Name: Akond Rahman
aarahman@ncsu.edu
HW#2
Date: Feb 06, 2016

Answer to RQ-1:
Why k-means clustering is sub-optimal?
In k-means clustering the cluster centroids are decided by cluster means. The data is split halfway between cluster means, and this can lead to suboptimal splits.

List 3 advantages and 3 disadvantages of k-means clustering?
Answer:
Advantages

- Simple algorithm
- Executes fast for large datasets while keeping number of clusters to be small
- Produces tighter clusters than those produced by hierarchical clusters

Disadvantages

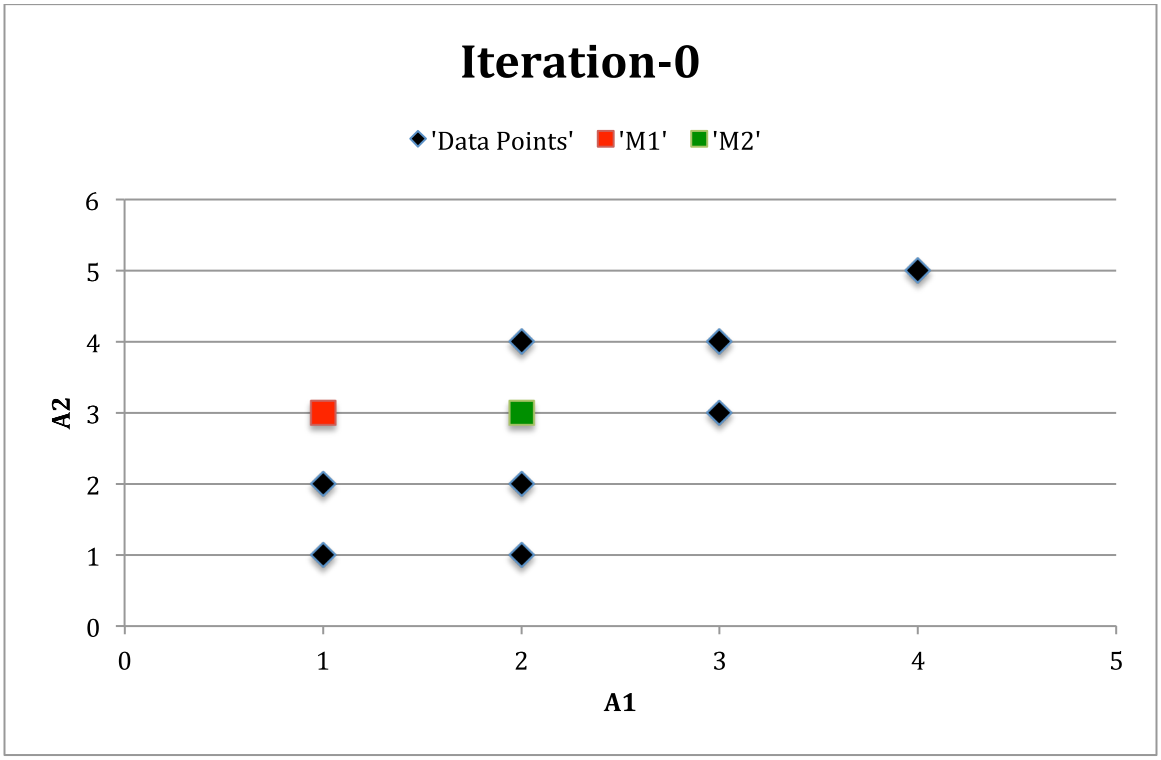
- K-means clustering is strongly sensitivity to outliers and noise
- K-means doesn't work well with non-circular cluster shape such as clusters that are non-elliptical
- K-means suffers from being sub-optimal

As K-means is sub-optimal, describe 3 ways to select initial centroids that may help finding good solution.
Answer:

- first select initial centroids more than k, and then from those centroids select the initial centroids
- perform multiple runs of the k-means algorithm to determine an optimal selection of centroids
- apply bisecting k-means

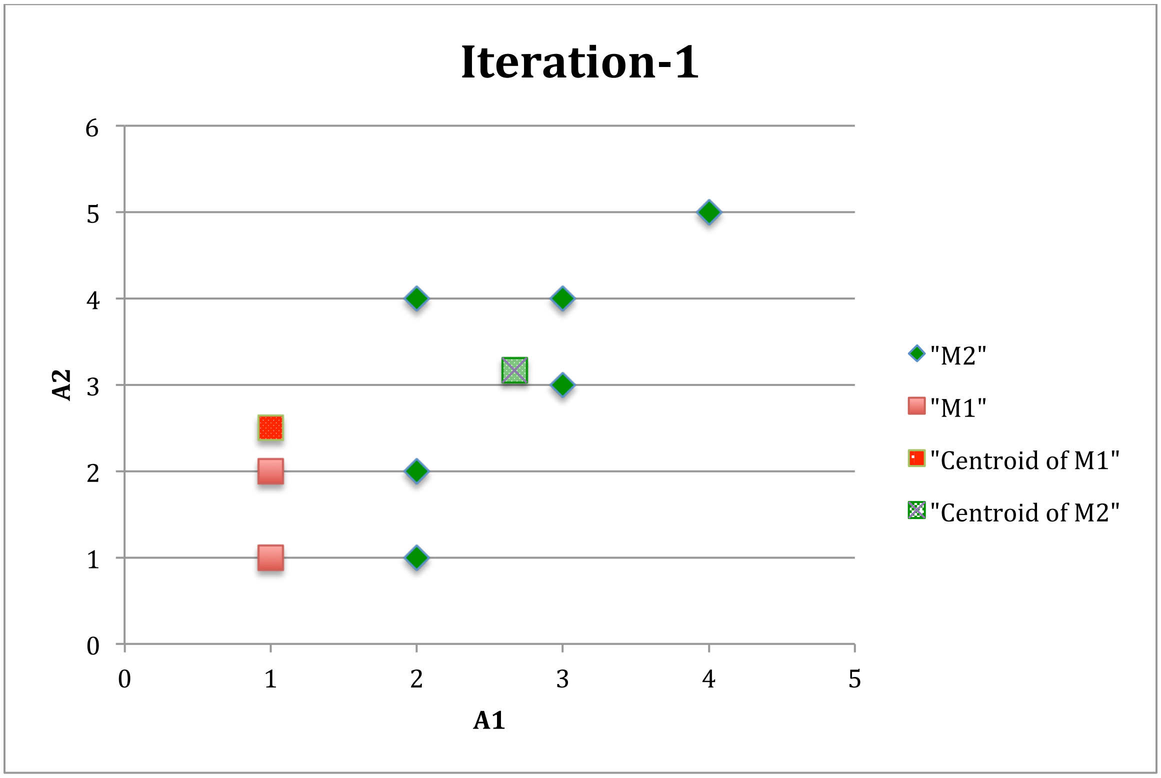
Answer to 1(d)

Iteration-0:



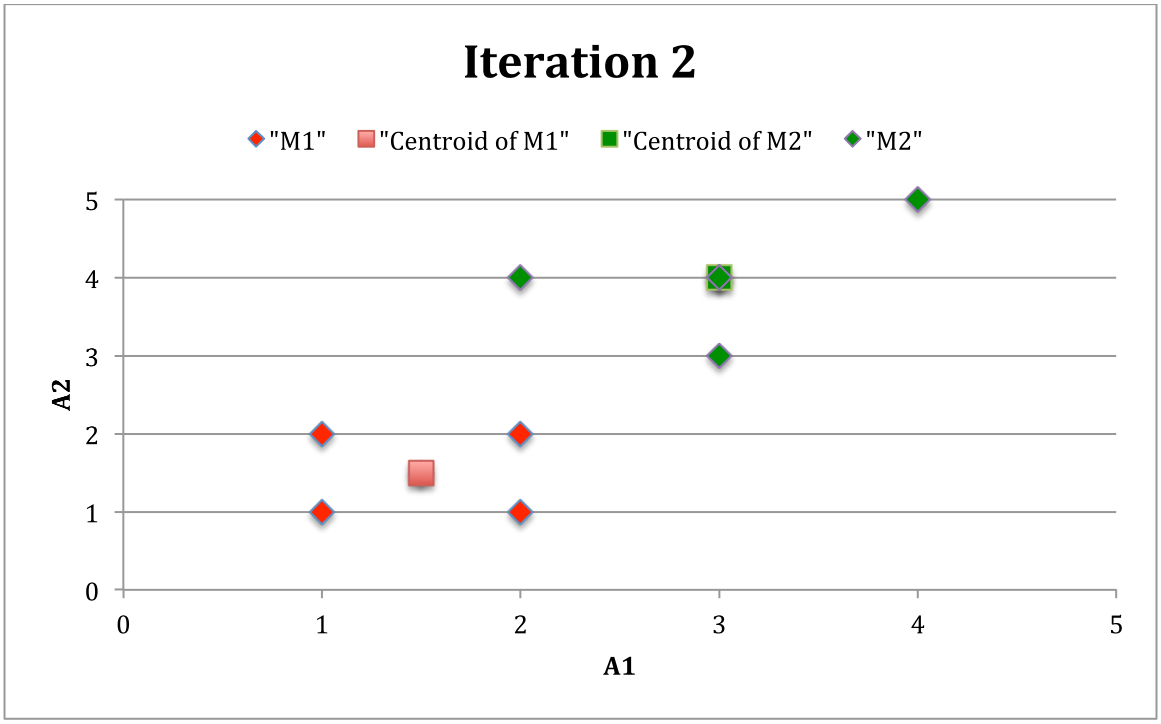
Iteration-1:

| ID | Distance from M1 | Distance from M12 | Decision |
|---|---|---|---|
| 1 | 2 | 2.24 | M1 |
| 2 | 2.234 | 2 | M2 |
| 3 | 1 | 1.41 | M1 |
| 4 | 1.41 | 1 | M2 |
| 5 | 2 | 1 | M2 |
| 6 | 2.24 | 1.414 | M2 |
| 7 | 3.61 | 2.83 | M2 |
| 8 | 1.41 | 1 | M2 |



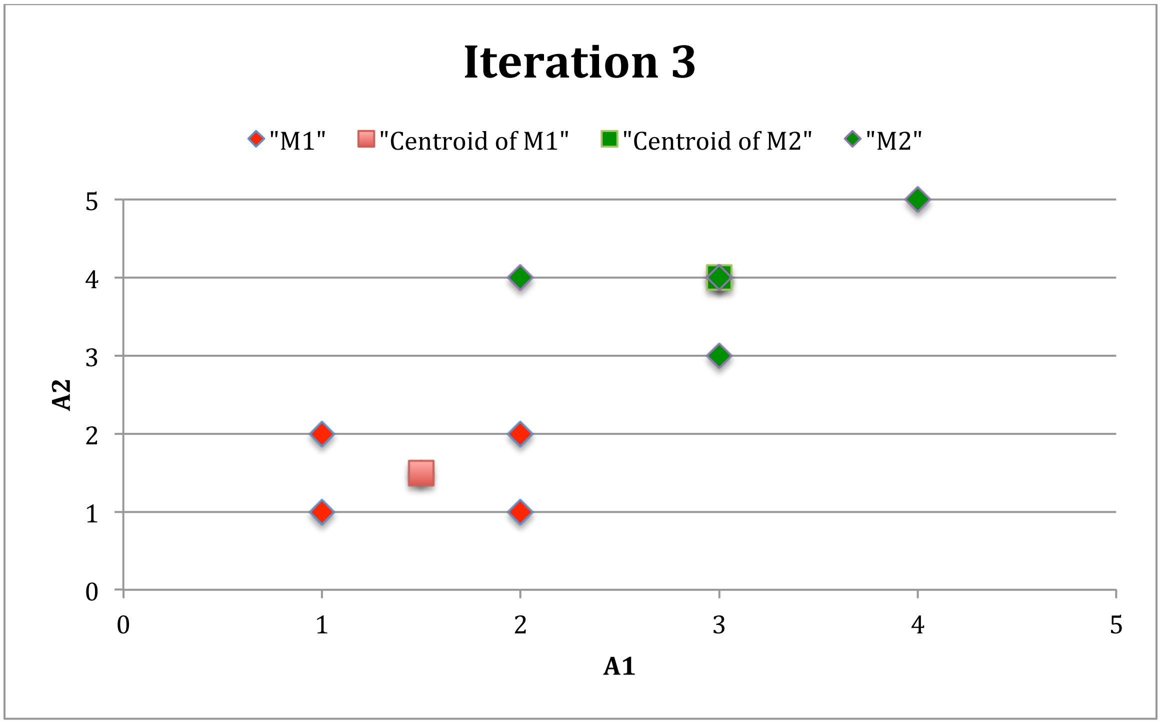
Iteration-2:

| ID | Distance from M1 | Distance from M2 | Decision |
|---|---|---|---|
| 1 | 0.5 | 2.74 | M1 |
| 2 | 1.19 | 2.27 | M1 |
| 3 | 0.5 | 2.04 | M1 |
| 4 | 1.19 | 1.35 | M1 |
| 5 | 2.50 | 0.37 | M2 |
| 6 | 3.20 | 0.89 | M2 |
| 7 | 4.61 | 2.26 | M2 |
| 8 | 2.69 | 1.07 | M2 |



Iteration-3:

| ID | Distance from M1 | Distance from M2 | Decision |
|---|---|---|---|
| 1 | 1.00 | 3.61 | M1 |
| 2 | 0.71 | 1.41 | M1 |
| 3 | 0.71 | 2.83 | M1 |
| 4 | 0.71 | 1.41 | M1 |
| 5 | 2.12 | 1.00 | M2 |
| 6 | 2.92 | 0.00 | M2 |
| 7 | 4.30 | 1.41 | M2 |
| 8 | 2.55 | 1.00 | M2 |



Answer to Q2:
Part-a

Step-0:

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 2.24 | 3.61 | 4.47 |
| 2 | 1 | 0 | 2.0 | 3.16 | 3.61 |
| 3 | 2.24 | 2.00 | 0 | 1.41 | 3.0 |
| 4 | 3.61 | 3.16 | 1.41 | 0 | 2.24 |
| 5 | 4.47 | 3.61 | 3.0 | 2.24 | 0 |

Step-1:

| | (1,2) | 3 | 4 | 5 |
|---|---|---|---|---|
| (1,2) | 0 | 2.0 | 3.16 | 3.61 |
| 3 | 2.00 | 0 | 1.41 | 3.0 |
| 4 | 3.16 | 1.41 | 0 | 2.24 |
| 5 | 3.61 | 3.0 | 2.24 | 0 |

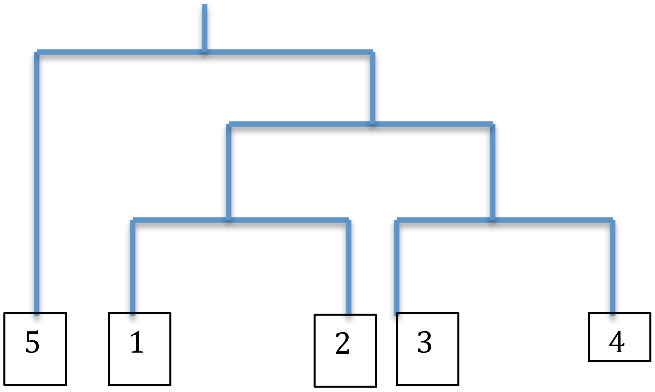
Step-2:

| | (1,2) | (3,4) | 5 |
|---|---|---|---|
| (1,2) | 0 | 2.0 | 3.61 |
| (3,4) | 2.00 | 0 | 2.24 |
| 5 | 3.61 | 2.24 | 0 |

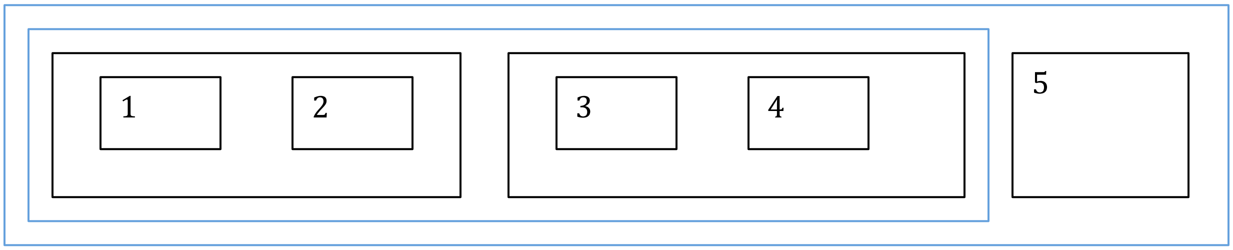
Step-3:

| | ((1,2), (3,4)) | 5 |
|---|---|---|
| ((1,2), (3,4)) | 0 | 2.24 |
| 5 | 2.24 | 0 |

Dendogram:



Nested Cluster



Part-b

Step-0:

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 2.24 | 3.61 | 4.47 |
| 2 | 1 | 0 | 2.0 | 3.16 | 3.61 |
| 3 | 2.24 | 2.00 | 0 | 1.41 | 3.0 |
| 4 | 3.61 | 3.16 | 1.41 | 0 | 2.24 |
| 5 | 4.47 | 3.61 | 3.0 | 2.24 | 0 |

Step-1:

| | (1,2) | 3 | 4 | 5 |
|---|---|---|---|---|
| (1,2) | 0 | 2.24 | 3.61 | 4.47 |
| 3 | 2.24 | 0 | 1.41 | 3.0 |
| 4 | 3.61 | 1.41 | 0 | 2.24 |
| 5 | 4.47 | 3.0 | 2.24 | 0 |

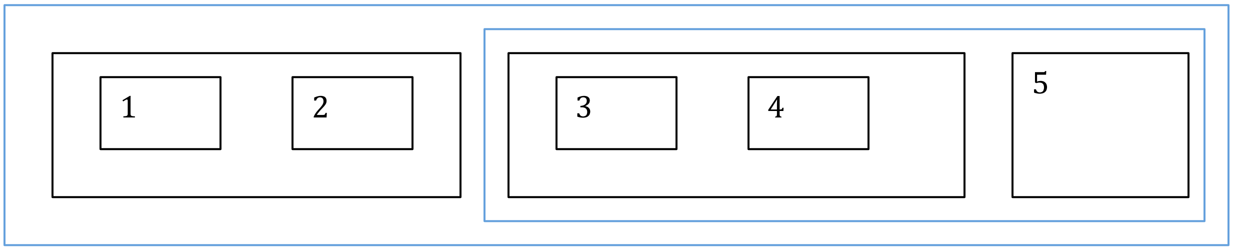
Step-2:

| | (1,2) | (3,4) | 5 |
|---|---|---|---|
| (1,2) | 0 | 3.61 | 4.47 |
| (3,4) | 3.61 | 0 | 3.0 |
| 5 | 4.47 | 3.0 | 0 |

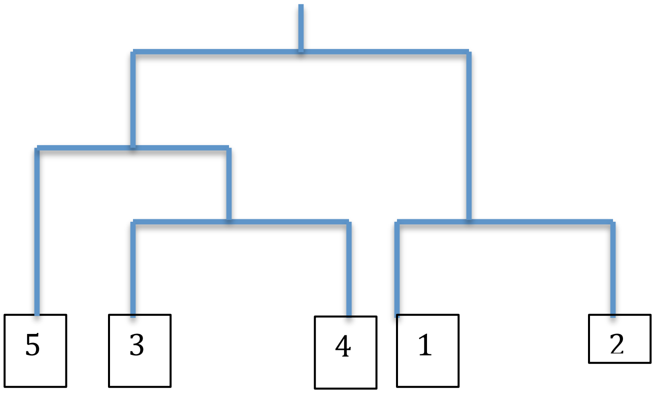
Step-3:

| | (1,2) | ((3,4), 5) |
|---|---|---|
| (1,2) | 0 | 4.47 |
| ((3,4), 5) | 4.47 | 0 |

Nested Cluster:



Dendogram:



Answer to Q3

a.

Core point:

The data point that has more than the number of min points(minPts) within the specified distance(Eps)

Boundary point:

The data point that is within the specified distance (Eps) of a core point

Noise point:

The data point that is neither a core point nor a boundary point

b.

Core points: 3

Boundary points: 2, 4, and 9

Noise points: 1, 5, 6, 7, 8, and 10

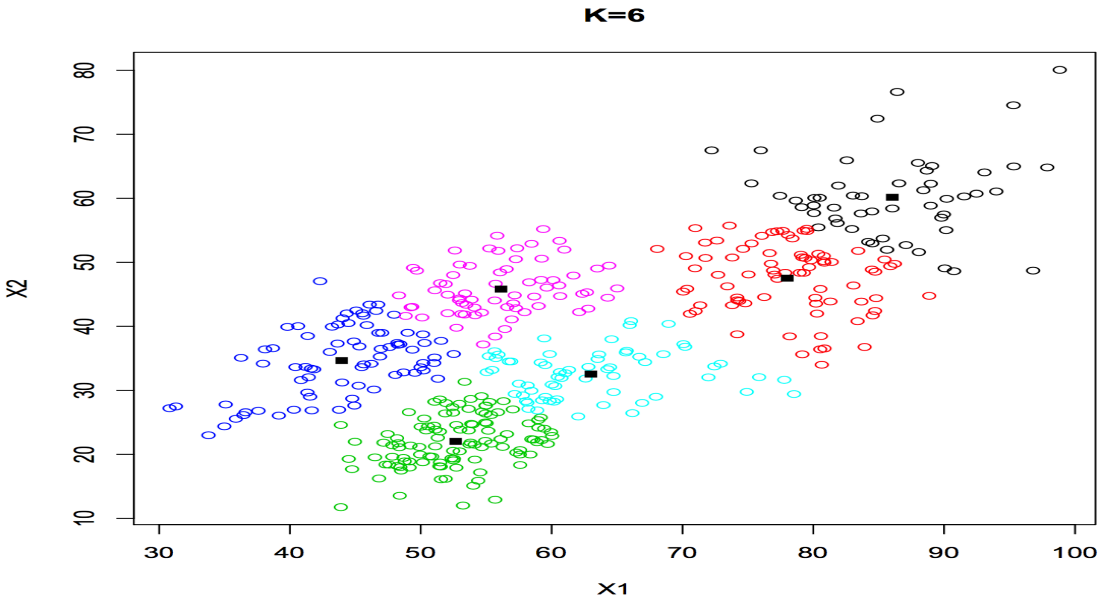
c.

- When dataset has high dimensional density
- When the clusters have varying densities e.g. one cluster has a lot of points, and the others have lesser data points

d.

DBSAC is deterministic in the sense that there wherever we start building clusters we end up finding the same clusters.

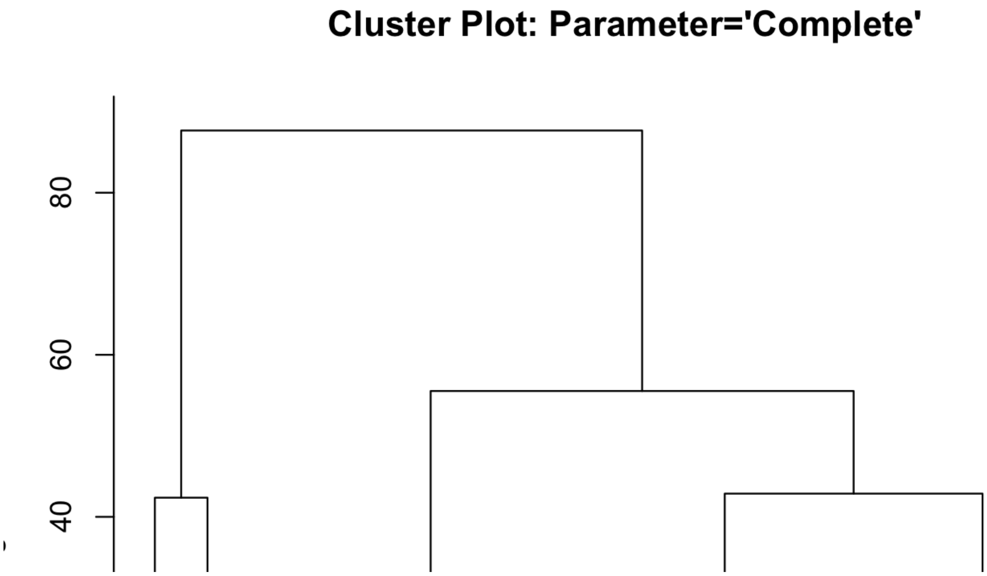
Answer to Q4:
(a)



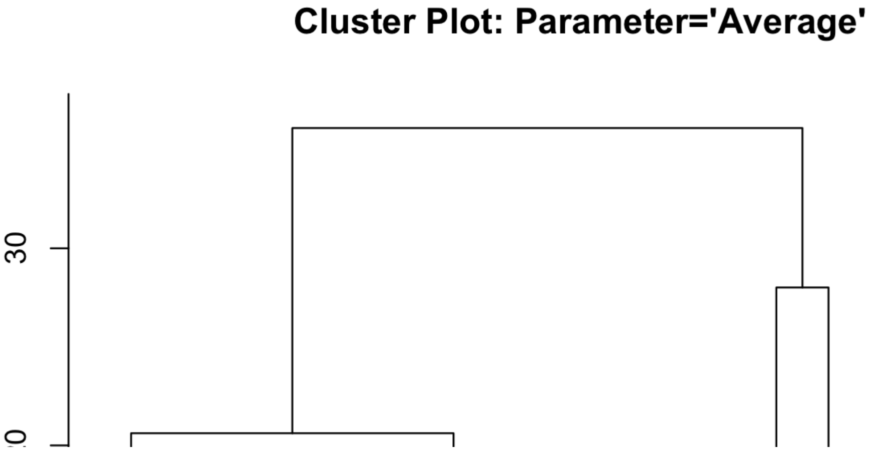
Source code:

```
temp = read.csv("hw2-data.csv", sep=",", row.names=1)
data_ <- as.matrix(temp)
#temp1[450,2]
print("Clustering started ... ")
cluster_no = 6
theCluster <- kmeans(data_,cluster_no)
#theCluster$cluster
plot(data_[,1], data_[,2],col=theCluster$cluster, xlab="X1",
 ylab="X2", main="K=6")
points(theCluster$centers, pch=15)
###
```

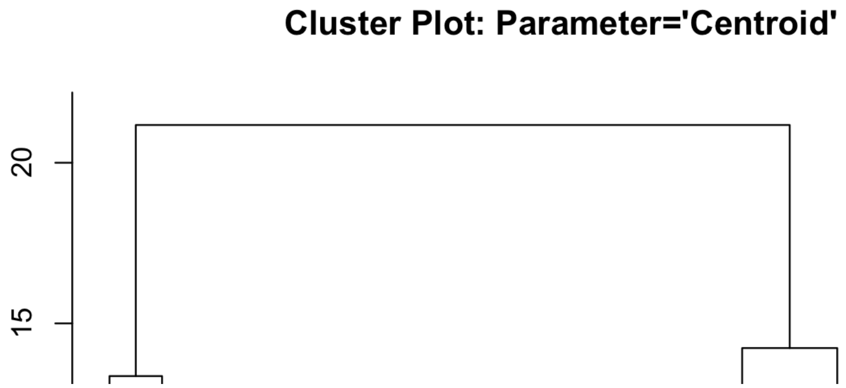
(b)
complete:



average:



centroid:



Source code:

```
temp = read.csv("hw2-data.csv", sep=",", row.names=1)
data_ <- as.matrix(temp)
print("Hierarchical Clustering started ... ")
hc_data_ <- hclust(dist(data_), method = "average")
groups <- cutree(hc_data_, h=20) # cut tree into 4 clusters
plot(hc_data_, main = "Cluster Plot: Parameter='Centroid'",
 labels = as.character(groups))
```