

```

x = csvread('input_data_numeric.csv', 0, 1);
t = csvread('target_data.csv');

eta = 0.02;
weight = 0.2*(1 - 2*rand(1,4));
bias = 1 - 2*rand;
B = 1/2;
nbrOfTargetPatterns = length(t(1,:));
CVec = zeros(nbrOfTargetPatterns,1);
for tInd = 1:nbrOfTargetPatterns
    tLoop = t(:,tInd);

    p = length(tLoop);

    for k = 1:10^5
        output = Output(weight, x, bias);
        H = EnergyFunction(tLoop, output);
        C = ErrorFunction(tLoop, output);
        if C == 0
            break;
        end
        mu = randi([1,p]);
        weightTmp = Update_w(weight, x, tLoop, output, eta, mu, bias, B);
        bias = UpdateBias(weight, x, tLoop, output, eta, mu, bias, B);
        weight = weightTmp;
    end
    CVec(tInd) = C;
end
disp([(1:nbrOfTargetPatterns)', CVec])

function H = EnergyFunction(t, output)
    sumVec = (t - output).^2;
    H = (1/2) * sum(sumVec);
end

function C = ErrorFunction(t, output)
    signOutput = sign(output);
    signOutput(signOutput == 0) = -1;
    sumVec = abs(signOutput - t);
    C = (1/2) * sum(sumVec);
end

function newWeight = Update_w(weight, x, t, output, eta, mu, bias, B)
    sumTot = weight * x(mu, :)' ;
    b = sumTot - bias;
    dw = eta * B * (t(mu) - output(mu)) * (1 - (tanh(B * b)).^2) * x(mu, :);
    newWeight = weight + dw;
end

function newBias = UpdateBias(weight, x, t, output, eta, mu, bias, B)
    sumTot = weight * x(mu, :)' ;
    b = sumTot - bias;
    dTheta = -eta * B * (t(mu) - output(mu)) * (1 - (tanh(B * b)).^2);
    newBias = bias + dTheta;
end

function output = Output(weight, x, bias)
    sumTot = weight * x';
    output = tanh((1/2) * (sumTot - bias))';
end

```