Robert F.K. Martin
ID 1505151
Csci 5523
30 Nov 2011
Homework 4

Problem 1:
1-itemsets(**frequent in bold**): <{**A**}>,<{**B**}>,<{**C**}>,<{**D**}>,<{E}>
2-itemsets: <{A} {A}>, <{A} {B}>, <{**A**} {**C**}> <{**A**} {**D**}>, <{A} {E}>
<{B} {A}>, <{B} {B}>, <{**B**} {**C**}>, <{**B**} {**D**}>, <{B} {E}>,
<{C} {A}>, <{C} {B}>, <{C} {C}>, <{**C**} {**D**}>, <{C} {E}>,
<{D} {A}>, <{D} {B}>, <{D} {C}>, <{D} {D}>, <{D} {E}>,
<{E} {A}>, <{E} {B}>, <{E} {C}>, <{E} {D}>, <{E} {E}>
3-itemsets: <{A} {C} {D}>, <{B} {C} {D}>
4-itemsets: nothing to generate

Problem 2a:
hierarchical, overlapping, partial

Problem 2b:
partitional, distinct, complete

Problem 2c:
partitional, overlapping, complete

Problem 3a:

Distance matrix generated from the 5 points:

	A	B	C	D	E
A	0	3.7417	3.1623	2.2361	1.4142
B	3.7417	0	2.00	5.9161	4.4721
C	3.1623	2.0	0	5.1962	4.0
D	2.2361	5.9161	5.1962	0	1.7321
E	1.4142	4.4721	4.0	1.7321	0

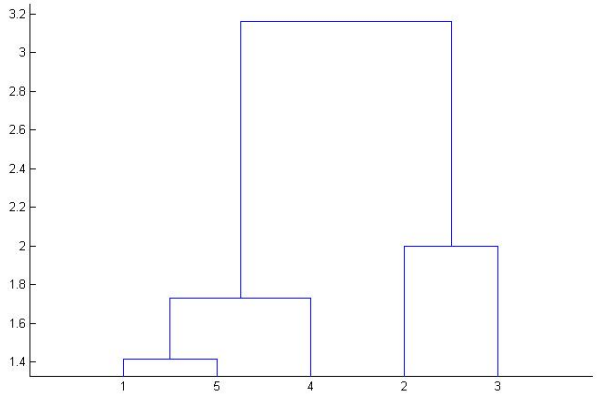

Figure 1: Matlab produced dendrogram from above data

Problem 3b:
Computing correlation requires the standard deviation of the two vectors to be computed. The vectors for d=(1,1,1) and e=(2,2,2) have a std of 0 which leads to an undefined value for correlation.

Problem 3c:

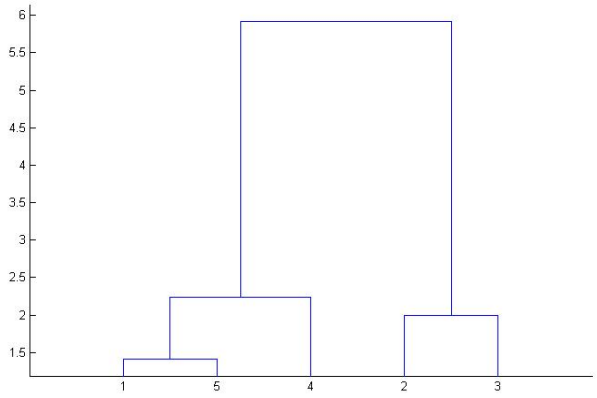

Figure 2: Matlab produced dendrogram from above data

Problem 4:
For eps=0.4, there will be one cluster, centered near point A. The cluster will include B, which is a border point. There are more than 4 points less than 0.4 from A and you can traverse to point B using points less than 0.4 from each other. But B only has 3 points within 0.4 of it so it is only a border point. Points C and D are noise. For eps=0.6, there will be two clusters. One cluster will be centered around C, the other will be somewhere in the vicinity of A and B, since both are core points are included with eps of each other. There is a possibility that C will also be part of the AB cluster, but it is hard to tell without actually checking each of the points in between. From what I can tell, they are not connected although it is probably very close.

eps	A	B	C	D
0.4	core	border	noise	noise
0.6	core	core	core	noise

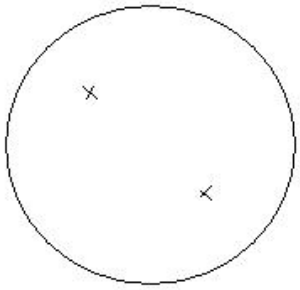

Figure 3: **Problem 5a:** The two cluster centers will be on opposite sides of the center probably along the same diameter.

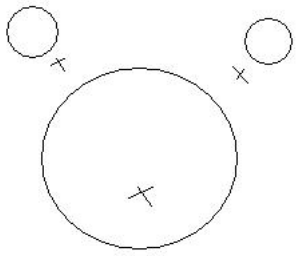

Figure 4: **Problem 5b:** The cluster centers will be as shown. Two will be between the smaller circles and the larger circle along a line joining their centers. The third will be somewhere on the lower half of the larger circle.

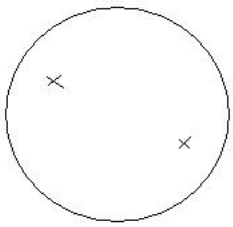

Figure 5: **Problem 5c:** The most likely solution will be the same as part a). However, if by chance, one of the initializing points is in the vicinity of the two far away points, then the two cluster centers may be at the centers of each of the circles.

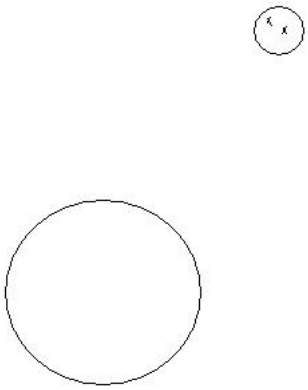

Figure 6: **Problem 5d:** The most likely solution is as shown with both cluster centers in the smaller, denser circle. However, if the algorithm is initialized with two points near the centers of each of the circles, the cluster centers may be the actual centers of each of the circles.

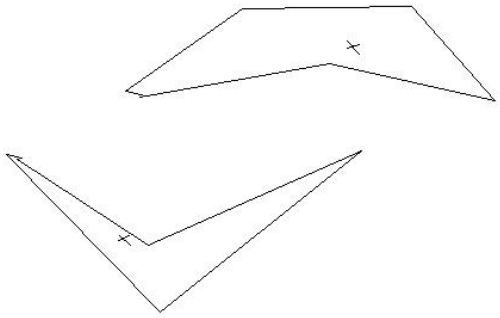

Figure 7: **Problem 5e:** The cluster centers will be slightly off-center of the crescents.

Problem 6a:
Single link will correctly return the three distinct clusters. Complete link will most likely break the larger circle into at least two regions and group the two smaller circles together.

Problem 6b:
single link will probably fail because of the noise and group all the points as one cluster. Complete would more likely be a better choice here because of the noise and it may sometimes group the two clusters separately.

Problem 6c:
Single link will find the correct clusters but complete link will probably include the nearer points of the two lower crescents with the upper crescent.

Problem 6d:
Single link will probably return the two clusters but complete link will include some of the closer points of the larger circle with the smaller circle.

Problem 7a:
As long as epsilon stays small enough that it can't cross the gap between the blobs, DBSCAN will still find two clusters.

Problem 7b:
Similar to a), if epsilon stays small enough that it can't include points from both circles, then it will still find two clusters.

Problem 7c:
DBSCAN will include some of the noise with both of the clusters. If the noise plus epsilon is great enough to bridge the gap between the two circles, then it will return a single cluster and possibly

make some distinct clusters in the noise. The epsilon required to create one cluster will be smaller than the epsilon from b) because of the noise.

Problem 7d:
DBSCAN will act the same way as in c). The density of the smaller circle will not have any effect on DBSCAN since we are only changing epsilon. Whether or not the two circles get clustered together has more to do with the noise density and location than the density of the individual circles.