Perceptron algorithm for AND Gate

```
clc;
clear all;
close all;

x1=[1 -1 1 -1];
x2=[1 1 -1 -1];
t=[1 -1 -1 -1];
w1=0;
w2=0;
b=0;
a=1;
th=0;
x3=[-4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9];

for i=1:4
    yin(i)=b+((x1(i)*w1)+(x2(i)*w2));
    if(yin(i)>th)
        y(i)=1;
    else
        if(yin(i)<th)
            y(i)=-1;
        else
            y(i)=0;
        end
    end
    if(y(i)~=t(i))
        w1=w1+(a*t(i)*x1(i));
        w2=w2+(a*t(i)*x2(i));
        b=b+(a*t(i));
    else
        w1=w1;
        w2=w2;
    end
end

disp('w1=');
disp(w1);
disp('w2=');
disp(w2);
disp('b=');
disp(b);

scatter(x1,x2);
hold on;

y1=-(w1/w2)*(x3-0.5);
axis([-2 2 -2 2]);
plot(x3,y1);
```

OUTPUT:

```
w1=    1

w2=    1

b=   -1
```

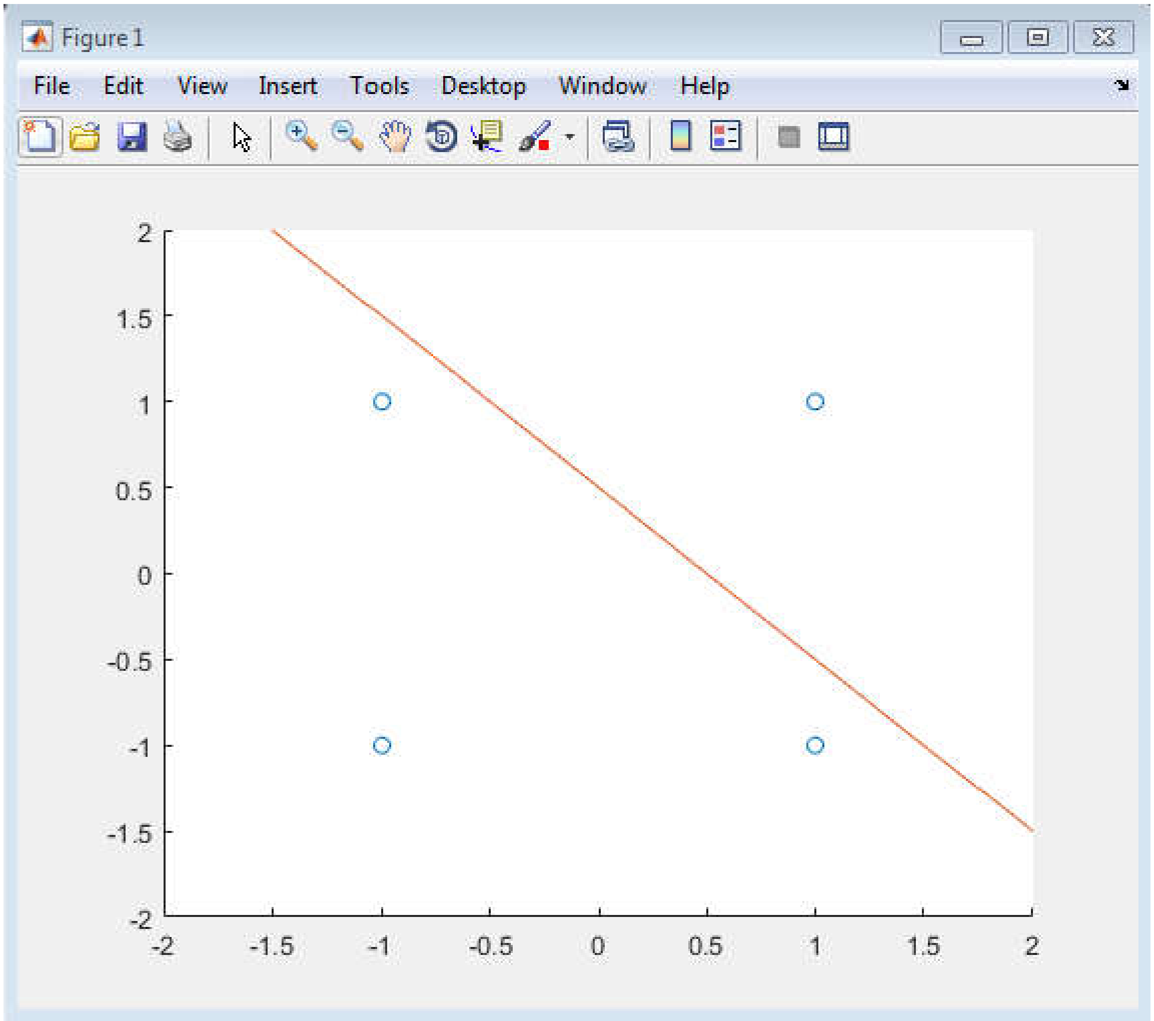

Fig 1: Output of Perceptron algorithm for AND gate

Perceptron algorithm for OR Gate

```
clc;
clear all;
close all;

x1=[1 -1 1 -1];
x2=[1 1 -1 -1];
t=[1 1 1 -1];
w1=0;
w2=0;
b=0;
a=1;
th=0;
x3=[-4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9];

for i=1:4
    yin(i)=b+((x1(i)*w1)+(x2(i)*w2));
    if(yin(i)>th)
        y(i)=1;
    else
        if(yin(i)<th)
            y(i)=-1;
        else
            y(i)=0;
        end
    end
    if(y(i)~=t(i))
        w1=w1+(a*t(i)*x1(i));
        w2=w2+(a*t(i)*x2(i));
        b=b+(a*t(i));
    else
        w1=w1;
        w2=w2;
    end
end
disp('w1=');
disp(w1);
disp('w2=');
disp(w2);
disp('b=');
disp(b);
scatter(x1,x2);
hold on;

y1=-(w1/w2)*(x3-(-1));
axis([-2 2 -2 2]);
plot(x3,y1);
```

OUTPUT:

```
w1=1

w2=1

b=1
```

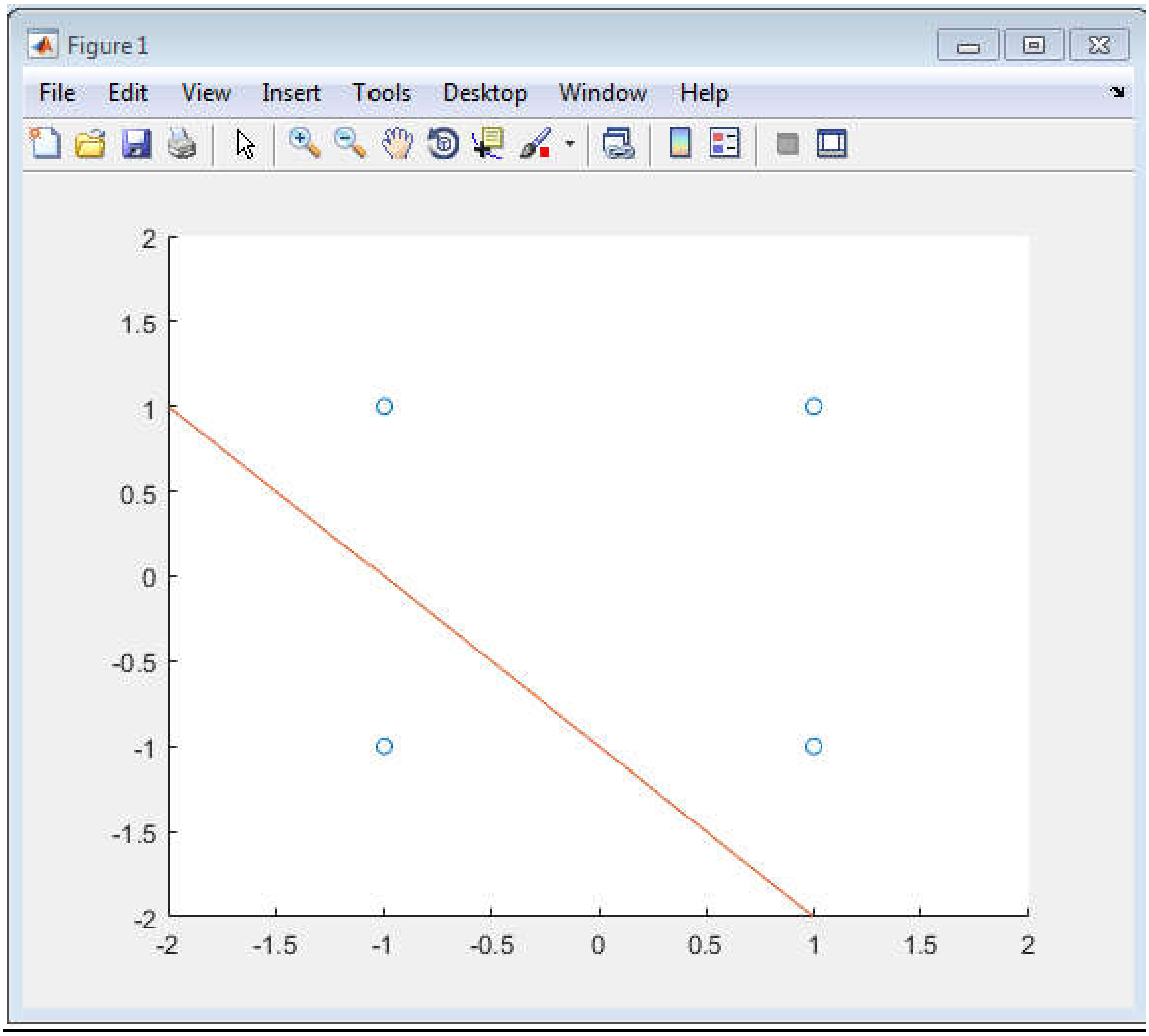

Fig 2: Output of Perceptron algorithm for OR gate