



RPS
GROUP OF INSTITUTIONS

Balana, Satnali Road,
Mohindergarh, Haryana 123029

Ph.: 91-1285-241431
Mob.: 09466275566, 09416150201

E-mail: info@rpsinstitutions.org
Web : www.rpsinstitutions.org

Approved by **AICTE** (Govt. of India) & Affiliated to **M.D. University**, Rohtak

RPS GROUP OF INSTITUTIONS, BALANA MOHINDERGARH

LAB MANNAUL

NEURAL NETWORK MATLAB

CSE 7TH SEM

Department of Computer Science and Engineering

2015-16

List of Experiments

1. How the weights and bias values affect the output of a neuron.
2. How the choice of activation function (or transfer function) affects the output of a neuron. Experiment with the following functions: identity (purelin), binary threshold (hardlim, hardlims) and sigmoid (log sig, tan sig).
3. How the weights and bias values are able to represent a decision boundary in the feature space.
4. How this decision boundary changes during training with the perceptron learning rule.
5. How the perceptron learning rule works for linearly separable problems.
6. How the perceptron learning rule works for non-linearly separable problems.
7. Write a program to construct matrices.
8. Write a program to construct the For, While and If constructs.
9. Write a program to classify problem with a multilayer perceptron.
10. Write a program to construct an AdA line for time series and target signal.
11. Write a program to plot a graph in 2-D.
12. Write a program to plot the sine and cosine wave.
13. Write a program to save data in file data1 mat dat2.mat.
14. Write a program to plot a graph in 3-D.
15. Write a program to classify problem with a single layer perceptron.

PROGRAM -1

Aim: Write a program for introduction about MatLab.

Introduction:

What is MATLAB ?

MATLAB started as an interactive program for doing matrix calculations and has now grown to a high level mathematical language that can solve integrals and differential equations numerically and plot a wide variety of two and three dimensional graphs. In this subject you will mostly use it interactively and also create MATLAB scripts that carry out a sequence of commands. MATLAB also contains a programming language that is rather like Pascal.

The first version of Matlab was produced in the mid 1970s as a teaching tool. The vastly expanded Matlab is now used for mathematical calculations and simulation in companies and government labs ranging from aerospace, car design, signal analysis through to instrument control & financial analysis. Other similar programs are Mathematica and Maple. Mathematica is somewhat better at symbolic calculations but is slower at large numerical calculations.

Recent versions of Matlab also include much of the Maple symbolic calculation program.

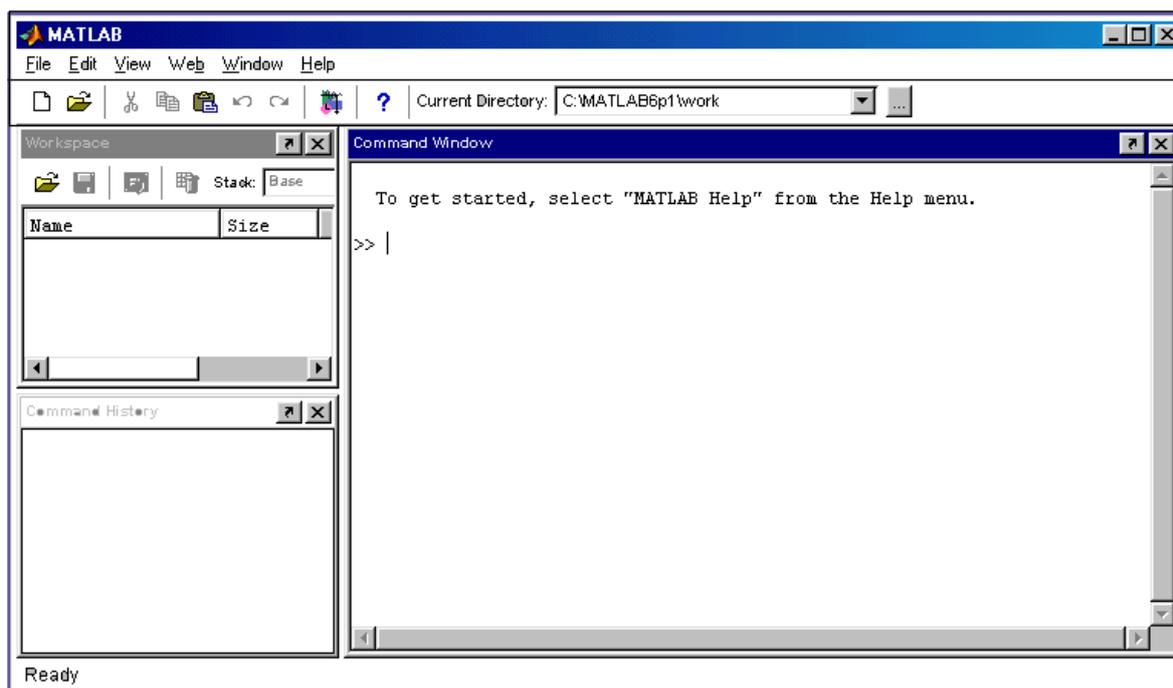
How to get started:

When the computer has started go through the following steps in the different menus

- Look for the Network Applications folder and double click on that
- Within this you will see a little icon for Matlab61 – double click on that

Within about 30 seconds Matlab will open (you may get a little message box about ICA applications – just click OK on this)

You should now see a screen like the picture below



This is the Matlab screen – it broken into 3 parts

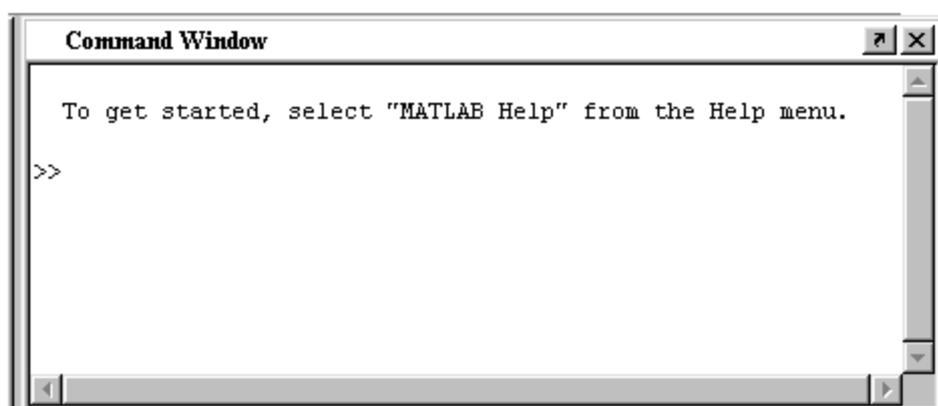
On the right you have the **Command Window** – this is where you type commands and usually the answers (or error messages) appear here too

On the top left you have the **Workspace** window – if you define new quantities (called variables) there names should be listed here.

On the bottom left you have Command History window – this is where past commands are remembered. If you want to re-run a previous command or to edit it you can drag it from this window to the command window to re-run it.

The Matlab prompt is >>.

Look at the **Command Window** and you will see the cursor flickerring after the >> prompt. This means that Matlab is waiting for further instructions.



Seeing the Matlab demo.

Just type the command

>> **demo** (and then press **Enter** or **Return** key)

and run the simple slideshow under the **Matrix manipulation** option.

This briefly shows you some of the things that Matlab can do - don't worry if you do not know what everything means here.

Matlab works to 15 significant figures but usually only shows 5 Figures:

Type **pi** and you will normally get 3.1416

You can change to the **long format** in which all 15 figures are displayed

```
>> pi .....
```

```
>> format long
```

```
>> pi .....
```

An easier and more flexible format is scientific notation with 5 significant figures

```
>> format short e
```

```
>> pi .....
```

Mostly we work in the short format.

To go back to short format

```
>> format short .....
```

Very large or small numbers use exponential format - e.g. $6.00e+23 = 6 \times 10^{23}$

Variable names:

Matlab distinguishes between lower case and capital letters - so **A** and **a** are different objects

for Matlab. The normal convention to use lower case names except for global variables.

Only

the first 19 characters in a variable name are important.

Matlab has many built in functions:

Among them are:

Trig functions sin, cos, tan, sec, cosec and cotan

(sin(x) etc assume that x is in radians)

Are the following what you expect ?

```
>> sin(pi/2) .....
```

```
>> cos(pi/2) .....
```

Remember that pi is a built-in constant

Inverse trig functions asin, acos, atan

(these are abbreviations for arcsin, arccos and arctan - the answers are returned in radians so that $\text{asin}(1) = \text{pi}/2$)

Exponential function exp

Logarithm functions log is log to base e, while

log10 and log2 are logs to bases 10 or 2

Hyperbolic functions cosh, sinh, and tanh

Matlab also has many other more sophisticated functions for solving linear equations, getting eigenvalues of matrices, solving differential equations or calculating integrals numerically.

Assorted tips and tricks:

- You can use some Unix commands at the MATLAB prompt, such as “ls” and “cd”.
- You can use the “Tab” key to complete commands.
- You can use the up arrow to see previous commands you’ve typed in. Just the up arrow alone will go through previous commands in reverse order. If you type an “s”, for example, and then the up arrow, you’ll see previous commands that started with “s”.
- You can access the Unix command line inside MATLAB by preceding commands with a “!”. This is a good way to execute Unix scripts or simple Fortran/C codes from within MATLAB, especially if you are transferring data between MATLAB and external codes.
- You can use “eval” to execute a string as a MATLAB expression. This can be very useful in m-files, such as when the filenames are variables.
- If you ever find yourself questioning the meaning of things, type “why”.

Getting help

MATLAB has an interactive “Help Desk” which includes documentation and examples for pretty much everything you’d want to do. However, it can be slow to navigate, so if you just want to find out how to use a particular function, it’s easier to use the “help” command. Useful commands for accessing MATLAB help are:

helpdesk – starts the MATLAB interactive help browser

help name – displays documentation for function/command ‘name.’

lookfor string – searches all help files for ‘string’ in the description (first comment line), and displays the function names with the description.

Suppose you want to find out how to use the tangent function, tan(x).

```
>> help tan
```

```
TAN Tangent of argument in radians.
```

```
TAN(X) is the tangent of the elements of X.
```

PROGRAM -2

Aim: Write a program to perform operations on two matrix using MatLab.

Program:

INPUT:

```
a=[3 5 7  
4 6 8  
5 8 9]
```

a =

```
3 5 7  
4 6 8  
5 8 9
```

```
b=[5 7 9  
3 4 7  
3 6 9]
```

b =

```
5 7 9  
3 4 7  
3 6 9
```

OUTPUT:

a+b

ans =

8	12	16
7	10	15
8	14	18

a-b

ans =

-2	-2	-2
1	2	1
2	2	0

a*b

ans =

51	83	125
62	100	150
76	121	182

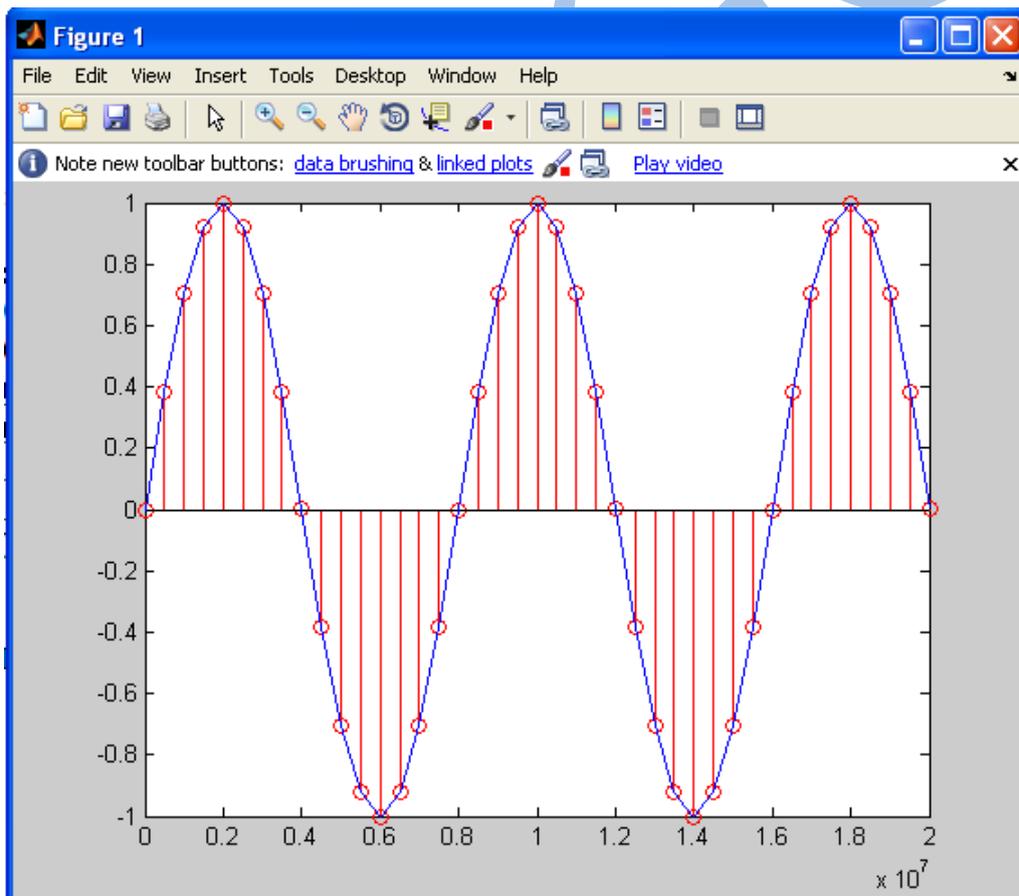
RRPSGOI

PROGRAM - 3

Aim: Write a program to create sin wave using MatLab.

```
t=[0:1:40];  
f=500;  
fs=8000;  
x=sin(2*pi*f/fs*t);  
figure(1);  
stem(t*f*1000,x,'r');  
hold on;  
plot(t*f*1000,x);
```

OUTPUT:



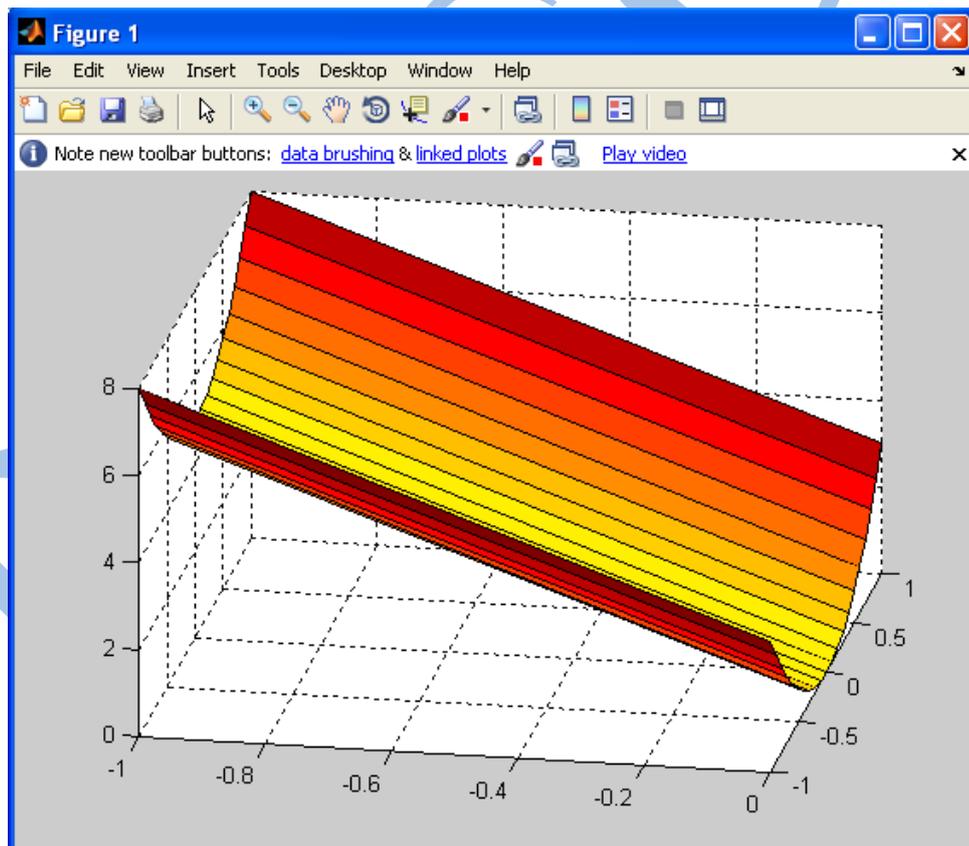
PROGRAM - 4

Aim: write a program for 2D graph plotting using MatLab.

Program:

```
[x,y]=meshgrid(-1:0.1:-1,1:0.1:1);  
z=5*x.^2+3*y.^2;  
contour(x,y,z)  
prism  
mesh(x,y,z)  
surf(x,y,z)  
view([20,10])  
view([10,30])
```

OUTPUT :



PROGRAM – 5

Aim: Write a program for 3D Plotting using MatLab.

Program:

x =

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

y =

1 1 1 1 1

2 2 2 2 2

3 3 3 3 3

4 4 4 4 4

5 5 5 5 5

z =

2 5 6 7 6

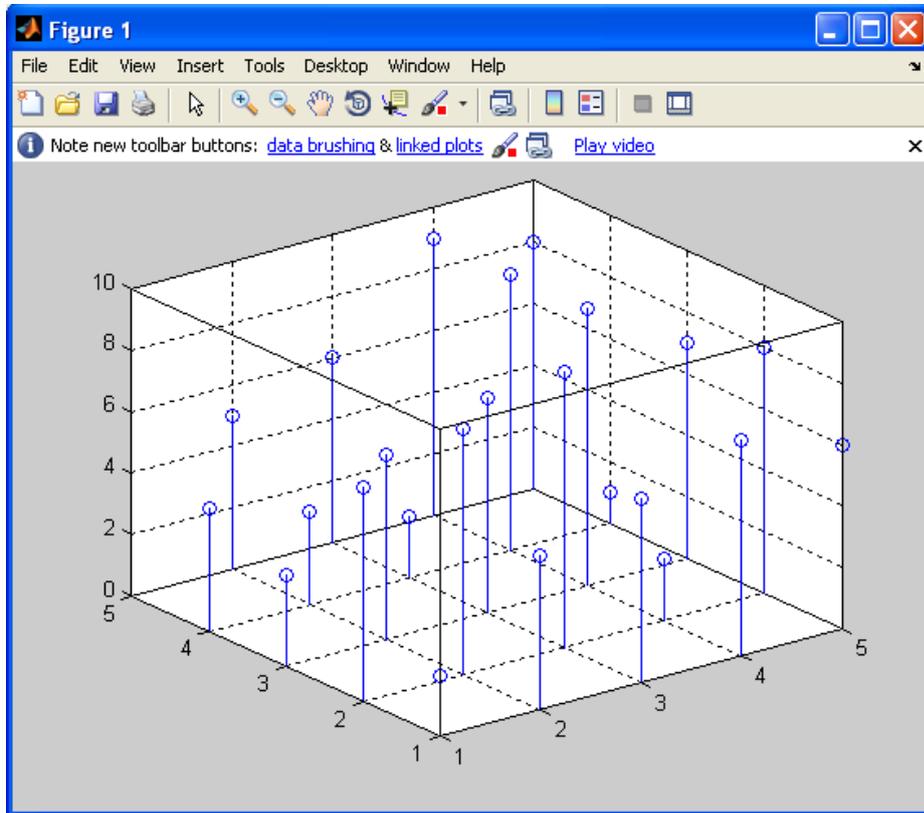
7 8 9 2 8

3 6 7 9 7

4 3 2 9 1

10 5 6 9 8

stem3(x,y,z)

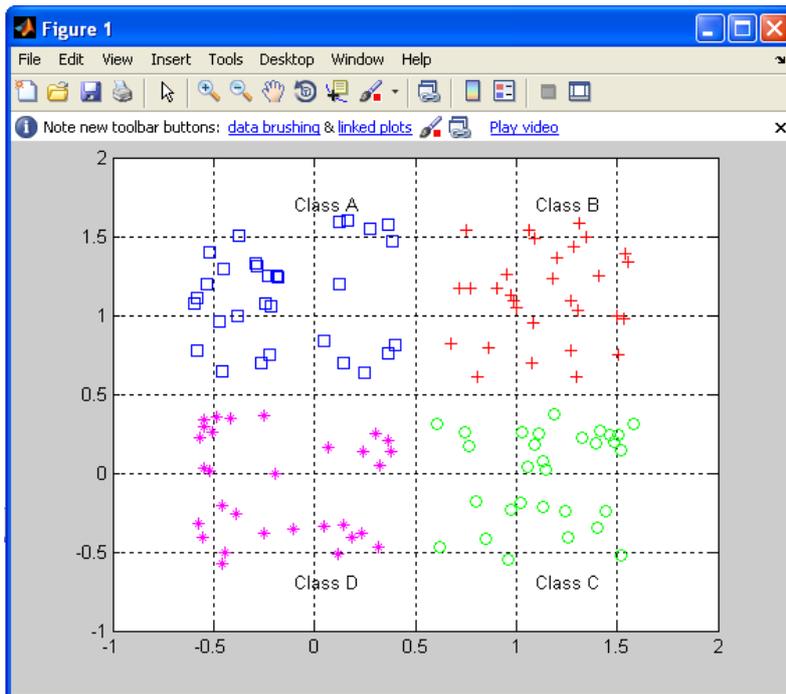
OUTPUT:

PROGRAM – 6

Aim: write a program for classification of a 4-class problem with a perceptron using MatLab.

Program:

```
close all, clear all, clc, format compact
K = 30;
q = .6;
A = [rand(1,K)-q; rand(1,K)+q];
B = [rand(1,K)+q; rand(1,K)+q];
C = [rand(1,K)+q; rand(1,K)-q];
D = [rand(1,K)-q; rand(1,K)-q];
plot(A(1,:),A(2:,:), 'bs')
hold on
grid on
plot(B(1,:),B(2:,:), 'r+')
plot(C(1,:),C(2:,:), 'go')
plot(D(1,:),D(2:,:), 'm*')
text(.5-q,.5+2*q,'Class A')
text(.5+q,.5+2*q,'Class B')
text(.5+q,.5-2*q,'Class C')
text(.5-q,.5-2*q,'Class D')
a = [0 1]';
b = [1 1]';
c = [1 0]';
d = [0 0]';
```

OUTPUT:

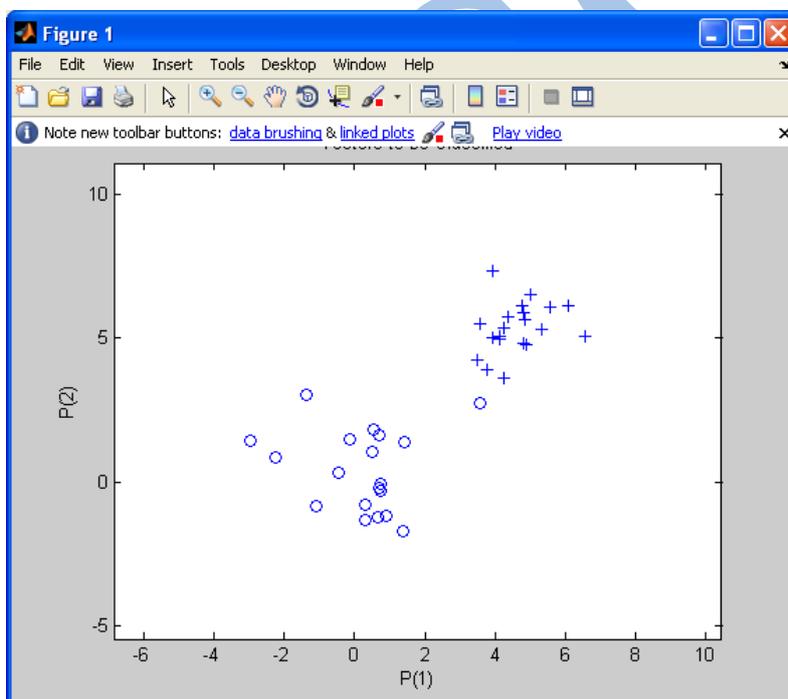
PROGRAM – 7

Aim: Write a program for classification of linearly separable data with a perceptron using MatLab.

PROGRAM:

```
N=20;  
offset=5;  
x=[randn(2,N) randn(2,N)+offset];  
y=[zeros(1,N) ones(1,N)];  
figure(1)  
plotpv(x,y);
```

OUTPUT:



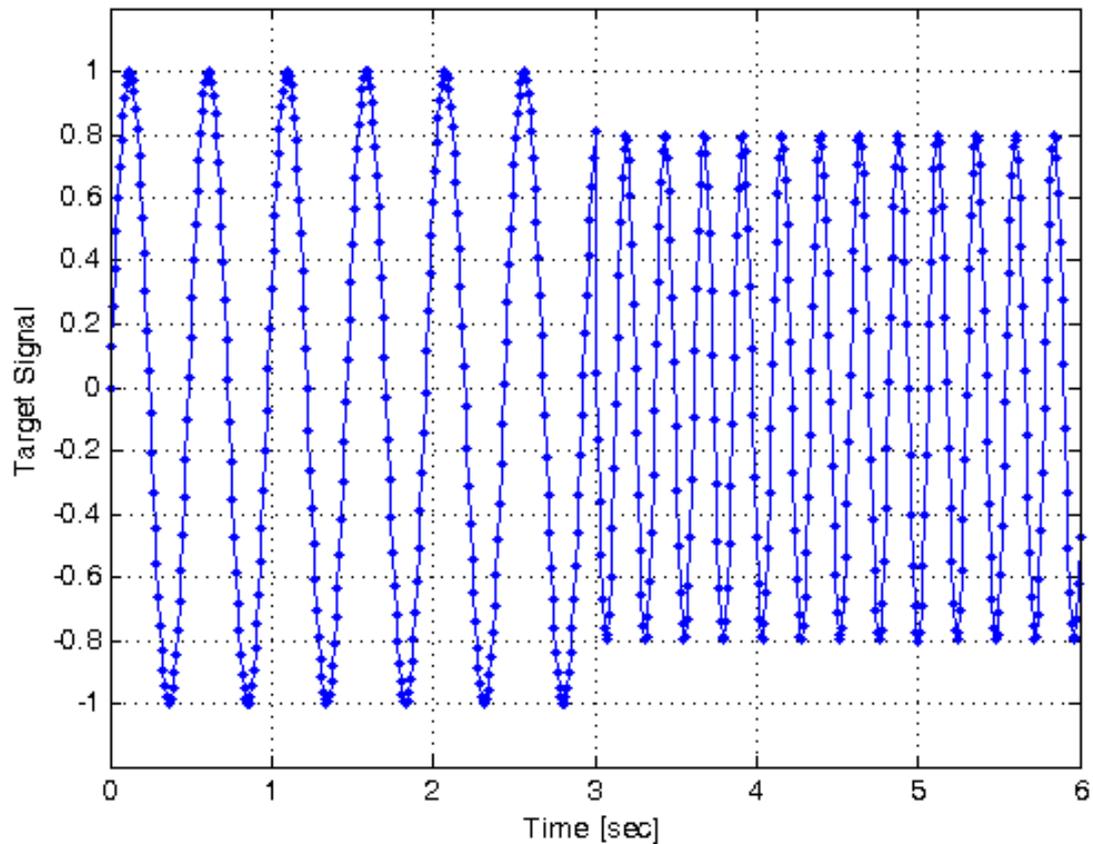
PROGRAM - 8

Aim: Write a program to define ADALINE time prediction using MatLab.

PROGRAM:

```
close all, clear all, clc, format compact
dt = 0.01;
t1 = 0 : dt : 3;
t2 = 3+dt : dt : 6;
t = [t1 t2];
y = [sin(4.1*pi*t1) .8*sin(8.3*pi*t2)];
plot(t,y,'.-')
xlabel('Time [sec]');
ylabel('Target Signal');
grid on
ylim([-1.2 1.2])
```

OUTPUT:



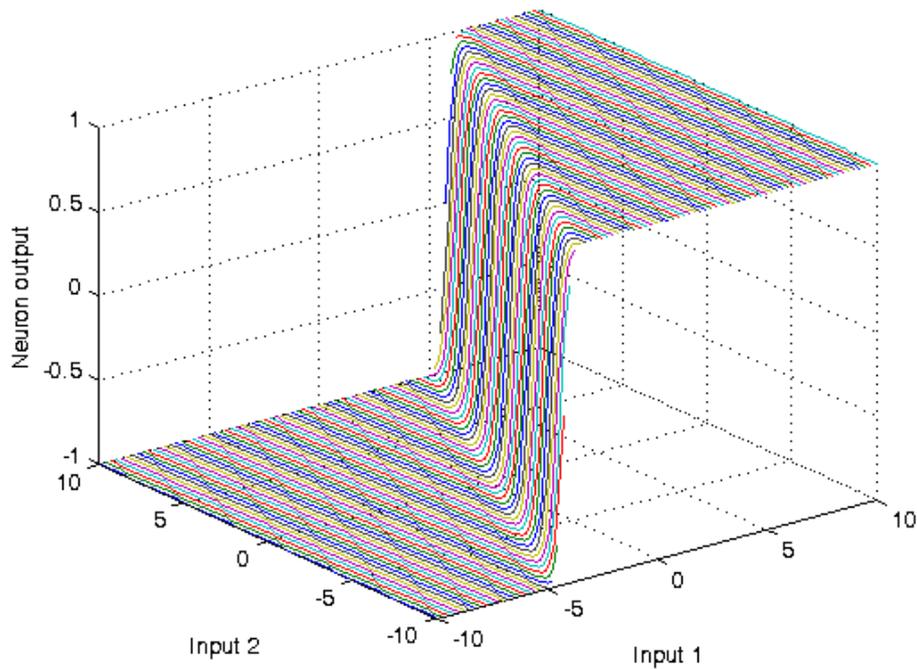
PROGRAM - 9

Aim: Write a program for neuron output using MatLab.

PROGRAM:

```
w = [4 -2]
b = -3
func = 'tansig'
p = [2 3]
activation_potential = p*w'+b
neuron_output = feval(func, activation_potential)
[p1,p2] = meshgrid(-10:.25:10);
z = feval(func, [p1(:) p2(:)]*w'+b );
z = reshape(z,length(p1),length(p2));
plot3(p1,p2,z)
grid on
xlabel('Input 1')
ylabel('Input 2')
zlabel('Neuron output')
```

OUTPUT:



PROGRAM - 10

Aim: Write a Program to plot the Function of a Complex Variable in 4D effect using MatLab.

PROGRAM:

```
r = (0:0.025:1)';
theta = pi*(-1:0.05:1);
z = r*exp(1i*theta);
w = z.^3;
surf(real(z),imag(z),real(w),imag(w))
xlabel('Real(z)')
ylabel('Img(z)')
zlabel('Real(w)')
cb = colorbar;
cb.Label.String = 'Imag(w)';
```

OUTPUT:

