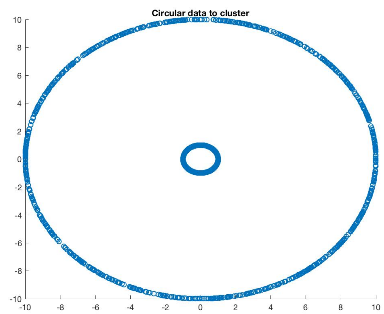
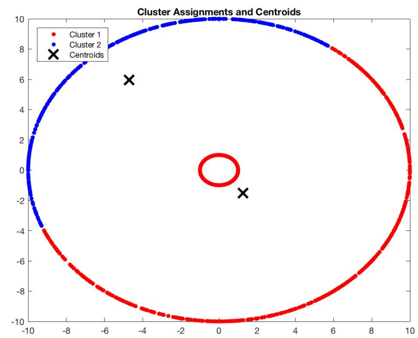
Question 3

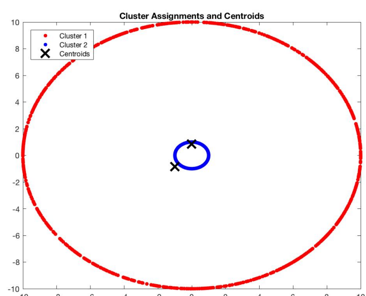
(a) The Original Data and K-means in Cartesian Coordinates



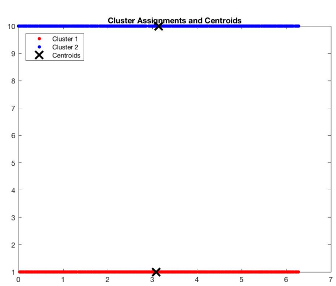


When K = 2 , The K-means method in Cartesian Coordinates basically means bisecting all data in Euclidean Space. However, the given data are distributed on two concentric circles, which cannot be bisected by a line in R^2 space.

(b) It is Clearly showed that the data distributed on each circle has been separated into two different clusters.



(c) If we look the data in the polar coordinate space, we can see such polar representation has separate the circle data into linear data. And K-means can easily bisect such data into two clusters in this space.



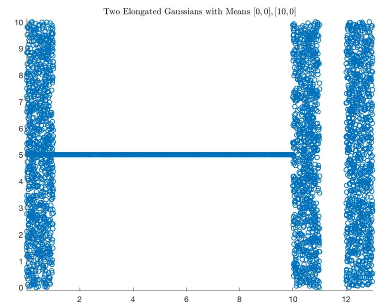
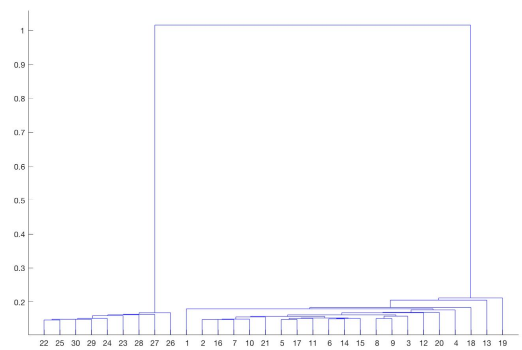
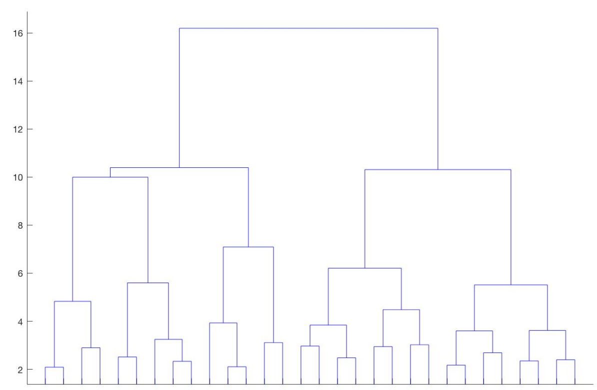
(x-axis is angle and y-axis is radius)

Code to produce k means result for Circular data in terms of Cartesian and Polar Coordinates

```
%% Load some random data laying on two circles.

%% Circle of radius 1.
% Note this is not uniform on the circle, despite being uniform in theta1.
theta1=2*pi*rand(1000,1);
X1=[cos(theta1),sin(theta1)];
%polar coor.
R1=ones(1000,1);
X1_p=[theta1,R1];

%% Circle of radius 2.
% Note this is not uniform on the circle, despite being uniform in theta1.

theta2=2*pi*rand(1000,1);
X2=10*[cos(theta2),sin(theta2)];

%polar Coor.
R2=10*ones(1000,1);
X2_p=[theta2,R2];



%% Concatenate data and plot
X=vertcat(X1,X2);
%Polar
X_p=vertcat(X1_p,X2_p)

close all;
scatter(X(:,1),X(:,2));
title('Circular data to cluster');


[idx,C] = kmeans(X,2,'Display','Iter','Replicates',100);


%Plot figure
figure;
plot(X(idx==1,1),X(idx==1,2),'r.','MarkerSize',12)
hold on
plot(X(idx==2,1),X(idx==2,2),'b.','MarkerSize',12)
plot(C(:,1),C(:,2),'kx',...
     'MarkerSize',15,'LineWidth',3)
```

```
legend('Cluster 1','Cluster 2','Centroids',...
       'Location','NW')
title 'Cluster Assignments and Centroids'
hold off


%Do k means for polar coordinates

[idx_p,C_p] = kmeans(X_p,2,'Display','Iter','Replicates',100);

%change centroid back to cartestion coordinates
C1_p=[cos(C_p(1,1)),sin(C_p(2,1))];
C2_p=[cos(C_p(1,2)),sin(C_p(2,2))];

%Plot figure for polar coordinates
figure;
plot(X(idx_p==1,1),X(idx_p==1,2),'r.','MarkerSize',12)
hold on
plot(X(idx_p==2,1),X(idx_p==2,2),'b.','MarkerSize',12)
plot(C1_p,C2_p,'kx',...
     'MarkerSize',15,'LineWidth',3)
legend('Cluster 1','Cluster 2','Centroids',...
       'Location','NW')
title 'Cluster Assignments and Centroids'
hold off
```

Question 4

Original Data	
Dendrogram by single linkage	
Dendrogram by complete linkage	

For single linkage, the distance between cluster $X_1$and X_2, $D(X_1,X_2) = \min D(x_1,x_2)$, where $x_1 \in X_1, x_2 \in X_2$.

And for complete linkage, the distance between cluster $X_1$and X_2, $D(X_1,X_2) = \max D(x_1,x_2)$, where $x_1 \in X_1, x_2 \in X_2$, where the size of the cluster become matter.

Accord the code that generates this dataset (you can see they are from S_1, S_2 S_3 and S_4) , any subclusters from S_1, S_2 or S_3 will merge first, before they merge with any subclusters from S_4 , because you can always find two neighbor subclusters in $S_1 \cup S_2 \cup S_3$, who have elements $D(x_1, x_2) < 1$. And 1 is the minimum distance between any point in S_3 and any point in S_4 .

```
DistortionMatrix=[1,0;0,10];
S1=rand(1000,2);
S1=S1*DistortionMatrix;

S2=rand(1000,2);
S2=S2*DistortionMatrix;
S2=S2+[10,0];

S3=10*rand(1000,2);
D2Matrix=[1,0;0,0];
S3 = S3*D2Matrix+[0,5]

S4=rand(1000,2);
S4=S4*DistortionMatrix;
S4=S4+[12,0];

X=vertcat(X1,X2,X3,X4);
```

However, the situation is different in complete linkage method. When the subclusters from different regions are **sufficiently large**, two neighbor subclusters S_3 or S_2 has $D(X_1, X_2) >> 1$. Then some subclusters in S_3 may merge with some subcluster in S_4 first, because the distance between S_3 and S_4 is neglectable to the size of such clusters.