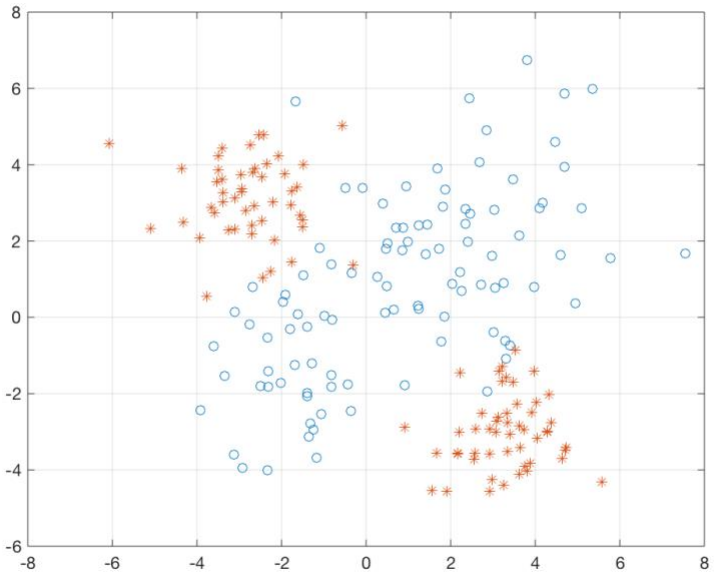
1.1

```
clear
load('hw5.mat');
x1=hw5_1;
x2=hw5_2;

%1.1
figure(1);
plot(x1(1,:),x1(2,:),'o',x2(1,:),x2(2,:),'*');
grid on
```



1.2

```
%1.2
X=[x1 x2];%1.2.1
target1=[ones(1,100); -1*ones(1,100)];%1.2.2
target2=[-1*ones(1,100); ones(1,100)];
T=[target1 target2];
meanX=mean(X,2);%1.2.3
a=0;
for i=[1:200]
    a=a+(X(:,i)-meanX).*(X(:,i)-meanX);
end
stdX=a/200;
X=(X-meanX)./stdX;
theta=0.1; %1.2.4  % criterion to stop
```

```
ita=0.1; % step size
Nh=10;  % number of hidden nodes, actual hidden nodes should be 11
(including a biase)
Ni=2; % dimension of input vector = number of input nodes, actual input
nodes should be 3 (including a biase)
No=2; % number of class = number of out nodes
%input nodes=2, plus 1 bias, hidden nodes=10, so the size of in-to-
hidden weights should be 10x3
a=-(0.2)^-0.5;b=(0.2)^-0.5;%1.2.5
c=-(5)^-0.5;d=(5)^-0.5;
weights_intohidden = (b-a).*rand(10,3) + a;
%hidden nodes=10, plus 1 bias, output  nodes=2, so the size of hidden-
to-out weights should be 2x11
weights_hiddentoout=(d-c).*rand(2,11) + c;
train=randperm(200, 1);%1.2.6
X_train=X(:,train);
t_train=T(:,train);
X_0=1;%1.2.7
net_j=weights_intohidden*[X_0 ;X_train];
Y=1.716*tanh(2/3*net_j); %1.2.8
Y_0=1;%1.2.9
net_k=weights_hiddentoout*[Y_0 ;Y];
Z=1.716*tanh(2/3*net_k);%1.2.10
delta_k=(t_train-
Z).*(1.716*2/3*sech(2/3*net_k).*sech(2/3*net_k));%1.2.11
delta_j=sum(delta_k.*weights_hiddentoout(:,2:11))'.*[1.716*2/3*sech(2/3
*net_j).*sech(2/3*net_j)];%1.2.12
delta_weights_hiddentoout=ita*delta_k*[Y_0;Y]'%1.2.13
delta_weights_intohidden=ita*delta_j*[X_0;X_train]'

%1.2.14
Sumj=0;
for p=[1:200]
    X_train=X(:,p);
    t_train=T(:,p);
    X_0=1;%1.2.7
    net_j=delta_weights_intohidden*[X_0 ;X_train];
    Y=1.716*tanh(2/3*net_j); %1.2.8
    Y_0=1;%1.2.9
    net_k=delta_weights_hiddentoout*[Y_0 ;Y];
    Z=1.716*tanh(2/3*net_k);%1.2.10
```

```
    Sumj=Sumj+[mean((t_train-Z).*(t_train-Z))];
end
errorrate=(Sumj-200)/200
if errorrate >= theta
    disp('Back to step 6')
else
    disp('Finished')
end
```

```
delta_weights_hiddentoout =

  列 1 至 7

   -0.0596    0.0034    0.0475   -0.0959   -0.0750    0.0915   -0.0106
   -0.0090    0.0005    0.0072   -0.0145   -0.0114    0.0139   -0.0016

  列 8 至 11

   -0.0357   -0.0883    0.0429    0.0967
   -0.0054   -0.0134    0.0065    0.0147


delta_weights_intohidden =

   -0.0076   -0.0028    0.0042
   -0.0159   -0.0059    0.0089
   -0.0002   -0.0001    0.0001
   -0.0036   -0.0013    0.0020
    0.0041    0.0015   -0.0023
    0.0336    0.0125   -0.0188
    0.0072    0.0027   -0.0040
    0.0023    0.0008   -0.0013
   -0.0259   -0.0097    0.0145
    0.0000    0.0000   -0.0000

errorrate =

    0.0025
```

1.3

```
%1.3
X_test=[[2;2],[-3;-3],[-2;5],[3;-4]];
for p=[1:4]
    X_train=X_test(:,p);
    net_j=delta_weights_intohidden*[X_0 ;X_train];
    Y=1.716*tanh(2/3*net_j); %1.2.8
    Y_0=1;%1.2.9
```

```
    net_k=delta_weights_hiddentoout*[Y_0;Y];
    Z=1.716*tanh(2/3*net_k);%1.2.10
    pw1=mean(([1;-1]-Z).*([1;-1]-Z));
    pw2=mean(([-1;1]-Z).*([-1;1]-Z));
    if pw1>pw2
        C(p)=2;
    else
        C(p)=1;
    end
end
disp(C)
```

```
     2     2     2     2
```