Homework 2
SM2001 : Data-Driven Methods in Engineering
Kirtan Premprakash Patel

Problem 1
Part (a)

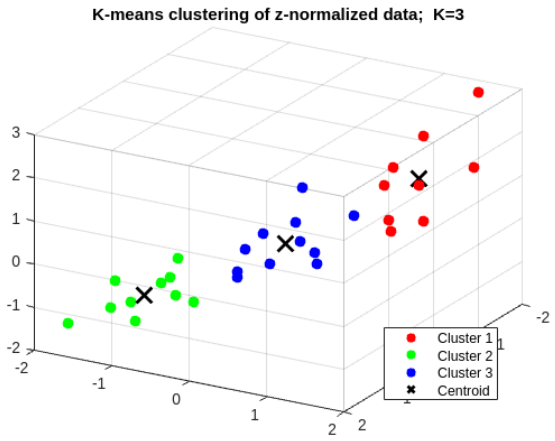


Figure 1: Clusters formed using K-means Clustering

K-means is sensitive to the initial placement of the centroids. Running the algorithm with different initializations and selecting the best result can help mitigate the risk of getting stuck in a local minimum. The algorithm is run multiple times with different initial centroid positions, and the solution with the lowest within-cluster sum of squared distances (WCSS) is chosen.

Part (b)

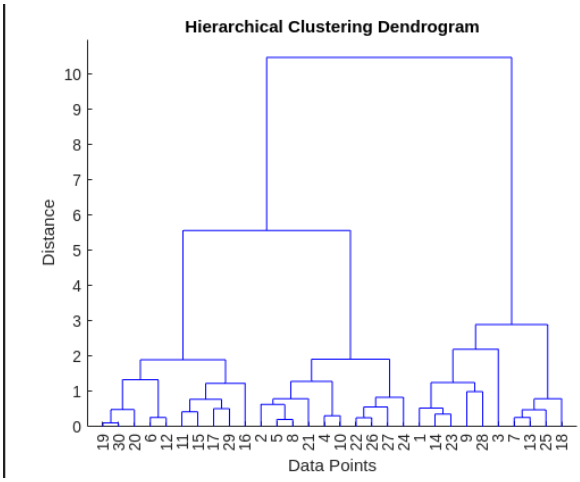
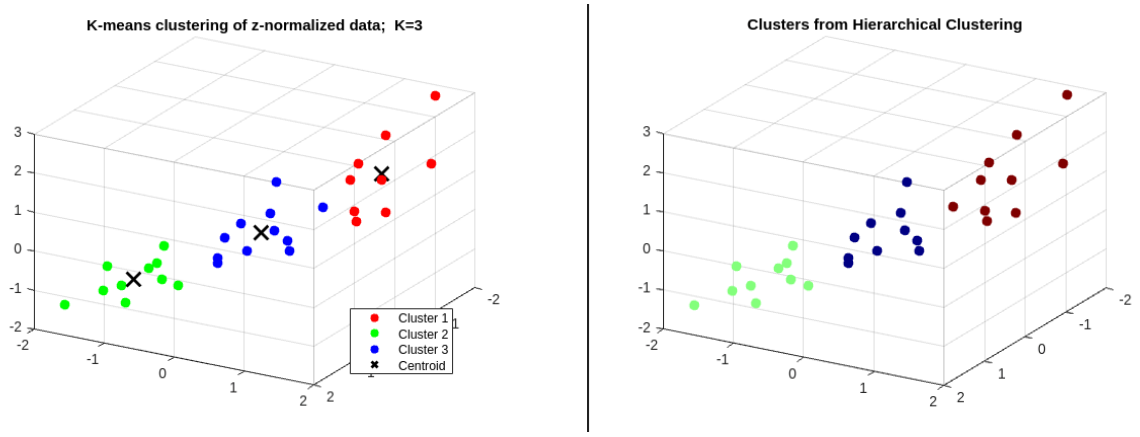


Figure 2: Dendogram created by Hierarchical Clustering

The dendogram can be used to visually understand the distance between the means of different datapoints. We start with with the two data points which are the closest in terms of euclidean distance i.e 19 and 30. These two are clubbed and their mean is then treated as a datapoint to calculate distances with other data points. In this way we can see that at each stage, the two data points with the smallest euclidean distance are paired up together to form a cluster. It is essential to note that repeating the clustering multiple times has no effect on the resultant clustering if the datapoints and the distance function remain the same. Hence, multiple iterations of this type of clustering can not be carried out in hopes of achieving a global optima. The clustering result obtained can not be assured to be optimal.

Comparison

The clustering obtained by hierarchical clustering and K-means clustering differs under certain scenarios.



(a) K-means Clusters

(b) Hierarchical Clusters

Figure 3: Comparison of results obtained from different clustering schemes

It can be seen that there is a difference in the allocation of cluster to a point that is almost evenly far from the the centroids of the K-means cluster.

In this case, it is observed that the datapoint is clustered along with the cluster of the nearest data point. This is because, the hierarchical clustering is agglomerative and hence if two data points, which should ideally belong to different clusters in the global optima, are near each other, they are clustered together. This can even be interpret as hierarchical clustering being more robust to outliers, while K-means clustering is sensitive to outliers as they can significantly impact the position of centroids.

It even suggests that K-means clustering is more suited to work well with clusters that are spherical, equally sied and have similar density.

APPENDIX A

MATLAB Codes

Problem 1

```
clear all
close all
clc

%% Load the data
X = load('data.mat').X;
% 30x3 data matrix for 30 different
% Column1 : strength [MPa]
% Column2 : strain to failure [%]
% Column3 : elastic modulus [GPa]

%% Normalize dataset
mean = mean(X);
stdDev = std(X);
zNormalData = normalize(X);
%X_reconstructed = zNormalData.*stdDev + mean;
%X_error = X_reconstructed - X;

%% Part (a) : K-means clustering
K = 3;
numInitializations = 1000;       % define the number of random
    initializations
choiceMultiple = 10;
% random centroids generated are numInitializations*choiceMultiple
% this is done to further increase randomness by using datasample
paddedNumInits = choiceMultiple*numInitializations;

% Variables to store the results
bestIdx = [];
bestCentroids = [];
bestWCSS = Inf;
randomCentroids = [randn(paddedNumInits, 1), randn(paddedNumInits, 1),
    randn(paddedNumInits, 1)];

for i = 1:numInitializations
    % Randomly initialize cluster centroids
    initialCentroids = datasample(randomCentroids, K, 'Replace', false
        );

    % Perform K-means clustering with these initial centroids
    [idx, centroids, sumd] = kmeans(zNormalData, K, 'Start',
        initialCentroids);

    % Calculate the total within-cluster sum of squares (WCSS)
    wcss = sum(sumd);

    % Check if this initialization results in a lower WCSS
    if wcss < bestWCSS
        bestIdx = idx;
        bestCentroids = centroids;
        bestWCSS = wcss;
        %disp(bestWCSS);
    end
end

```

```
52
53 % Plot the best clustered data
54 colors = ['r', 'g', 'b', 'c', 'm', 'y', 'k'];
55 figure;
56 hold on;
57 for i = 1:K
58     cluster_points = zNormalData(bestIdx == i, :);
59
60     scatter3(cluster_points(:, 1), cluster_points(:, 2),
           cluster_points(:, 3), 50, colors(i),'filled', 'DisplayName', ['
           Cluster ',num2str(i,'%d') ]);
61 end
62 colormap('jet');  % Set the colormap as needed
63 scatter3(bestCentroids(:, 1), bestCentroids(:, 2), bestCentroids(:, 3)
       , 200, 'k', 'Marker', 'x', 'LineWidth',2, 'DisplayName', 'Centroid'
       );
64 title(['K-means clustering of z-normalized data;  K=',num2str(K,'%d')
       ]);
65 legend('Location', 'Best');
66 grid on;
67 hold off;
68
69 % Set the perspective view
70 view(-30, -30); % Azimuth: 30 degrees, Elevation: 45 degrees
71
72
73 %% Part (b) : Hierarchical clustering
74
75 % Perform hierarchical clustering
76 Z = linkage(zNormalData, 'ward'); % You can choose different linkage
       methods
77
78 % Visualize the hierarchical clustering using a dendrogram
79 figure;
80 dendrogram(Z);
81
82 title('Hierarchical Clustering Dendrogram');
83 xlabel('Data Points');
84 ylabel('Distance');
85
86 % Use Hierarchical Clustering to form Clusters
87 T = cluster(Z,"maxclust",3);
88
89 % Display the clusters using different colors in a scatter plot
90 figure;
91 scatter3(zNormalData(:, 1), zNormalData(:, 2), zNormalData(:, 3), 50,
       T, 'filled');
92 title('Clusters from Hierarchical Clustering');
93 colormap('jet');
94
95 % Set the perspective view
96 view(-30, -30); % Azimuth: 30 degrees, Elevation: 45 degrees
```