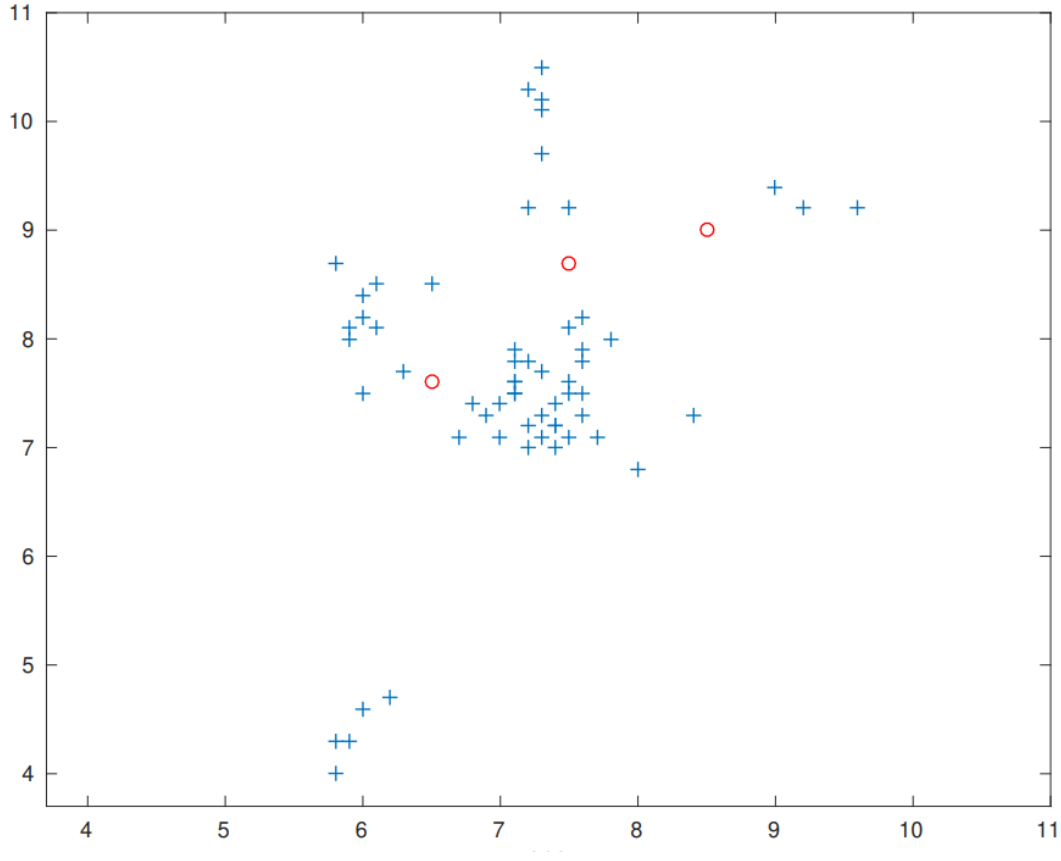
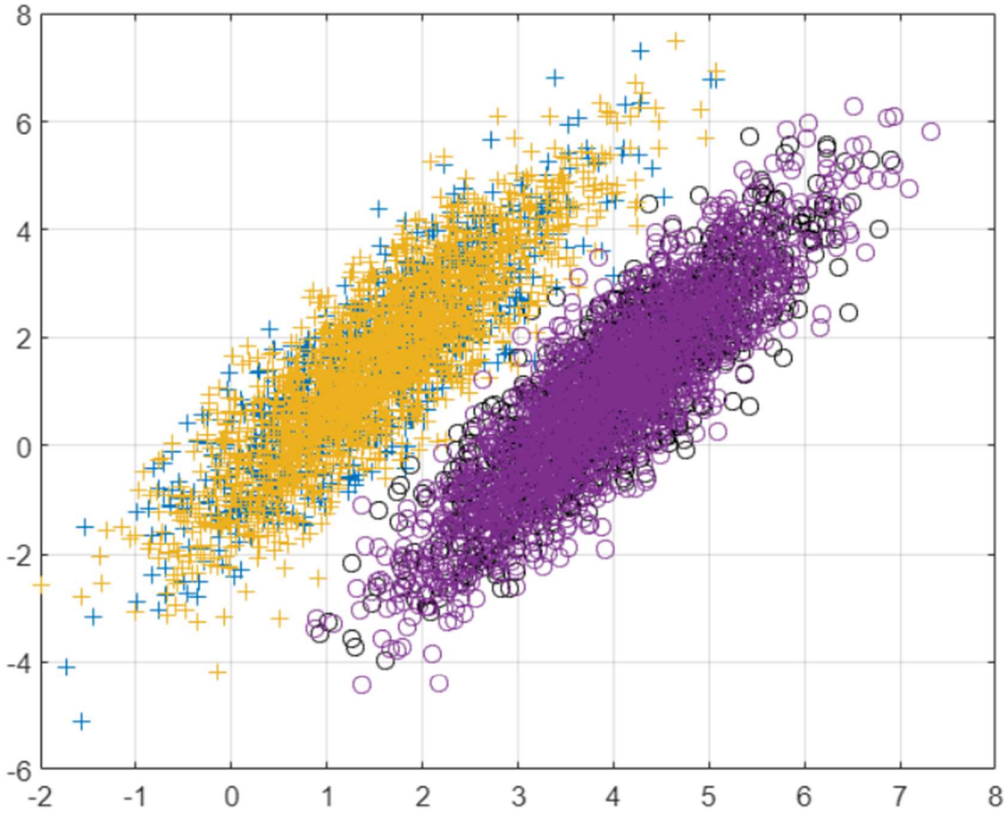
# MAT2003 (Optimization Techniques) Lab-11

**KHAN MOHD OWAIS RAZA**
**20BCD7138**

```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% MAT2003 (OT) Lab
clc
clear all
close all
% Orange-lemons data
function training_data = A;
% set seed for repeatable results
rng(2);
% Number of clusters
K = 5;
% Corresponding class labels for each training point
C = kmeans(training_data, K); % we use Matlab's built in function
% Concatenate labels to the training data (type: help cat)
training_data = cat(2, training_data, C);
% Random test data
test_data = [6.5 7.6;
7.5 8.7;
8.5 9];
% show test data on the same plot with the training data
plot(training_data(:,1), training_data(:,2), '+'); hold on;
xlabel(col_headers{1}); ylabel(col_headers{2});
% show test data as red circles
plot(test_data(:,1), test_data(:,2), 'ro');
axis([3.7 11 3.7 11]);
hold off;
end
```



```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% MAT2003 (OT) Lab
%% KNN Algorithm using Euclidean principle
clear;
close all;
clc;
mu = [1.5,1.5]; sigma = [1,1.5;1.5,3];
sampleNo = 3000;
testNo = 1000;
C1X = (mvnrnd(mu,sigma,sampleNo))';
mu = [4,1];sigma = [1,1.5;1.5,3];
C2X = (mvnrnd(mu,sigma,sampleNo))';
x = [C1X,C2X];
y1 = zeros(1,sampleNo);
y2 = ones(1,sampleNo);
y = [y1,y2];
test_x = [C1X(:,1:testNo),C2X(:,1:testNo)];
train_x = [C1X(:,testNo+1:sampleNo),C2X(:,testNo+1:sampleNo)];
test_y = [y1(:,1:testNo),y2(:,1:testNo)];
train_y = [y1(:,testNo+1:sampleNo),y2(:,testNo+1:sampleNo)];
plot(C1X(1,1:testNo),C1X(2,1:testNo),'+'); hold on; grid on;
plot(C2X(1,1:testNo),C2X(2,1:testNo),'ko');
plot(C1X(1,testNo+1:sampleNo),C1X(2,testNo+1:sampleNo),'+'); hold
on; grid on;
plot(C2X(1,testNo+1:sampleNo),C2X(2,testNo+1:sampleNo),'o');
knn_loop(test_x,train_x,2);
function test_data = knn_loop(test_data, tr_data,k)
numoftestdata = size(test_data,1);
numoftrainingdata = size(tr_data,1);
 for sample=1:numoftestdata
 R = repmat(test_data(sample,:),numoftrainingdata,1) ;
 euclideandistance = (R(:,1) - tr_data(:,1)).^2;
 [dist position] = sort(euclideandistance,'ascend');
 knearestneighbors=position(1:k);
 knearestdistances=dist(1:k);
 for i=1:k
 A(i) = tr_data(knearestneighbors(i),2);
 end
 M = mode(A);
 if (M~=1)
 test_data(sample,2) = mode(A);
 else
 test_data(sample,2) = tr_data(knearestneighbors(1),2);
 end
 end
end
```


8
6
4
2
0
-2
-4
-6
-2
-1
0
1
2
3
4
5
6
7
8