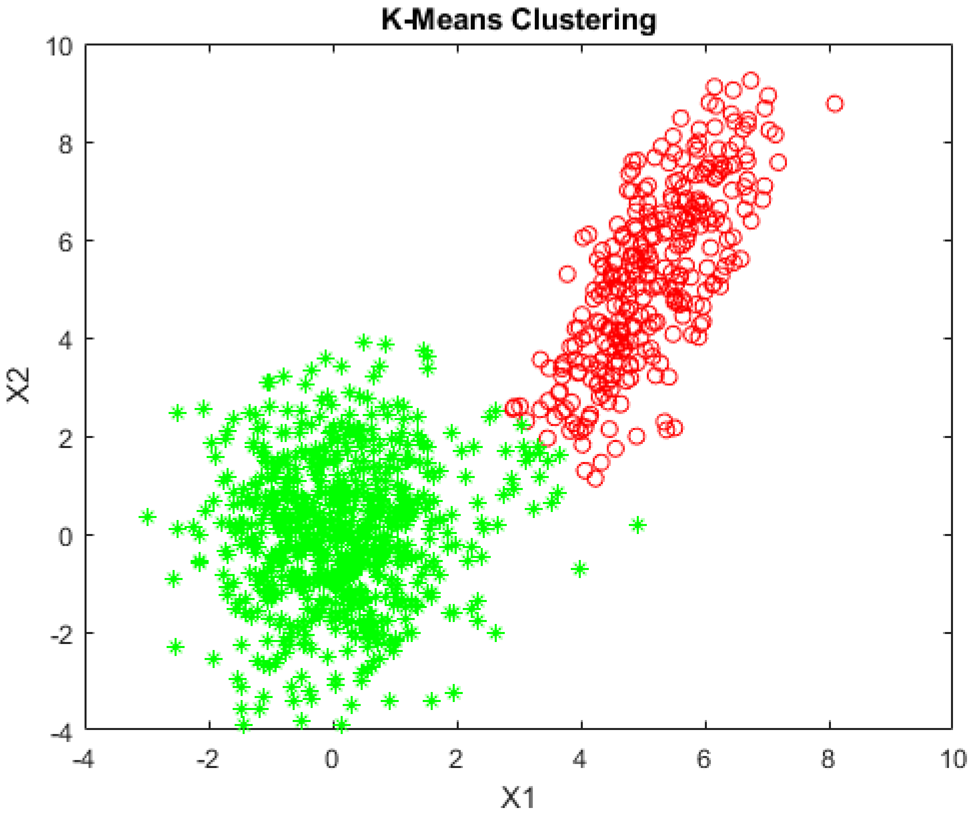
```
close all
clear all

load data.mat

temp = zeros(1,length(data));

% Randomly initialise the centroid index
Index_centroid = randi(length(data),2,1);

% Store the indices
centroids = data(:,Index_centroid);

% Start a loop, compare distances and keep updating the centroid until we
% reach a stable solution where there is no more updates
while 1
    % update temp
    for j=1:length(data)
        magnitude = sum((centroids - repmat(data(:,j), 1, 2)).^2);
        [y, index] =  min(magnitude);
        temp(j) = index;
    end
    Previous_centroids = centroids;

    % Update the centroids of the cluster
    for j=1:length(data)
        for k = 1:2
            samples = data(:,temp==k);
            centroids(:,k) = mean(samples,2);
        end
    end

    % Exit of the infinte loop is when there are no more updates
    if Previous_centroids == centroids
        break;
    end
end



disp('The centroid are (Each column represents a cluster center): ')
disp(centroids)

figure
plot(data(1,temp==1),data(2,temp==1),'ro')
hold on
plot(data(1,temp==2),data(2,temp==2),'g*')
xlabel('X1')
ylabel('X2')
title('K-Means Clustering')
```

```
disp('The new entry points belong to the upper cluster whose centroids are
 ~(5,5)')
```

```
The centroid are (Each column represents a cluster center):
    5.1334    0.1162
    5.2614    0.0267

The new entry points belong to the upper cluster whose centroids are ~(5,5)
```



Published with MATLAB® R2022a