```
clc; clear;
%%
% Name: Devosmita Chatterjee
% Assignment2

x = csvread('input_data_numeric.csv',0,1);

target = [1, -1, 1, 1, 1, -1, 1, 1, 1, 1, 1, 1, -1, 1, 1, 1];%ls
%target = [-1, -1, -1, -1, -1, -1, -1, -1, 1, -1, -1, -1, -1, -1, 1, 1];%ls
%target = [1, 1, 1, -1, 1, 1, 1, 1, -1, -1, -1, 1, 1, 1, -1, 1];%ls
%target = [1, 1, 1, 1, -1, -1, -1, -1, -1, 1, -1, -1, 1, -1, -1, -1];%nls
%target = [-1, -1, -1, 1, -1, -1, -1, 1, 1, 1, -1, 1, -1, -1, -1, 1];%nls
%target = [-1, -1, 1, -1, 1, 1, 1, -1, -1, 1, -1, 1, 1, 1, -1, -1];%nls

%%
a = -0.2;
b = 0.2;
w = (b-a).*rand(1,4) + a;

%%
a1 = -1;
b1 = 1;
theta = (b1-a1).*rand + a1;

%%
eta = 0.02;

for t = 1:10^5
    t
    %for mu = 1:16;
    mu = randi([1 16]);

    % calculate O using equation given, with b calculated using 4 original
inputs for each mu
    O1 = tanh(0.5*(-theta+sum(w.*x(mu,:))));

    %calculate error for output layer
    error = (target(mu)-O1)*(1-O1^2)*0.5;
    for input = 1:4
        %calculate weight update and add to weight
        w_update = eta*error*x(mu,input);
        w(input) = w(input) + w_update;
    end
    % calculate threshold update and subtract from threshold
    theta_update = eta*error;
    theta = theta - theta_update;

end

%%
O = zeros(1,16);
out = zeros(1,16);
for mu = 1:16
    O(mu) = tanh(0.5*(-theta+sum(w.*x(mu,:))));
```

```
    if (O(mu) < 0)
        out(mu) = -1;
    end
    if (O(mu) > 0||O(mu) == 0)
        out(mu) = +1;
    end
end

%%
if (out(:,:) == target(:,:))
    fprintf("function is linearly separable\n")
else
    fprintf("function is not linearly separable\n")
end
```