# Intelligent Data Analysis

Homework 3

Sai Soudamini Chintala

chintasd@mail.uc.edu

M12484209

1. Perform k-means clustering with this dataset for values of k to be 3, 4, 5, 6, 7, and 8. For each case of k run the clustering algorithm with three different initial cluster centers and select the one with the lowest SSE value. Plot the SSE against the values of k. Report the following in the submitted work: (Use Matlab kmeans function or any other similar toolbox)?

**Importing data into data matrix:**

data = csvread(‘StudentData2.csv’);

data = data (1:50,2:5); (Ignoring the mean from rows and serial number from columns)

1. **Plot of the SSE values against the values of k:**

**Matlab code:**

k = [3,4,5,6,7,8];

arr\_sumd = int16.empty(6,0); %declaring an array of integers of size 6.

for i = 1:6

[idx,C,sumd] = kmeans(data,k(i),'Replicates',3,'Display','final');

arr\_sumd(i) = sum(sumd);

end

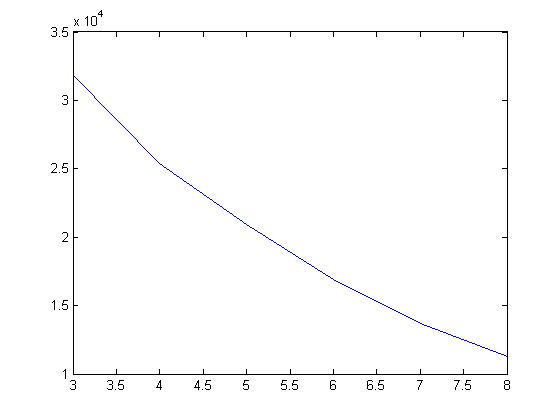
plot(k,arr\_sumd);

%Replicates is used to iterate the kmeans with unique initial cluster centers for three times and returns values for small sumd.

%arr\_sumd is the array that stores the SSE for clustering using k values 3,4,5,6,7,8.

%plot gives us the relation between k and SSE.

**Output:**



As the number of clusters increase the SSE value decreases.

1. **A plot of the silhouette coefficients for the data points in each clustering. (Each value of k results in one clustering)**

**Matlab code:**

k = [3,4,5,6,7,8];

arr\_sil = [];

for i = 1:6

[idx,C,sumd] = kmeans(data,k(i),'Replicates',3,'Display','final');

figure();

silhouette(data, idx);

s = silhouette(data, idx);

arr\_sil(i) = mean(s);

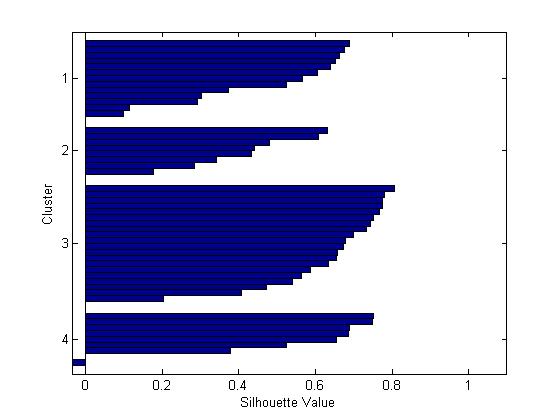
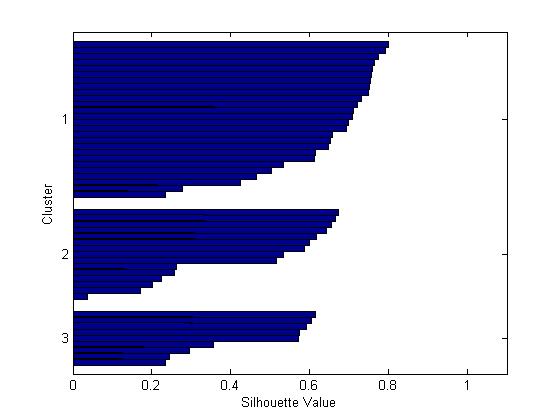
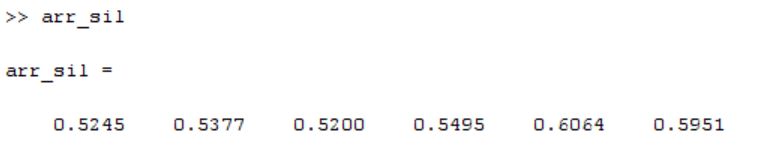
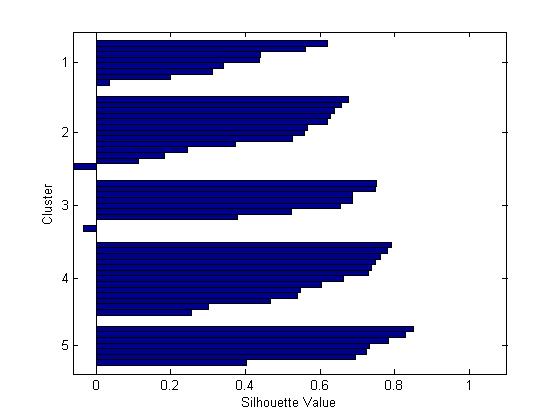
end

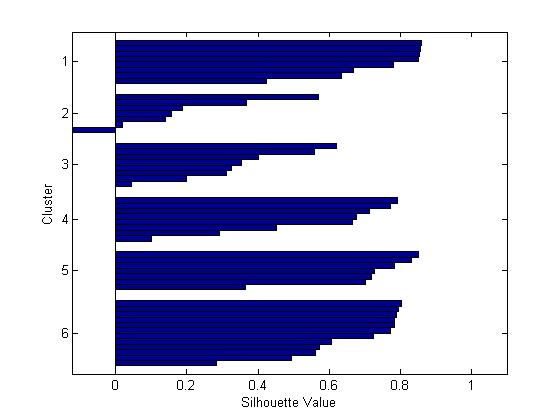
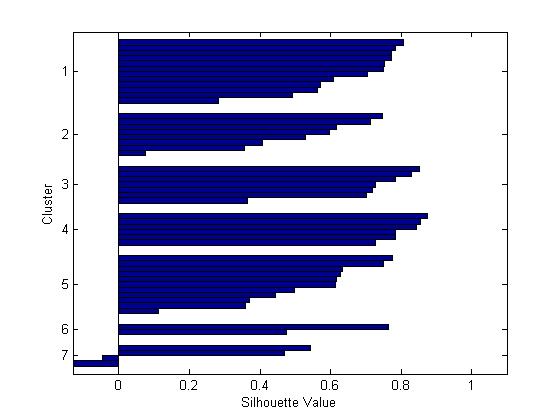
disp(arr\_sil);

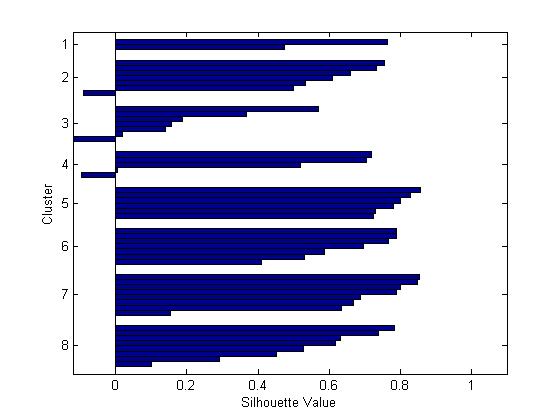
%figure() is used to give new figure every time in the loop.

%sillouette(data,idx) will plot the values as shown below.

%arr\_sil is used to store the sillouette values for each k.

**Output: **

****



1. **What is the best number of clusters for this dataset. Justify your choice for the best number of clusters**.

Best number of clusters for this data set is 7, as the silhouette value is maximum when the value of k (number of clusters) is 7.

1. **For your choice of the best number of clusters report the centroids of all the clusters (Call this as Clustering-1).**

**Matlab code:**

k = [3,4,5,6,7,8];

arr\_idx = int16.empty(6,0);

arr\_C = [];

arr\_sumd = int16.empty(6,0);

arr\_sil = [];

for i = 1:6

[idx,C,sumd] = kmeans(data,k(i),'Replicates',3,'Display','final');

arr\_sumd(i) = sum(sumd);

s = silhouette(data, idx);

arr\_sil(i) = mean(s);

arr\_C = cat(1,arr\_C,C);

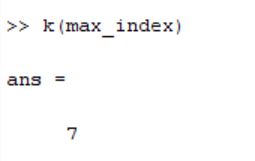
end

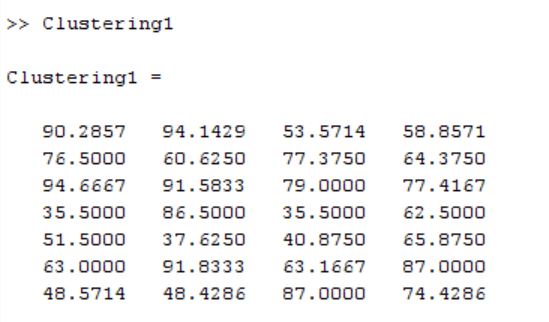
[max, max\_index] = max(arr\_sil);

Clustering1 = arr\_C(sum(k(1:max\_index-1))+1:sum(k(1:max\_index)),:);

disp(Clustering1);

**Output:**

****

****

Centroids of 7 clusters in Clustering1.

1. **Generate 50 random 4-dimensional random data points such that each attribute can take values between 0 and 100. With this dataset form the same number of clusters as selected by you in (c) above. Report the centroids and populations of the clusters. Compare the SSE for this dataset with the SSE for the provided dataset. Comment on the differences between the two values**

**Matlab code:**

rand\_data = randi([1 100],50,4);

[rand\_idx, rand\_C, rand\_sumd] = kmeans(rand\_data, k(max\_index));

population(1:k(max\_index)) = 0;

reshape(rand\_idx,1,50);

for i = 1:k(max\_index)

population(i) = sum(rand\_idx==i);

end

disp(rand\_C);

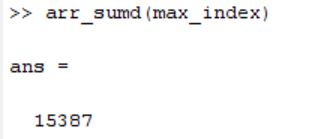
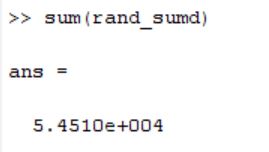
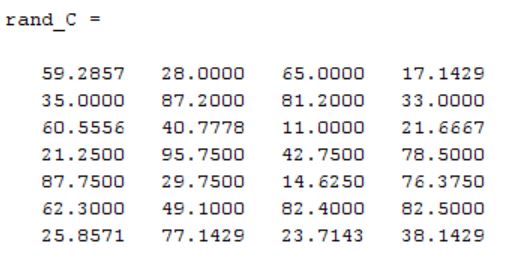
disp(population);

disp(arr\_sumd(max\_index));

disp(sum(rand\_sumd));

**Output:**

****



The SSE for random data is greater than the original data set. That means we have meaningful clusters.

1. Perform hierarchical clustering for the students’ scores dataset. Generate and show dendrograms for the cases (i) Single-Linkage clustering (Clustering-2), and (ii) Complete-Linkage clustering (Clustering-3). Use Euclidean distance for computing distance between data points. Report the following in the submitted work: (Use Matlab functions pdist and linkage, or any other similar toolbox.)

**Hierarchical Clustering:**

pair\_dist = pdist(data);

single\_linkage = linkage(data,'sinlge');

complete\_linkage = linkage(data,'complete');

* 1. **Dendrograms for the two clusterings (Clustering-2 and Clustering-3)**

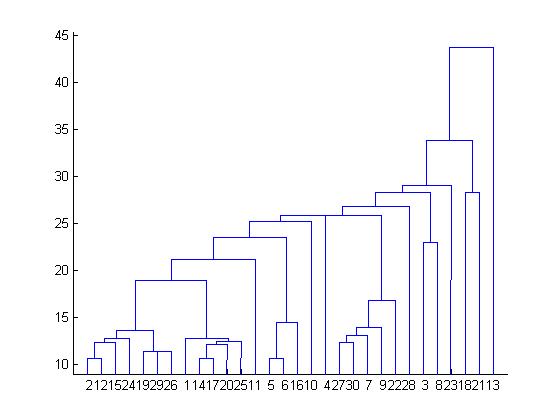
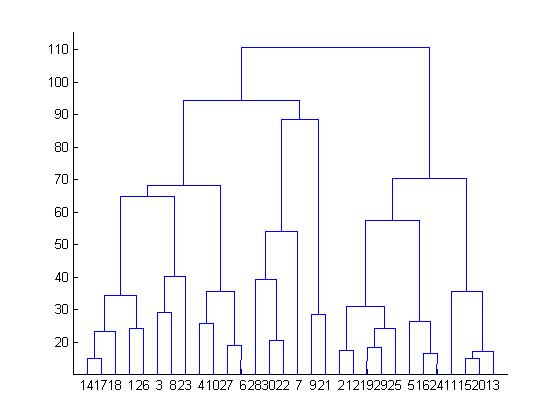
figure();

dendrogram(complete\_linkage);

figure();

dendrogram(single\_linkage);

**Output: Complete Linkage:Single Linkage:**



* 1. **Cluster compositions for each case when we need only four clusters. Write the data points included in each cluster and compute their centroids**.

**Matlab code:**

Clustering2\_idx = cluster(single\_linkage, 'maxclust',4);

Clustering3\_idx = cluster(complete\_linkage, 'maxclust', 4);

sin\_clust1 = [];

sin\_clust2 = [];

sin\_clust3 = [];

sin\_clust4 = [];

com\_clust1 = [];

com\_clust2 = [];

com\_clust3 = [];

com\_clust4 = [];

Clustering2\_idx = reshape(Clustering2\_idx,1,50);

Clustering3\_idx = reshape(Clustering3\_idx,1,50);

for i = 1:50

switch Clustering2\_idx(i)

case 1

sin\_clust1 = cat(1,sin\_clust1,data(i,:));

case 2

sin\_clust2 = cat(1,sin\_clust2,data(i,:));

case 3

sin\_clust3 = cat(1,sin\_clust3,data(i,:));

case 4

sin\_clust4 = cat(1,sin\_clust4,data(i,:));

end

switch Clustering3\_idx(i)

case 1

com\_clust1 = cat(1,com\_clust1,data(i,:));

case 2

com\_clust2 = cat(1,com\_clust2,data(i,:));

case 3

com\_clust3 = cat(1,com\_clust3,data(i,:));

case 4

com\_clust4 = cat(1,com\_clust4,data(i,:));

end

end

sin\_centroid1 = mean(sin\_clust1,1);

sin\_centroid2 = mean(sin\_clust2,1);

sin\_centroid3 = mean(sin\_clust3,1);

sin\_centroid4 = mean(sin\_clust4,1);

com\_centroid1 = mean(com\_clust1,1);

com\_centroid2 = mean(com\_clust2,1);

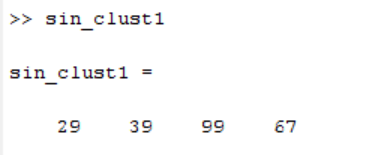
com\_centroid3 = mean(com\_clust3,1);

com\_centroid4 = mean(com\_clust4,1);

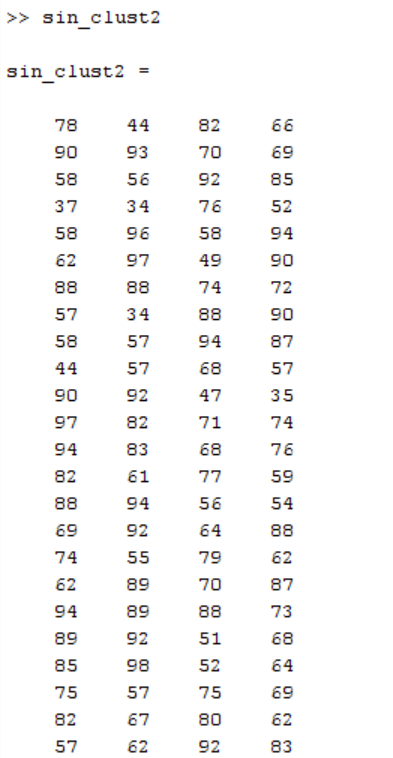
**Output:**

**Single Linkage:**

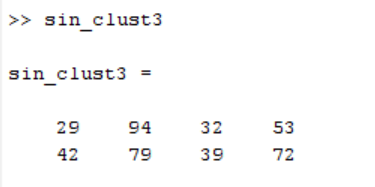
**Cluster1:**



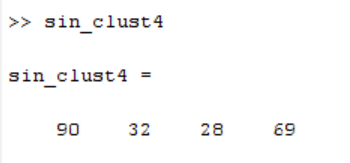
**Cluster2:**



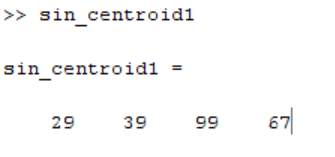
**Cluster 3:**

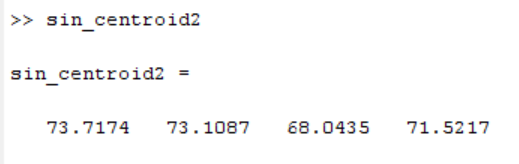


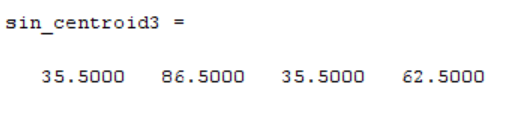
**Cluster4:**

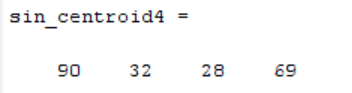


**Centroids:**



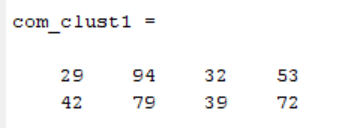




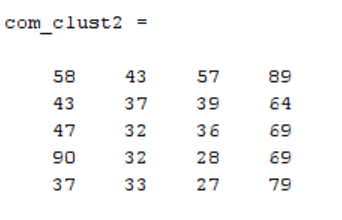


**Complete Linkage:**

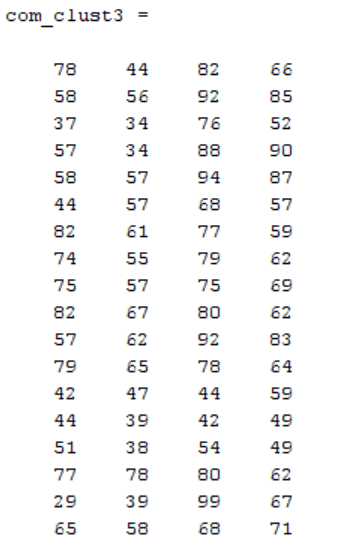
**Cluster 1:**



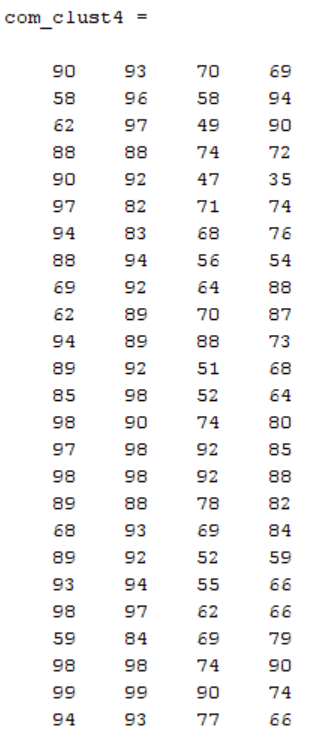
**Cluster 2:**



**Cluster 3:**



**Cluster 4:**



**Centroids:**









* 1. **Comment on any differences in the cluster centers and cluster compositions for the two different clusterings as performed in (b) above**

In single linkage cluster 2 has consists of 90% of data while in complete linkage the data is distributed between 2, 3, 4 clusters.

* 1. **Compute Rand Index for the comparison of Clustering-2 and Clustering-3 and show the counts a, b, c, and d as determined for computing the Rand index. Explain the meaning of each count and why such counts have been obtained for this dataset and these clusterings in this comparison**

**Matlab code:**

a=0;

b=0;

c=0;

d=0;

for i = 1:50

for j = i+1:50

if Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering3\_idx(i)==Clustering3\_idx(j)

a = a+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering3\_idx(i)~=Clustering3\_idx(j)

b = b+1;

elseif Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering3\_idx(i)~=Clustering3\_idx(j)

c = c+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering3\_idx(i)==Clustering3\_idx(j)

d = d+1;

end

end

end

disp(a);

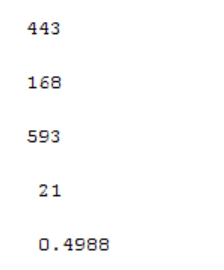
disp(b);

disp(c);

disp(d);

disp((a+b)/(a+b+c+d));

**Output**



A is the count of pair of points which are in one cluster in both Clustering2 and Clustering3.

B is the count of pair of points which are in different clusters in both Clustering2 and Clustering3.

C is the count of pair of points which are in one cluster in Clustering2 and in different clusters is Clustering3.

D is the count of pair of points which are in different clusters in Clustering2 and in one cluster in Clustering3.

As all the points in Clustering2 belong to cluster 2 most of the pairs belong to same cluster 2 but they are distributed among clusters 2, 3, 4 in Clustering3 so C value is high among them.

Rand index suggest that both the cluster do not agree on half of the pair of points.

1. Compute Rand Index for the comparison of Clustering-1 and Clustering-2 and show the counts a, b, c, and d as determined for computing the Rand index. Explain the meaning of each count and why such counts have been obtained for this dataset and these clusterings in this comparison

**Matlab code:**

a=0;

b=0;

c=0;

d=0;

k = [3,4,5,6,7,8];

arr\_idx = [];

arr\_C = [];

arr\_sumd = int16.empty(6,0);

arr\_sil = [];

for i = 1:6

[idx,C,sumd] = kmeans(data,k(i),'Replicates',3,'Display','final');

arr\_sumd(i) = sum(sumd);

arr\_idx = cat(3,arr\_idx,idx);

s = silhouette(data, idx);

arr\_sil(i) = mean(s);

arr\_C = cat(1,arr\_C,C);

end

[max, max\_index] = max(arr\_sil);

Clustering1\_idx = narr\_idx(:,:,max\_index);

disp(Clustering1\_idx);

for i = 1:50

for j = i+1:50

if Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering1\_idx(i)==Clustering1\_idx(j)

a = a+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering1\_idx(i)~=Clustering1\_idx(j)

b = b+1;

elseif Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering1\_idx(i)~=Clustering1\_idx(j)

c = c+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering1\_idx(i)==Clustering1\_idx(j)

d = d+1;

end

end

end

disp(a);

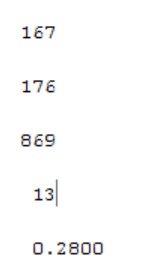
disp(b);

disp(c);

disp(d);

disp((a+b)/(a+b+c+d));

**Output:**



A is the count of pair of points which are in one cluster in both Clustering2 and Clustering1.

B is the count of pair of points which are in different clusters in both Clustering2 and Clustering1.

C is the count of pair of points which are in one cluster in Clustering2 and in different clusters is Clustering1.

D is the count of pair of points which are in different clusters in Clustering2 and in one cluster in Clustering1.

As all the points in Clustering2 belong to cluster 2 most of the pairs belong to same cluster 2 but they are distributed among clusters 1, 2, 4, 5, 6, 7 in Clustering1 so C value is high among them.

Rand index suggest that both the cluster do not agree on majority of the pair of points. They are totally different.

**Total Matlab Code:**

%data = data(1:50,2:5);

**%Question 1**

k = [3,4,5,6,7,8];

arr\_idx = [];

arr\_C = [];

arr\_sumd = int16.empty(6,0);

arr\_sil = [];

for i = 1:6

[idx,C,sumd] = kmeans(data,k(i),'Replicates',3,'Display','final');

arr\_sumd(i) = sum(sumd);

figure();

silhouette(data, idx);

s = silhouette(data, idx);

arr\_idx = cat(3,arr\_idx,idx);

arr\_sil(i) = mean(s);

arr\_C = cat(1,arr\_C,C);

end

%disp(arr\_sil);

[max, max\_index] = max(arr\_sil);

%disp(sum(k(1:max\_index-1)));

%disp(sum(k(1:max\_index)));

Clustering1 = arr\_C(sum(k(1:max\_index-1))+1:sum(k(1:max\_index)),:);

disp(Clustering1);

Clustering1\_idx = arr\_idx(:,:,max\_index);

disp(Clustering1\_idx);

rand\_data = randi([1 100],50,4);

[rand\_idx, rand\_C, rand\_sumd] = kmeans(rand\_data, k(max\_index));

population(1:k(max\_index)) = 0;

reshape(rand\_idx,1,50);

for i = 1:k(max\_index)

population(i) = sum(rand\_idx==i);

end

disp(rand\_C);

disp(population);

disp(arr\_sumd(max\_index));

disp(sum(rand\_sumd));

**%Question 2**

pair\_dist = pdist(data);

single\_linkage = linkage(pair\_dist,'single');

complete\_linkage = linkage(pair\_dist,'complete');

figure();

dendrogram(single\_linkage);

figure();

dendrogram(complete\_linkage);

Clustering2\_idx = cluster(single\_linkage, 'maxclust',4);

Clustering3\_idx = cluster(complete\_linkage, 'maxclust', 4);

sin\_clust1 = [];

sin\_clust2 = [];

sin\_clust3 = [];

sin\_clust4 = [];

com\_clust1 = [];

com\_clust2 = [];

com\_clust3 = [];

com\_clust4 = [];

Clustering2\_idx = reshape(Clustering2\_idx,1,50);

Clustering3\_idx = reshape(Clustering3\_idx,1,50);

for i = 1:50

switch Clustering2\_idx(i)

case 1

sin\_clust1 = cat(1,sin\_clust1,data(i,:));

case 2

sin\_clust2 = cat(1,sin\_clust2,data(i,:));

case 3

sin\_clust3 = cat(1,sin\_clust3,data(i,:));

case 4

sin\_clust4 = cat(1,sin\_clust4,data(i,:));

end

switch Clustering3\_idx(i)

case 1

com\_clust1 = cat(1,com\_clust1,data(i,:));

case 2

com\_clust2 = cat(1,com\_clust2,data(i,:));

case 3

com\_clust3 = cat(1,com\_clust3,data(i,:));

case 4

com\_clust4 = cat(1,com\_clust4,data(i,:));

end

end

sin\_centroid1 = mean(sin\_clust1,1);

sin\_centroid2 = mean(sin\_clust2,1);

sin\_centroid3 = mean(sin\_clust3,1);

sin\_centroid4 = mean(sin\_clust4,1);

com\_centroid1 = mean(com\_clust1,1);

com\_centroid2 = mean(com\_clust2,1);

com\_centroid3 = mean(com\_clust3,1);

com\_centroid4 = mean(com\_clust4,1);

a=0;

b=0;

c=0;

d=0;

for i = 1:50

for j = i+1:50

if Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering3\_idx(i)==Clustering3\_idx(j)

a = a+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering3\_idx(i)~=Clustering3\_idx(j)

b = b+1;

elseif Clustering2\_idx(i)==Clustering2\_idx(j) && Clustering3\_idx(i)~=Clustering3\_idx(j)

c = c+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && Clustering3\_idx(i)==Clustering3\_idx(j)

d = d+1;

end

end

end

disp(a);

disp(b);

disp(c);

disp(d);

disp((a+b)/(a+b+c+d));

**%Question 3**

a=0;

b=0;

c=0;

d=0;

for i = 1:50

for j = i+1:50

if Clustering2\_idx(i)==Clustering2\_idx(j) && fin\_idx(i)==fin\_idx(j)

a = a+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && fin\_idx(i)~=fin\_idx(j)

b = b+1;

elseif Clustering2\_idx(i)==Clustering2\_idx(j) && fin\_idx(i)~=fin\_idx(j)

c = c+1;

elseif Clustering2\_idx(i)~=Clustering2\_idx(j) && fin\_idx(i)==fin\_idx(j)

d = d+1;

end

end

end

for i = 1:k(nmax\_index)

population(i) = sum(fin\_idx==i);

end

disp(a);

disp(b);

disp(c);

disp(d);

disp((a+b)/(a+b+c+d));