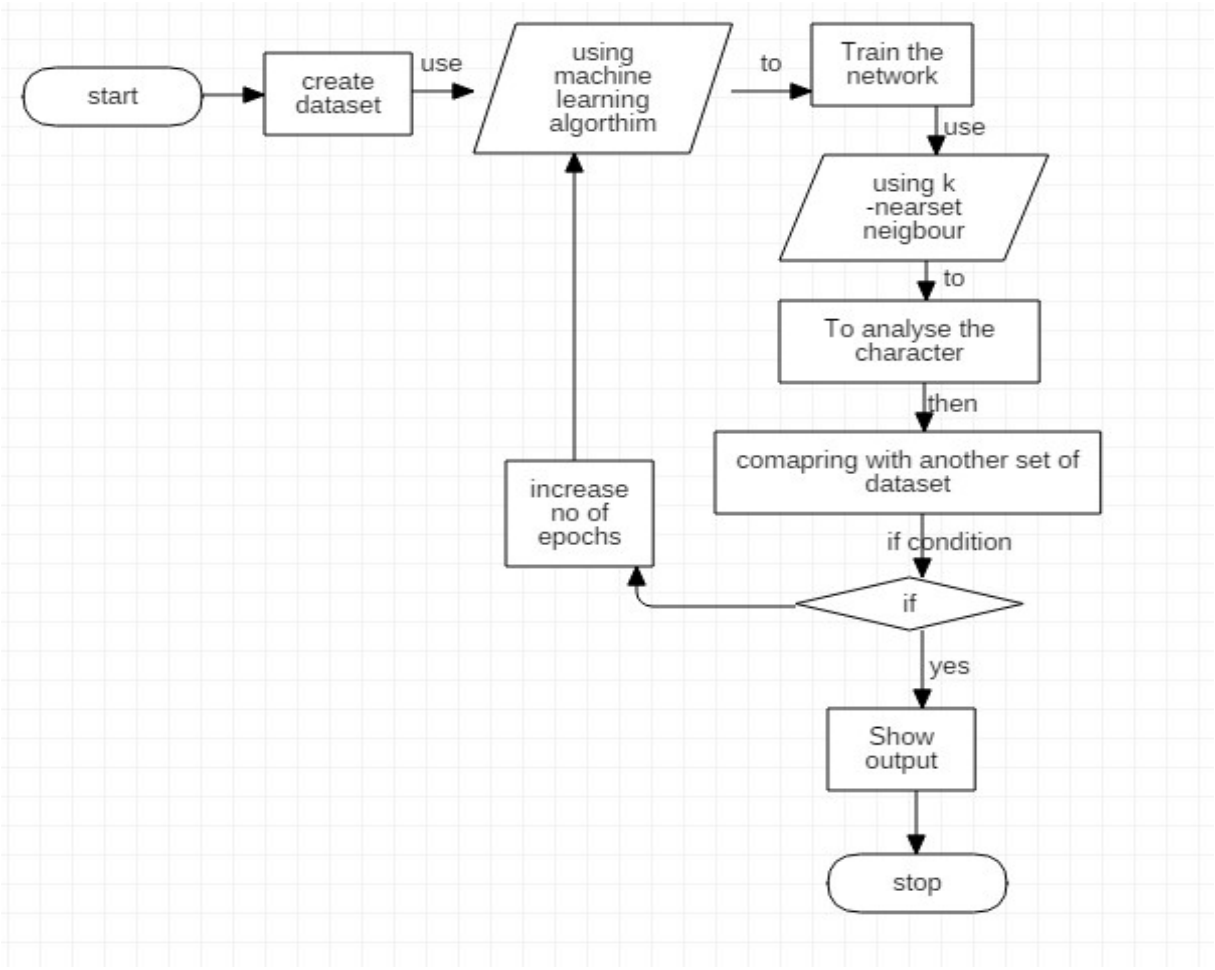
Algorithim using matlab code

```
%3. Network Input

p=[A2(1:end);B2(1:end);C2(1:end);D2(1:end);E2(1:end);F2(1:end);G2(1:end);H2(1:e
nd);I2(1:end);J2(1:end);K2(1:end);L2(1:end);M2(1:end);N2(1:end);O2(1:end);P2(1:e
nd);Q2(1:end);R2(1:end);S2(1:end);T2(1:end);U2(1:end);V2(1:end);W2(1:end);X2(1
:end);Y2(1:end);Z2(1:end)]';

%4. Desired Target Network
t=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26];
% pr = min max function

PR=zeros(9,9);
PR(:,2)=1;
%5. Use of Backpropagation Functions
net=newff(minmax(p),[26,1],{'logsig','purelin'});
%6. Network Training
net.trainParam.epochs=100;
net.trainParam.lr=0.5;
erf(p)
%error function
net=train(net,p,t);


%7. Experiment Input Patterns with Other Models
% sim = testing
results=sim(net,A2(1:end)');
y=results;

results=round(y);
disp('calculated value');
disp(y)
disp(results)
if results==1'
disp('Alphabet A');
elseif results==2'
disp('Alphabet B');
elseif results==3'
disp('Alphabet C');
elseif results==4'
disp('Alphabet D');
elseif results==5'
```

```
disp('Alphabet E');
elseif results==6'
disp('Alphabet F');
elseif results==7'
disp('Alphabet G');
elseif results==8'
disp('Alphabet H');
elseif results==9'
disp('Alphabet I');
elseif results==10'
disp('Alphabet J');
elseif results==11'
disp('Alphabet K');
elseif results==12'
disp('Alphabet L');
elseif results==13'
disp('Alphabet M');
elseif results==14'
disp('Alphabet N');
elseif results==15'
disp('Alphabet O');
elseif results==16'
disp('Alphabet P');
elseif results==17'
disp('Alphabet Q');
elseif results==18'
disp('Alphabet R');
elseif results==19'
disp('Alphabet S');
elseif results==20'
disp('Alphabet T');
elseif results==21'
disp('Alphabet U');
elseif results==22'
disp('Alphabet V');
elseif results==23'
disp('Alphabet W');
elseif results==24'
disp('Alphabet X');
elseif results==25'
disp('Alphabet Y');
elseif results==26'
disp('Alphabet Z');
else
```

```
disp('Letter not recognized')
end
```

3.2 DESIGN FRAMEWORK



3.3 DATSET, DATASOURCE, CHARACTERIZATION, PREPROCESSING

DATASETS

```
%1. Formatting Patterns for Introduction
A1=[0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 1 0 1 0 0;
0 0 1 0 1 0 0;
0 1 0 0 0 1 0;
0 1 1 1 1 1 0;
0 1 0 0 0 1 0;
0 1 0 0 0 1 0];
B1=[1 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 0 1 1 0];
C1=[0 0 1 1 1 0 0;
0 1 0 0 0 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0];
D1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1];
E1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
```

```
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1];
F1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0];

G1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1];
H1=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1];
I1=[1 1 1 1 1 1 1;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
1 1 1 1 1 1 1];
```

```
J1=[1 1 1 1 1 1 1;
1 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
1 1 1 1 0 0 0];
K1=[1 0 0 0 0 0 0;
1 0 0 0 1 0 0;
1 0 0 1 0 0 0;
1 0 1 0 0 0 0;
1 1 0 0 0 0 0;
1 0 1 0 0 0 0;
1 0 0 1 0 0 0;
1 0 0 0 1 0 0;
1 0 0 0 0 1 0];
L1=[1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1];
M1=[1 0 0 0 0 0 1;
1 1 0 0 0 1 1;
1 0 1 0 1 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1];
N1=[0 0 0 0 0 0 0;
1 0 0 0 0 0 1;
1 1 0 0 0 0 1;
1 0 1 0 0 0 1;
1 0 0 1 0 0 1;
1 0 0 0 1 0 1;
1 0 0 0 0 1 1;
1 0 0 0 0 0 1;
```

```
1 0 0 0 0 0 1];
O1=[0 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 1 1 1 1 0];
P1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0];
Q1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 1 0 1;
1 0 0 0 1 0 1;
1 1 1 1 1 1 1;
0 0 0 0 1 1 1];
R1=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 1 0 0 0;
1 0 0 0 1 0 0;
1 0 0 0 0 1 0;
1 0 0 0 0 0 1];
S1=[0 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
0 0 0 0 0 0 1;
0 0 0 0 0 0 1;
```

```
0 0 0 0 0 0 1;
1 1 1 0 1 1 0];
T1=[1 1 1 1 1 1 1;
1 0 0 1 0 0 1;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0];
U1=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 1 1 1 1 0];
V1=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0;
0 0 1 0 1 0 0;
0 0 0 0 0 0 0;
0 0 0 1 0 0 0];
W1=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 1 0 1 0 1;
0 1 0 0 0 1 0];
X1=[1 0 0 0 0 0 1;
0 1 0 0 0 0 1;
0 0 1 0 0 1 0;
0 0 0 1 1 0 0;
0 0 0 1 1 0 0;
0 0 1 0 1 0 0;
```

```
0 1 0 0 0 1 0;
1 0 0 0 0 1 0;
1 0 0 0 0 0 1];
Y1=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0];

Z1=[1 1 1 1 1 1 1;
0 0 0 0 0 0 1;
0 0 0 0 0 1 0;
0 0 0 0 1 0 0;
0 0 0 1 0 0 0;
0 0 1 0 0 0 0;
0 1 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1];


%Patterns for Experiments (Other Letter Patterns)
A2=[0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 1 0 1 0 0;
0 0 1 0 1 0 0;
0 1 0 0 0 1 0;
0 1 1 1 1 1 0;
0 1 0 0 0 1 0;
0 1 0 0 0 1 1];
B2=[1 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 0 1 1 0];
C2=[0 0 1 1 1 0 0;
```

```
0 1 0 0 0 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0];
D2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1];
E2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1];
F2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0];
G2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1];
```

```
H2=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1];
I2=[1 1 1 1 1 1 1;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
1 1 1 1 1 1 1];
J2=[1 1 1 1 1 1 1;
1 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
1 1 1 1 0 0 0];
K2=[1 0 0 0 0 0 0;
1 0 0 0 1 0 0;
1 0 0 1 0 0 0;
1 0 1 0 0 0 0;
1 1 0 0 0 0 0;
1 0 1 0 0 0 0;
1 0 0 1 0 0 0;
1 0 0 0 1 0 0;
1 0 0 0 0 1 0];
L2=[1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
```

```
1 1 1 1 1 1 1];
M2=[1 0 0 0 0 0 1;
1 1 0 0 0 1 1;
1 0 1 0 1 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1];
N2=[0 0 0 0 0 0 0;
1 0 0 0 0 0 1;
1 1 0 0 0 0 1;
1 0 1 0 0 0 1;
1 0 0 1 0 0 1;
1 0 0 0 1 0 1;
1 0 0 0 0 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1];
O2=[0 1 1 1 1 1 0;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 1 1 1 1 0];
P2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0];
Q2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 1 0 1;
1 0 0 0 1 0 1;
```

```
1 1 1 1 1 1 1;
0 0 0 0 1 1 1];
R2=[1 1 1 1 1 1 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 1 1 1 1 1 1;
1 0 0 1 0 0 0;
1 0 0 0 1 0 0;
1 0 0 0 0 1 0;
1 0 0 0 0 0 1];
S2=[0 1 1 1 1 1 1;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1;
0 0 0 0 0 0 1;
0 0 0 0 0 0 1;
0 0 0 0 0 0 1;
1 1 1 0 1 1 0];
T2=[1 1 1 1 1 1 1;
1 0 0 1 0 0 1;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0];
U2=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 1 1 1 1 0];
V2=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0;
```

```
0 0 1 0 1 0 0;
0 0 0 0 0 0 0;
0 0 0 1 0 0 0];
W2=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 0 1 0 0 1;
1 0 1 0 1 0 1;
0 1 0 0 0 1 0];
X2=[1 0 0 0 0 0 1;
0 1 0 0 0 0 1;
0 0 1 0 0 1 0;
0 0 0 1 1 0 0;
0 0 0 1 1 0 0;
0 0 1 0 1 0 0;
0 1 0 0 0 1 0;
1 0 0 0 0 1 0;
1 0 0 0 0 0 1];
Y2=[1 0 0 0 0 0 1;
1 0 0 0 0 0 1;
0 1 0 0 0 1 0;
0 0 1 0 1 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0;
0 0 0 1 0 0 0];
Z2=[1 1 1 1 1 1 1;
0 0 0 0 0 0 1;
0 0 0 0 0 1 0;
0 0 0 0 1 0 0;
0 0 0 1 0 0 0;
0 0 1 0 0 0 0;
0 1 0 0 0 0 0;
1 0 0 0 0 0 0;
1 1 1 1 1 1 1];
```

Data Set Information:

Data set information for character recognition contains binary representation of data set.

Attribute Information:

Black =1

White =0

CHAPTER – 4

RESULT AND DISCUSSION

SOURCE CODE FOR BACK PROPAGATION ALGORITHIM:

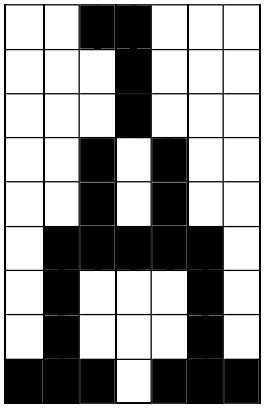
```
%3. Network Input

p=[A2(1:end);B2(1:end);C2(1:end);D2(1:end);E2(1:end);F2(1:end);G2(1:end);H2(1:e
nd);I2(1:end);J2(1:end);K2(1:end);L2(1:end);M2(1:end);N2(1:end);O2(1:end);P2(1:e
nd);Q2(1:end);R2(1:end);S2(1:end);T2(1:end);U2(1:end);V2(1:end);W2(1:end);X2(1
:end);Y2(1:end);Z2(1:end)]';

%4. Desired Target Network
t=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26];
% pr = min max function

PR=zeros(9,9);
PR(:,2)=1;
%5. Use of Backpropagation Functions
net=newff(minmax(p),[26,1],{'logsig','purelin'});
%6. Network Training
net.trainParam.epochs=100;
net.trainParam.lr=0.5;
erf(p)
%error function
net=train(net,p,t);


%7. Experiment Input Patterns with Other Models
% sim = testing
results=sim(net,A2(1:end)');
y=results;

results=round(y);
disp('calculated value');
disp(y)
disp(results)
if results==1'
disp('Alphabet A');
elseif results==2'
disp('Alphabet B');
elseif results==3'
disp('Alphabet C');
elseif results==4'
```

```
disp('Alphabet D');
elseif results==5'
disp('Alphabet E');
elseif results==6'
disp('Alphabet F');
elseif results==7'
disp('Alphabet G');
elseif results==8'
disp('Alphabet H');
elseif results==9'
disp('Alphabet I');
elseif results==10'
disp('Alphabet J');
elseif results==11'
disp('Alphabet K');
elseif results==12'
disp('Alphabet L');
elseif results==13'
disp('Alphabet M');
elseif results==14'
disp('Alphabet N');
elseif results==15'
disp('Alphabet O');
elseif results==16'
disp('Alphabet P');
elseif results==17'
disp('Alphabet Q');
elseif results==18'
disp('Alphabet R');
elseif results==19'
disp('Alphabet S');
elseif results==20'
disp('Alphabet T');
elseif results==21'
disp('Alphabet U');
elseif results==22'
disp('Alphabet V');
elseif results==23'
disp('Alphabet W');
elseif results==24'
disp('Alphabet X');
elseif results==25'
disp('Alphabet Y');
elseif results==26'
```

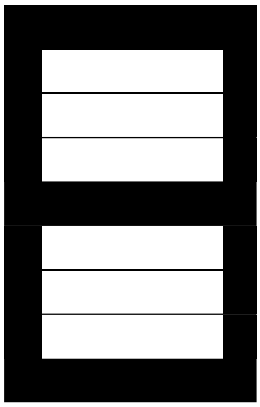
```
disp('Alphabet Z');
else
disp('Letter not recognized')
end
```

METHODOLOGY:-

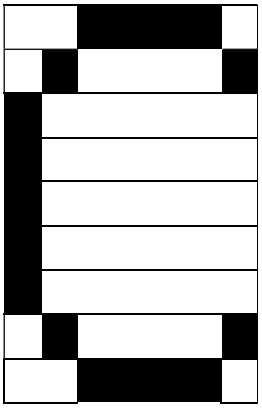
Pattern Letters



The Letter ‘A’



The Letter ‘B’



The Letter ‘C’

Patterns using 9x7 matrix patterns. Each color element is exemplified when:

a. Black color worth 1

b. Colorless is 0

Create Code for Build Back Propagation Network in Command Editor

```
clear all;

Network                                                                    Input
p=[A2(1:end);B2(1:end);C2(1:end);D2(1:end);E2(1:end);F2(1:end);G2(1:end);H2(
1:end);I2(1:end);J2(1:end);K2(1:end);L2(1:end);M2(1:end);N2(1:end);O2(1:end);
P2(1:end);Q2(1:end);R2(1:end);S2(1:end);T2(1:end);U2(1:end);V2(1:end);W2(1:e
nd);X2(1:end);Y2(1:end);Z2(1:end)]';
```

```
  Desired Target Network
t=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26];
PR=zeros(63,2);
PR(:,2)=1;

  Use of Backpropagation Functions
net=newff(minmax(p),[26,1],{'logsig','purelin'});

  Network Training
net.trainParam.epochs=100;
net.trainParam.goal=1e-25;
net.trainParam.lr=0.5;
erf(p)
net=train(net,p,t);

  Experiment Input Patterns with Other Models
results=sim(net,Z2(1:end)');
y=results;
results=round(y);
```

Test Result of Backpropagation Network on Command Window

```
results=

   1.9694

results =

    2
```

Letter B

Explanation

Formation of Letter Patterns

Using a matrix size of 7x9 with a value of -1 and 1 as a form of representation (binary).

Network Input / Network Input

Merging examples of pattern letters A, B and C.

A1 [1: end] means 1x63 matrix size, so if combined and transposed into a 63x3 matrix.

Desired Network Target

The target uses a binary representation (0-1).

t = [1 2 3] is a 1x3 matrix. This code is used as an introduction that:

1 = Letter A

2 = Letter B

3 = Letter C

PR = zeros (63.2) is a 63x2 matrix with elements valued at '-1 'based on the number of elements of each input pattern is 63 with binary representation (0-1) which means there are 2 pieces of element.

PR (:, 2) = 1 means that the PR matrix whose previous element is worth '0 'is changed to value 1' in column 2.

Use of Backpropagation Function

net = newff (minmax (p), [3,1], {'logsig', 'purelin'})

PR: Rx2 order matrix which states the minimum and maximum values of each input unit (there are R input units).

3: number of hidden neurons with 'logsig' activation fungso

1: number of output neurons with activation function 'purelin

logsig: unipolar sigmoid activation function

purelin: a linear function has an output value equal to its input value.

Network / Network Training

net.trainParam.epochs = 3

net.trainParam.epochs = 4: This network will be trained with maximum epoch 3.

net = net (train, p, t)

net (train, p, t): training function ('train') input pattern p to target on net network.

Result of Network Training

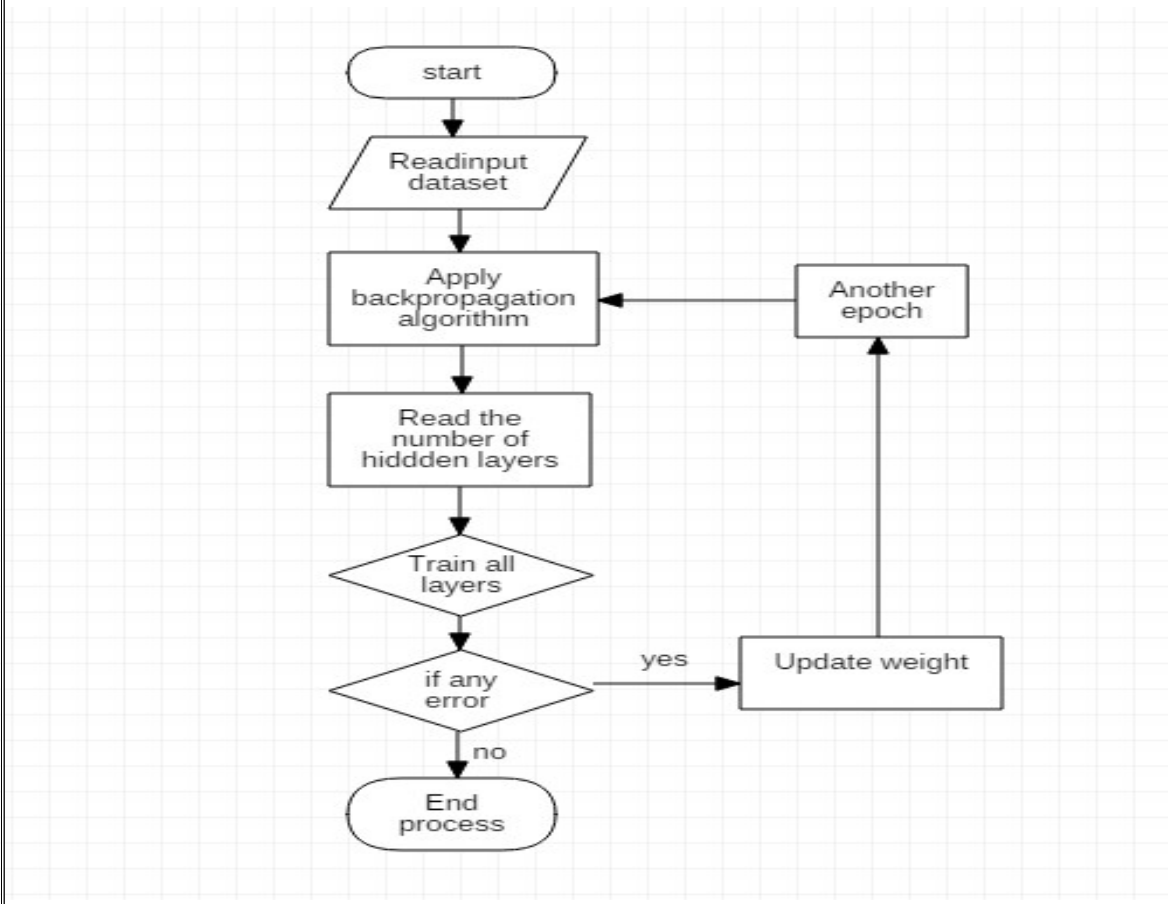
Iteration training is completed in 3 epochs. At epoch 1, performance = 0.675 (meaning there is only 1 pattern that is correctly recognized). At epoch 2, performance = 0.3 (meaning there are only 2 patterns that are correctly recognized). And at epoch 3, performance = 0 (meaning all patterns are correctly recognized).

Experiment Results

result = sim (net, B2 (1: end) ');

results = round (results);
sim: a function used to calculate network output to match the target. Target A is worth (1), target B is worth (2), and target C is worth (3). If the A2 (letter A with another model with 1x63 matrix size) pattern is attempted, the backpropagation network that has been formed produces a value of (1) meaning A2 pattern

FLOW CHART:-



SOURCE CODE:

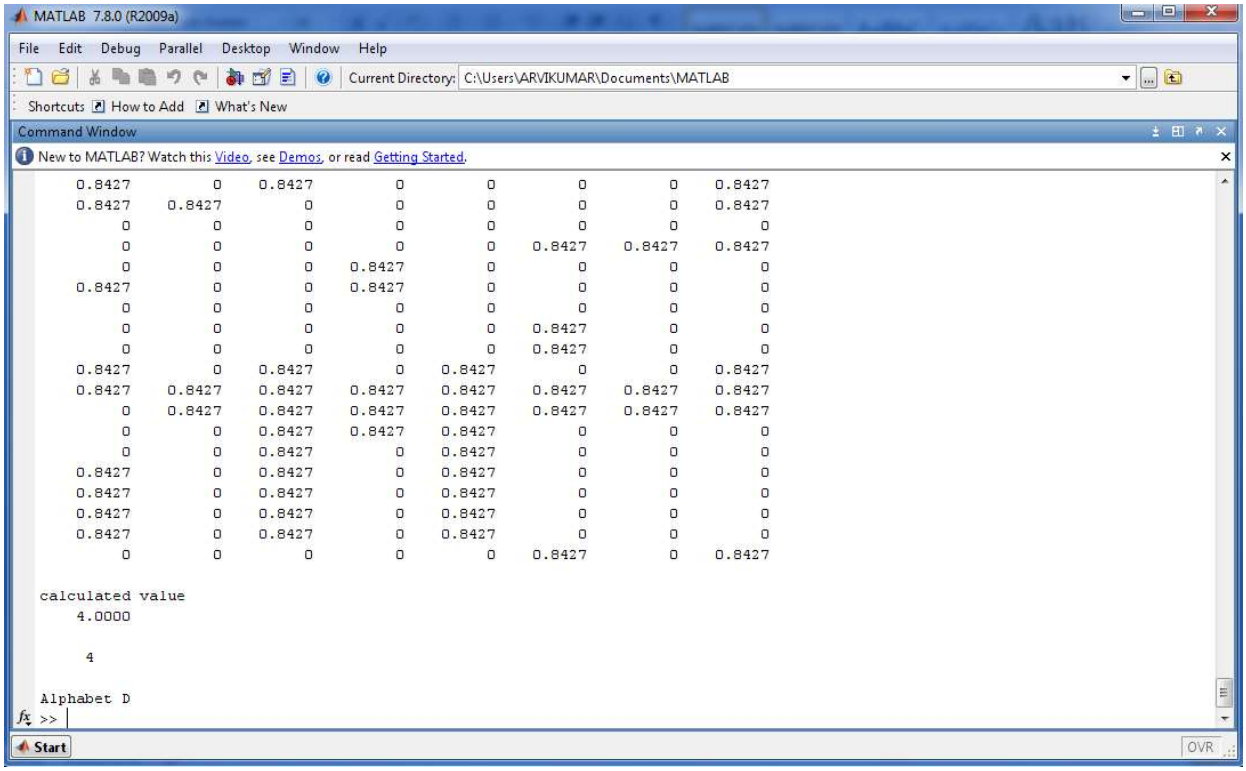
```
%Network Input
p=[A2(1:end);B2(1:end);C2(1:end);D2(1:end);E2(1:end);F2(1:end);G2(1:end);H2(1
:end);I2(1:end);J2(1:end);K2(1:end);L2(1:end);M2(1:end);N2(1:end);O2(1:end);P
2(1:end);Q2(1:end);R2(1:end);S2(1:end);T2(1:end);U2(1:end);V2(1:end);W2(1:end
);X2(1:end);Y2(1:end);Z2(1:end)]';

%4. Desired Target Network
t=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26];
PR=zeros(63,2);
PR(:,2)=1;

%5.Use of Backpropagation Functions
net=newff(minmax(p),[26,1],{'logsig','purelin'});

%6. Network Training
net.trainParam.epochs=100;
net.trainParam.lr=0.5;
erf(p)
net=train(net,p,t);

%7. Experiment Input Patterns with Other Models
results=sim(net,D2(1:end)');
y=results;
results=round(y);
disp('calculated value');
disp(y)
disp(results) if
results==1'
    disp('Alphabet A');
elseif results==2'
    disp('Alphabet B');
elseif results==3'
    disp('Alphabet C');
elseif results==4'
    disp('Alphabet D');
elseif results==5'
    disp('Alphabet E');
elseif results==6'
```

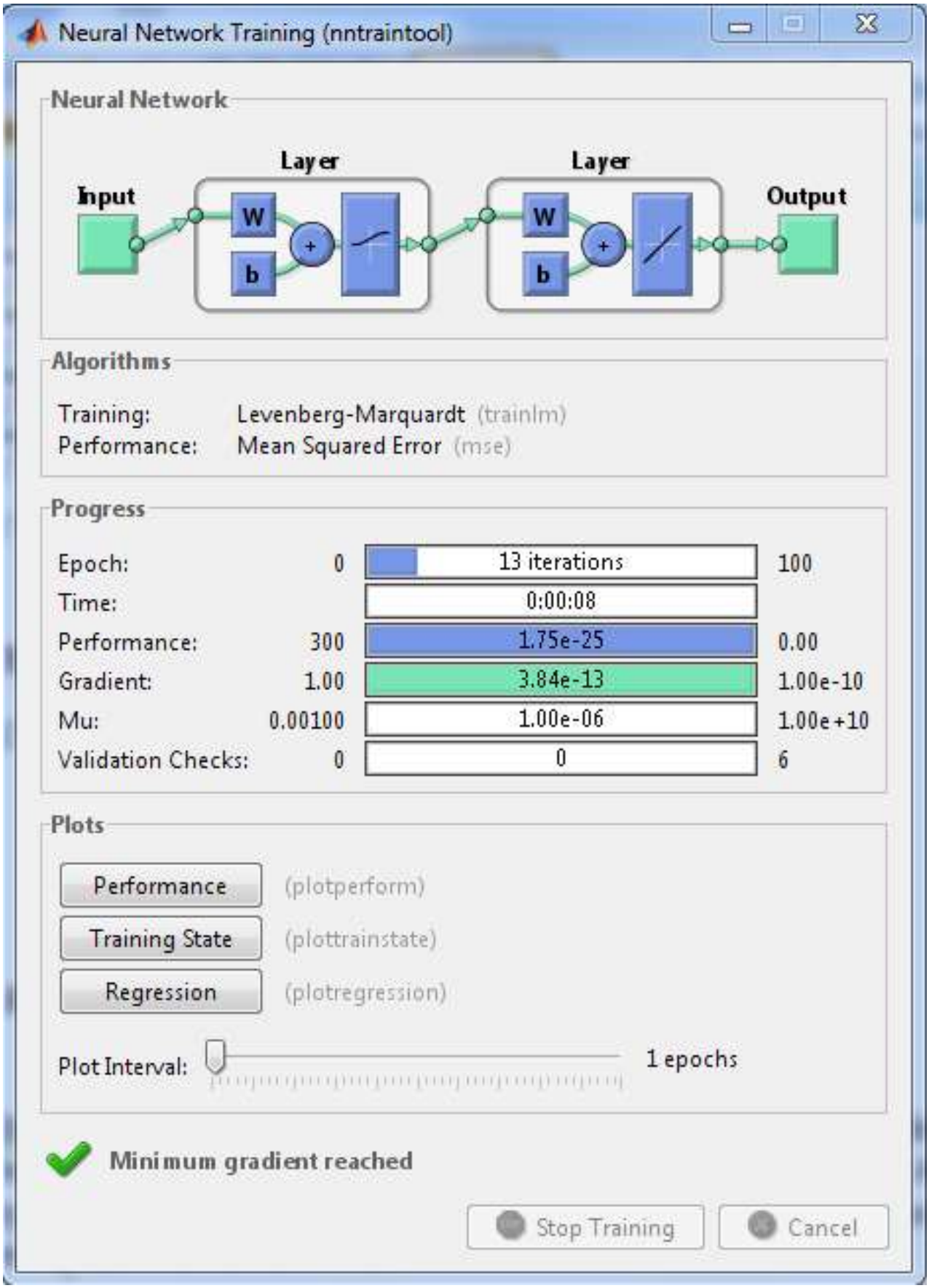
```
        disp('Alphabet    F');
elseif results==7'
        disp('Alphabet    G');
elseif results==8'
        disp('Alphabet    H');
elseif results==9'
        disp('Alphabet     I');
elseif results==10'
        disp('Alphabet     J');
elseif results==11'
        disp('Alphabet    K');
elseif results==12'
        disp('Alphabet    L');
elseif results==13'
        disp('Alphabet   M');
elseif results==14'
        disp('Alphabet    N');
elseif results==15'
        disp('Alphabet    O');
elseif results==16'
        disp('Alphabet    P');
elseif results==17'
        disp('Alphabet    Q');
elseif results==18'
        disp('Alphabet    R');
elseif results==19'
        disp('Alphabet    S');
elseif results==20'
        disp('Alphabet    T');
elseif results==21'
        disp('Alphabet    U');
elseif results==22'
        disp('Alphabet    V');
elseif results==23'
        disp('Alphabet   W');
elseif results==24'
        disp('Alphabet    X');
elseif results==25'
        disp('Alphabet    Y');
elseif results==26'
        disp('Alphabet    Z');
else
```

```
        disp('Letter not recognized')
end
```

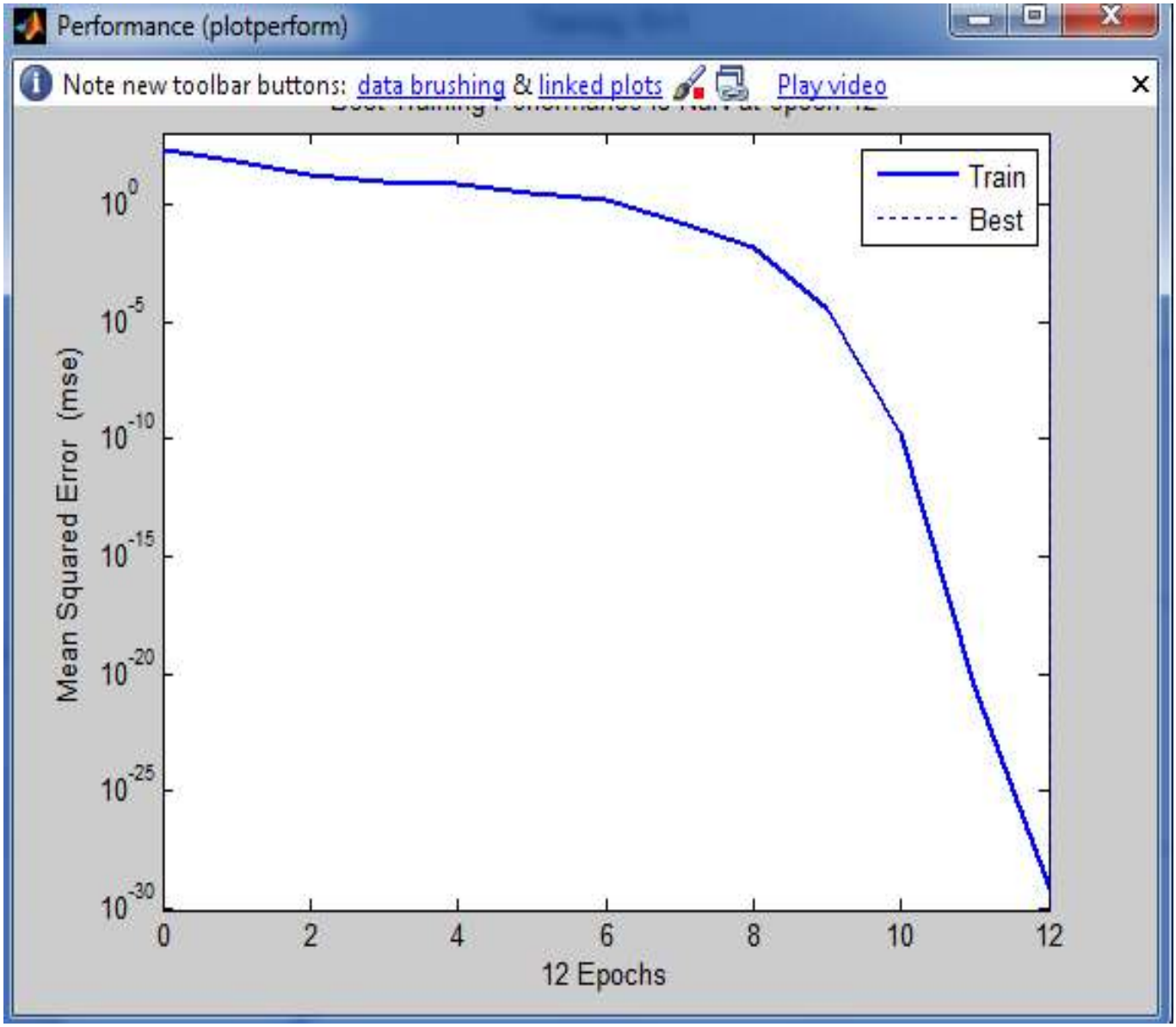
OUTPUT:

WITHOUT SETTING PERFORMANCE GOAL:

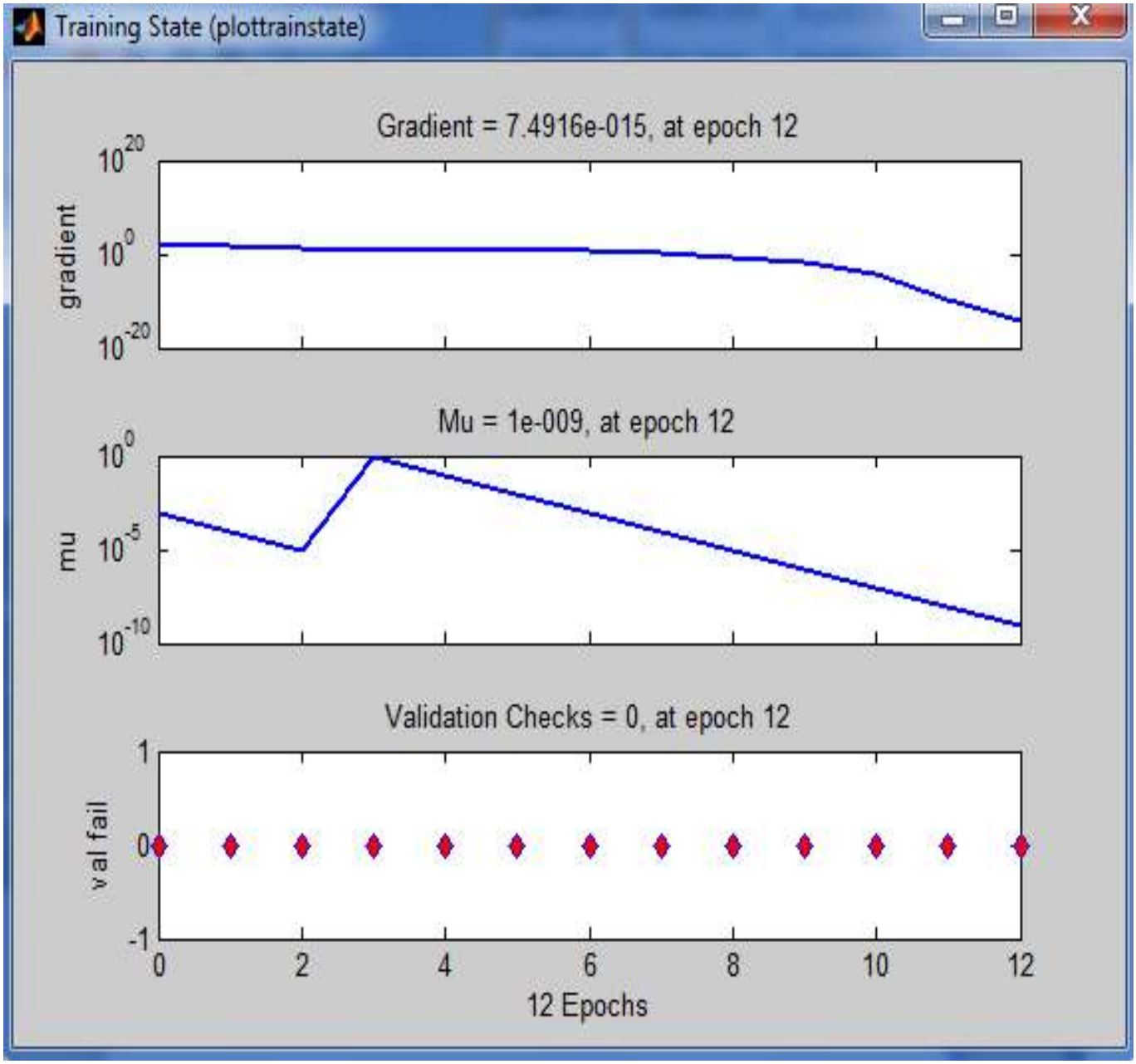



Neural Network Training (nntraintool)
Neural Network
Input
Layer
W
b
Layer
W
b
Output
Algorithms
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Progress
Epoch: 0 13 iterations 100
Time: 0:00:08
Performance: 300 1.75e-25 0.00
Gradient: 1.00 3.84e-13 1.00e-10
Mu: 0.00100 1.00e-06 1.00e+10
Validation Checks: 0 0 6
Plots
Performance (plotperform)
Training State (plottrainstate)
Regression (plotregression)
Plot Interval: 1 epochs
Minimum gradient reached
Stop Training
Cancel

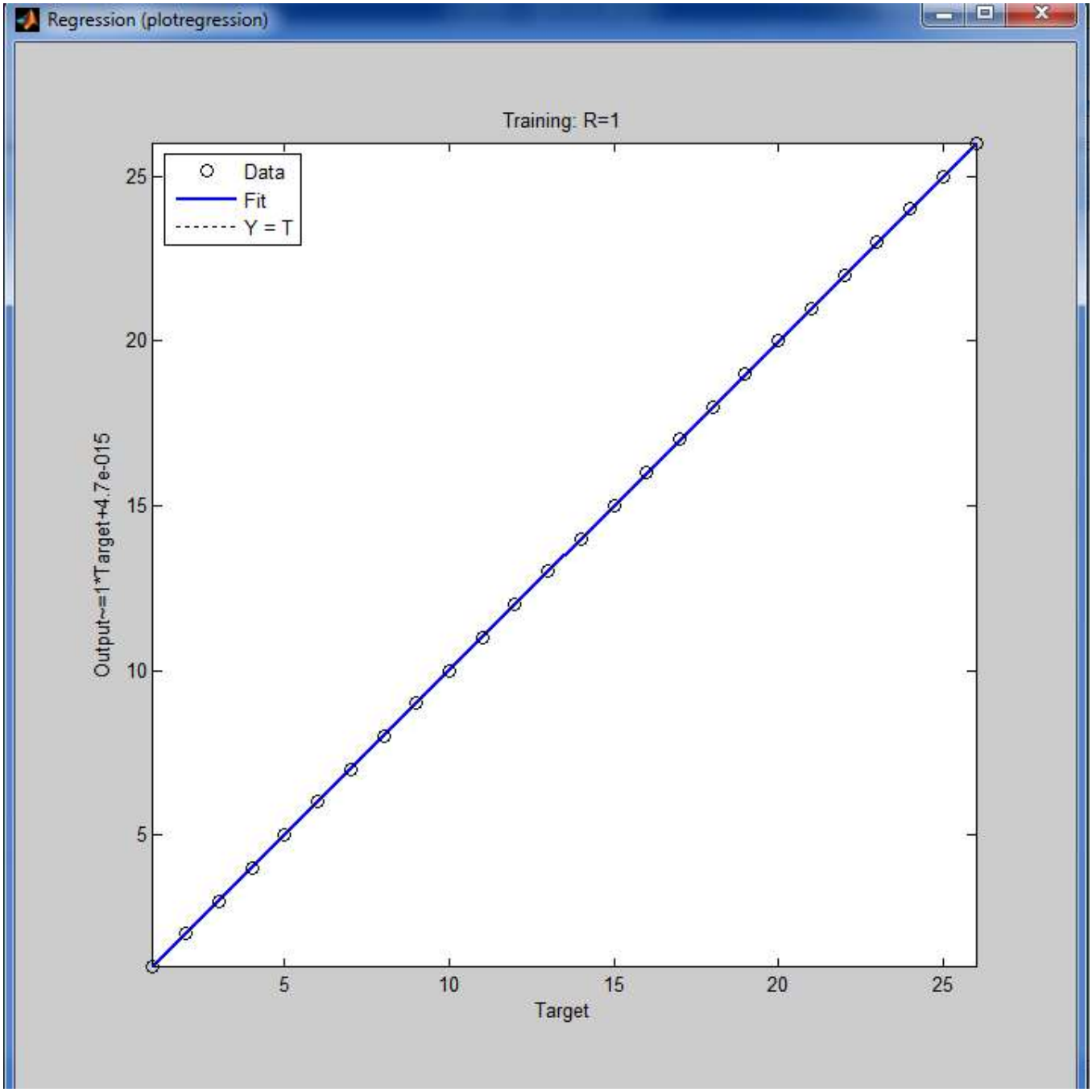
Performance



Training state:



Regression:


Regression (plotregression)
Training: R=1
Data
Fit
Y = T
Output~=1*Target+4.7e-015
Target
5
10
15
20
25

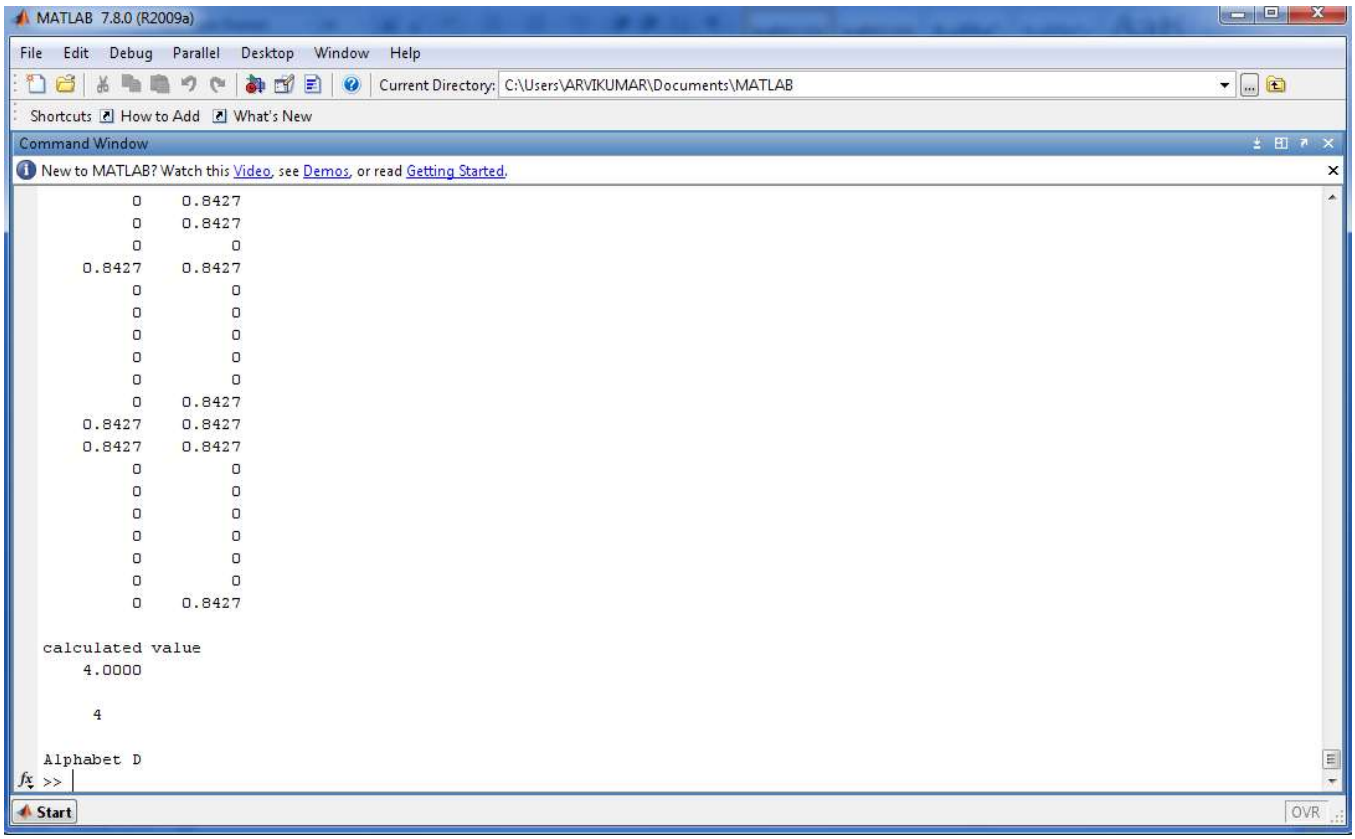
AFTER SETTING PERFORMANCE GOAL AS 1e-25

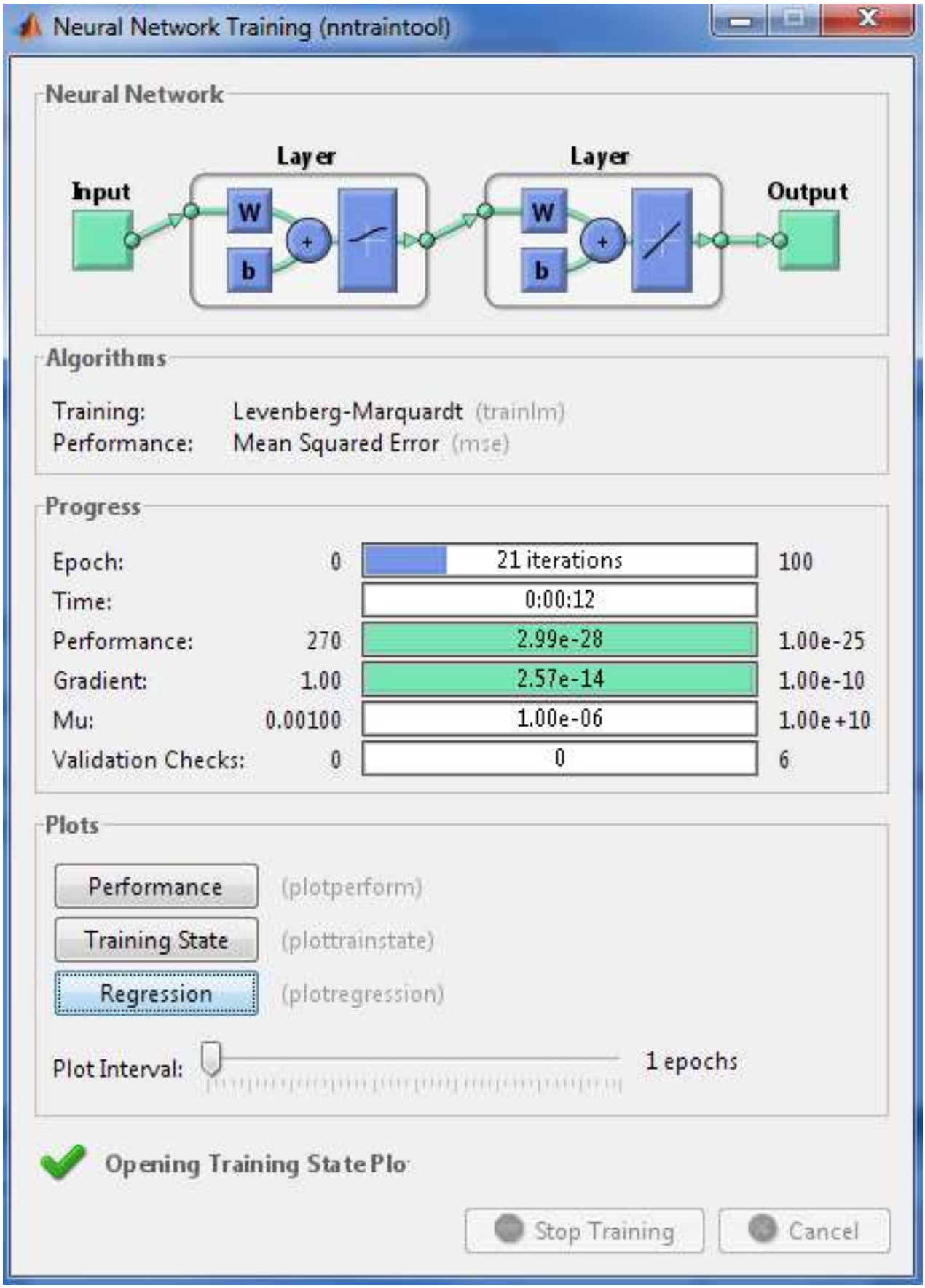
The maximum performance will be set to 1e-25

We use net.trainParam.goal=1e-25; this line to set the performance

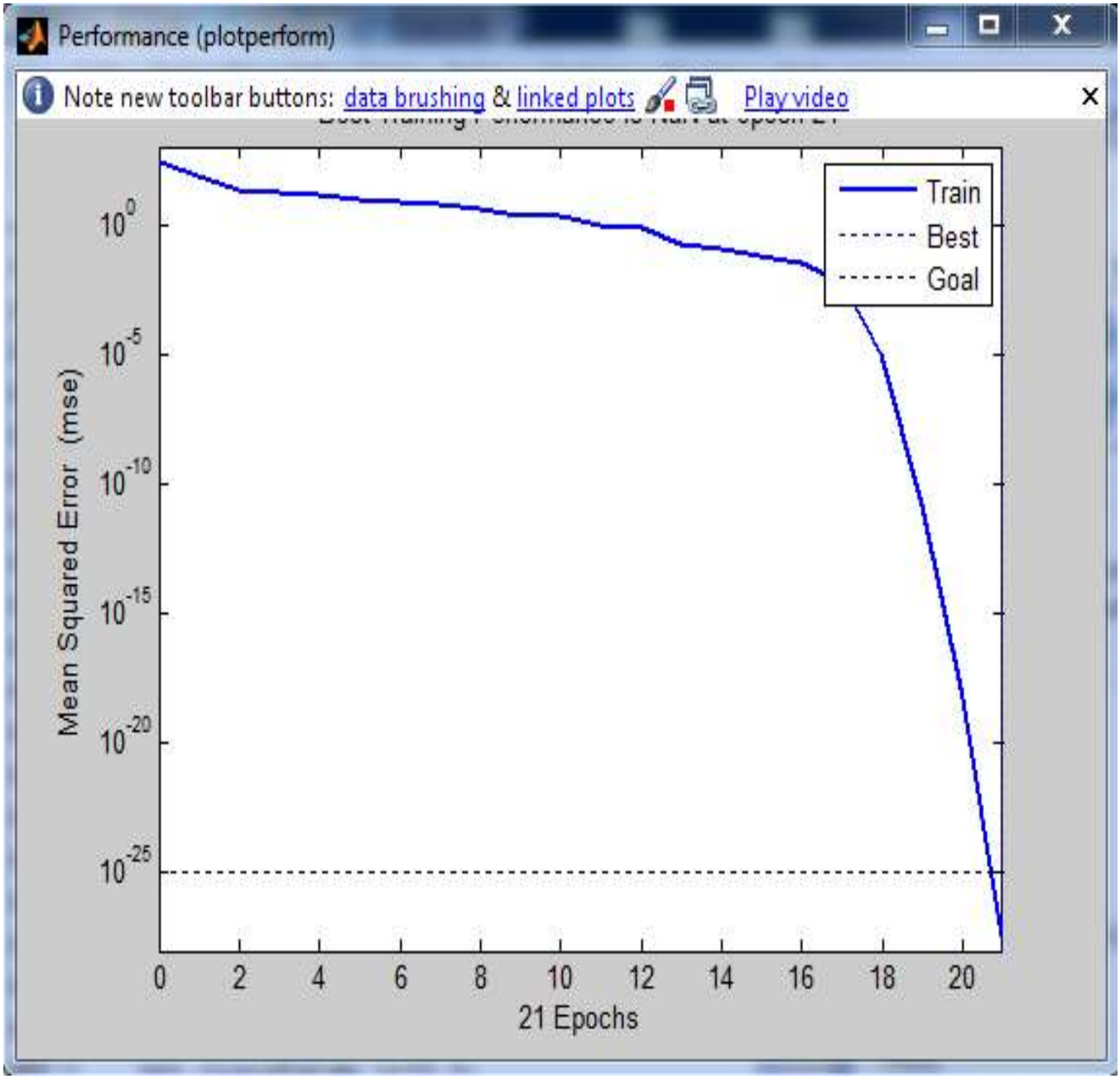
net.trainParam.goal=1e-25;

And the calculation and weight updation stopped when it reaches the maximum goal which is 1e-25

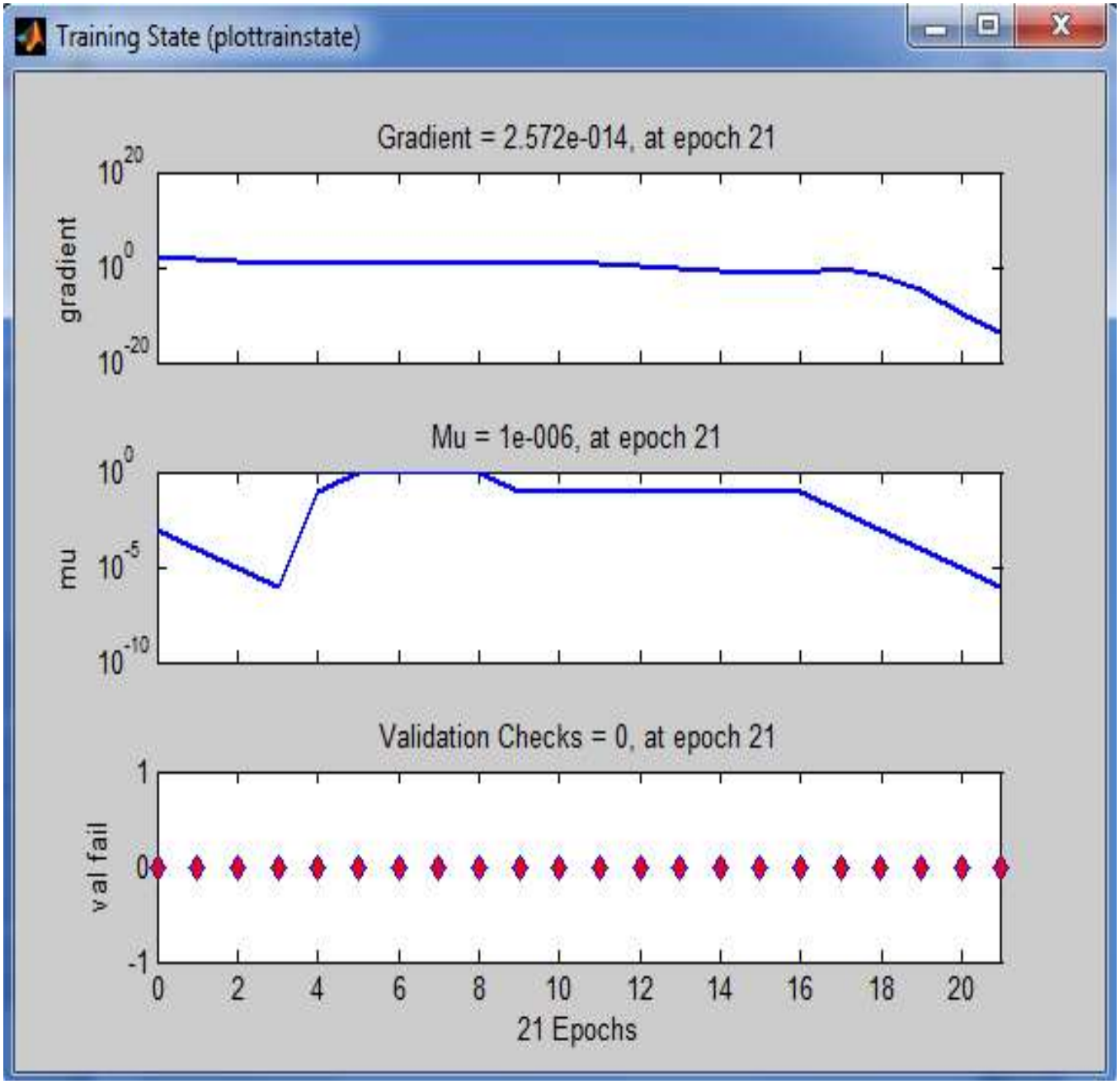



Neural Network Training (nntraintool)
Neural Network
Input
Layer
W
b
Layer
W
b
Output
Algorithms
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Progress
Epoch: 0 21 iterations 100
Time: 0:00:12
Performance: 270 2.99e-28 1.00e-25
Gradient: 1.00 2.57e-14 1.00e-10
Mu: 0.00100 1.00e-06 1.00e+10
Validation Checks: 0 0 6
Plots
Performance (plotperform)
Training State (plottrainstate)
Regression (plotregression)
Plot Interval: 1 epochs
Opening Training State Plo
Stop Training
Cancel

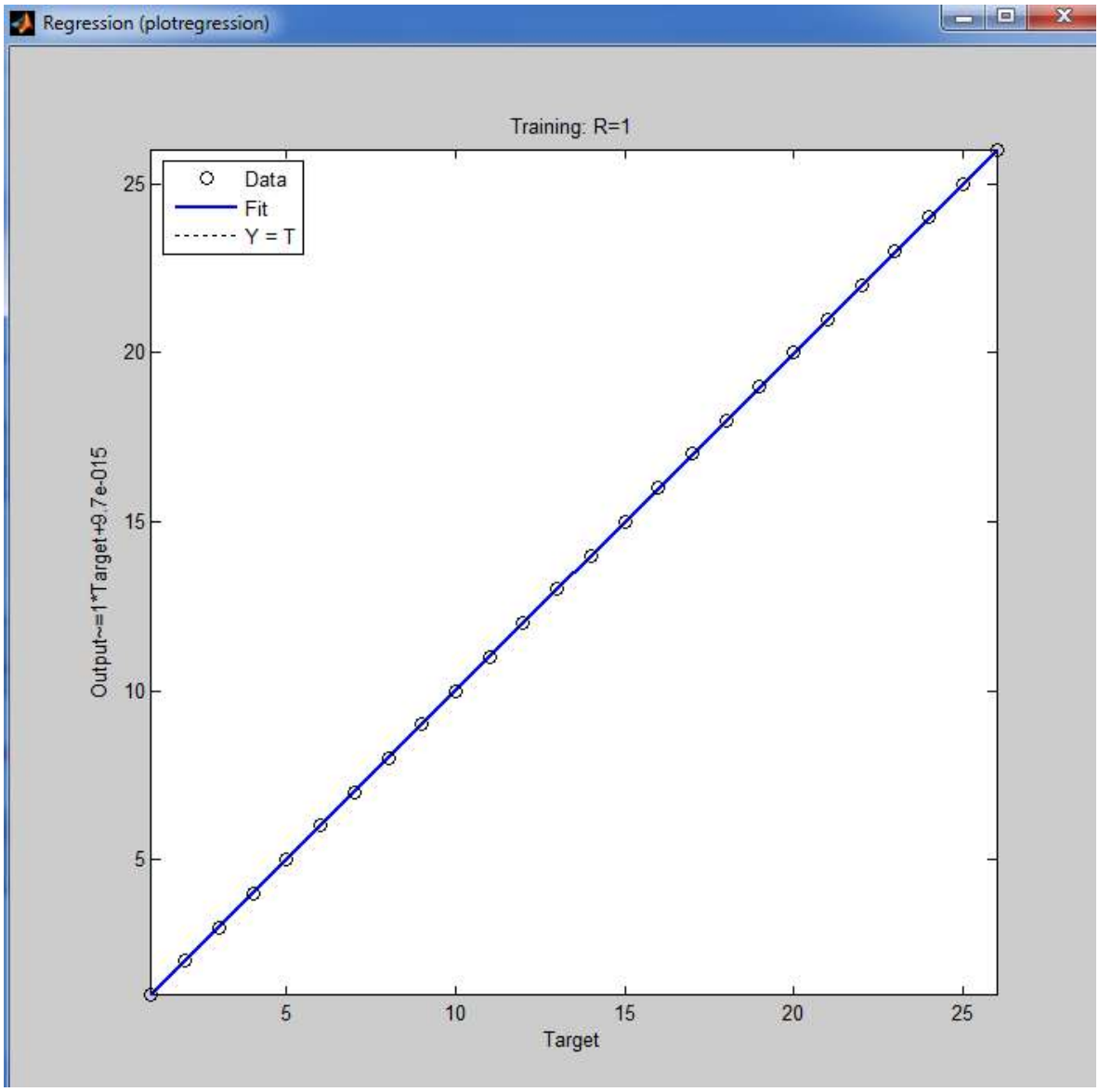
Performance:



Training state:



Regression:



CONCLUSION:-

The results of structure analysis shows that ifthe number of hidden nodes increases the number of epochs taken to recognize the character isalso increases. As the hidden layer and epochs increases the accuracy of the result increases so we get the recognized alphabet from the result.

REFERENCES:-

[1]. Verma B.K,\Handwritten Hindi Character Recognition Using Multilayer Perceptronand Radial Basis
Function Neural Network",IEEE International Conference on Neural Network, 4, pp. 2111-2115, 1995.
[2]. Sutha .J, Ramraj.N,\Neural Network Based Online Tamil Handwritten Character Recognition Syste, IEEE International Conference on Computational Intelligence and Multimedia Application, 2007, 2, 13-15, Dec.2007, Page(s): 446-450, 2007.
[3]. Yuelong Li Jinping Li Li Meng,\Character Recognition Based on Hierarchical RBF Neural Networks" I ntelligent Systems Design and Applications, 2006. ISDA '06. Sixth International Conference, 1, On Page(s): 127-132, 2006.
[4]. Y.Y. Chung, and M.T.Wong,\Handwritten Character Recognition by Fourier De-scriptors and Neural
[5]. Network", Proceeding of IEEE TENCON - Speech and Image Technologies for C.
[6]. Gonzalez, Rafael C., and Richard E.Woods. Digital Image Processing. Addison-Wesley, 1992. p. 518.