**Q4.**

%% Implement Nearest Neighbour algorithm without reading Q5.

clear all; close all; clc;%% Generate a distribution for each covariance matrix defined.

load('F:\Fall 2017\Math Foundations for ML\Assignments\Homework 10\hw10p4\_data.mat') % Reading the data given..

%% Generating a set of random values for the 2 variable r.

mu1 = [1;0];

sigma1 = [1 0;0 1];

%rng default % for reproducability

r = mvnrnd(mu1,sigma1,1000);

r = r';

%% each data point in r is to be compared with X1 and X2 separately..

for j = 1:1000

for i = 1:100

d1(j,i) = sqrt((r(1,j) - X1(1,i))^2 + (r(2,j) - X1(2,i))^2);

end

for i = 1:100

d2(j,i) = sqrt((r(1,j) - X2(1,i))^2 + (r(2,j) - X2(2,i))^2);

end

end

for k = 1:1000

mind1(k) = min(d1(k,:));

end

for k = 1:1000

mind2(k) = min(d2(k,:));

end

p1 = 0; p2 = 0;

for k = 1:1000

if mind1(k)<mind2(k)

plot(r(1,k),r(2,k),'bo'); hold on;

p1 = p1 + 1; % to find the number of r which are being classified as X1

end

if mind1(k)> mind2(k)

plot(r(1,k),r(2,k),'k\*'); hold on;

p2 = p2 + 1; % to find the number of r which are being classified as X1

end

end

