```
%%%%%%%% One_Layer_Perceptron_Code  %%%%%%%%%

clear all;


target_trainingset = cell2mat(struct2cell(load("target_trainingset",'-mat')));
xTrainingset = zscore(cell2mat(struct2cell(load("xTrainingsetNormalized",'-mat'))));

NeuronNb = 15;


eta = 0.009;
N = length(xTrainingset(:,1));
w = normrnd(0,1,NeuronNb,2)/sqrt(2);
W= normrnd(0,1,1,NeuronNb)/sqrt(4);
theta2 =zeros(1,1);
theta1 = zeros(NeuronNb,1);

target_validationset = cell2mat(struct2cell(load("target_validationset",'-mat')));
xValidationset = zscore(cell2mat(struct2cell(load("xValidationset",'-mat'))));

Nval = length(xValidationset);


C = [];


%%
tic
for i = 1:10E2

    for JJ = 1:N/2

        idx=randperm(length(xTrainingset(:,:)'),1);
        mu = idx;

        A = w*xTrainingset(mu,:)'; % random patterns

        bj = A - theta1;
        V = tanh(bj);

        Bi = dot(W,V) - theta2;
        O = tanh(Bi);

        delW = eta*(target_trainingset(mu)-O).*(sech(Bi).^2).*V;
        delw = eta*(target_trainingset(mu)-O)*(sech(Bi).^2).*(sech(bj).^2).↙
*xTrainingset(mu,:).*W';

        delTheta2 = eta*(target_trainingset(mu)-O).*((sech(Bi).^2));
        delTheta1 = eta*(target_trainingset(mu)-O).*(sech(Bi).^2).*(sech(bj').^2).*W;
```

```matlab
        w = w + delw;
        W = W + delW';

        theta1 = theta1 - delTheta1';
        theta2 = theta2 - delTheta2;

    end
    Outputs = [];
    for mu = 1:Nval

        B = w*xValidationset(mu,:)';
        bjval = B - theta1;
        Vval = tanh(bjval);

        Bival = dot(W,Vval) - theta2;
        Oval = tanh(Bival);
        Outputs = [Outputs,Oval];
    end
    C = (1/(2*Nval))*sum(abs(sign(Outputs)-target_validationset'),2);
    if C < 0.12
        C
        break
    end
end


toc
```