HW #3

Question 1 (10 points):

Many partitional clustering algorithms that automatically determine the number of clusters claim that this is an advantage. List two situations in which this is not the case.

Answer: (a) When there is hierarchical structure in the data. Most algorithms that automatically determine the number of clusters are partitional, and thus, ignore the possibility of subclusters. (b) When clustering for utility. If a certain reduction in data size is needed, then it is necessary to specify how many clusters (cluster centroids) are produced.

Question 2 (20 points)

You are given a dataset with 100 records and are asked to cluster the data. You use K-means to cluster the data, but for all values for K, 1<= K <= 100, the K-means algorithm returns only one non-empty cluster. You then apply an incremental version of K-means, but obtain exactly the same result. How is this possible? How would single link or DBSCAN handle such data?

Answer: (a) The data consists completely of duplicates of one object. (b) Single link (and many of the other agglomerative hierarchical schemes) would produce a hierarchical clustering, but which points appear in which cluster would depend on the ordering of the points and the exact algorithm. However, if the dendrogram were plotted showing the proximity at which each object is merged, then it would be obvious that the data consisted of duplicates. DBSCAN would find that all points were core points connected to one another and produce a single cluster.

Question 3 (20 points)
Using the below data, compute the silhouette coefficient for each point, each of the two clusters, and the overall clustering.

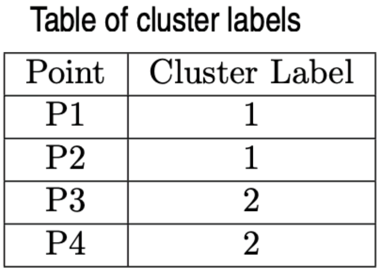
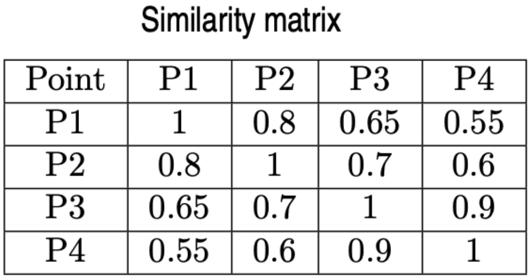


Table of cluster labels

Point	Cluster Label
P1	1
P2	1
P3	2
P4	2



Similarity matrix

Point	P1	P2	P3	P4
P1	1	0.8	0.65	0.55
P2	0.8	1	0.7	0.6
P3	0.65	0.7	1	0.9
P4	0.55	0.6	0.9	1

SC = (b-a)/ max(a,b)

Point p1:
a = distance(p1 to p2) = 1-0.8 = 0.2
b = min (average distance of p1 to p3 and p4) = (0.35+0.45)/2=0.4
SC = (0.4-0.2)/0.4 =0.5

Point p2:
a = 0.2
b = min (average distance of p2 to p3 and p4) = (0.3+0.4)/2 = 0.35
SC = (0.35-0.2)/0.35 = 0.43

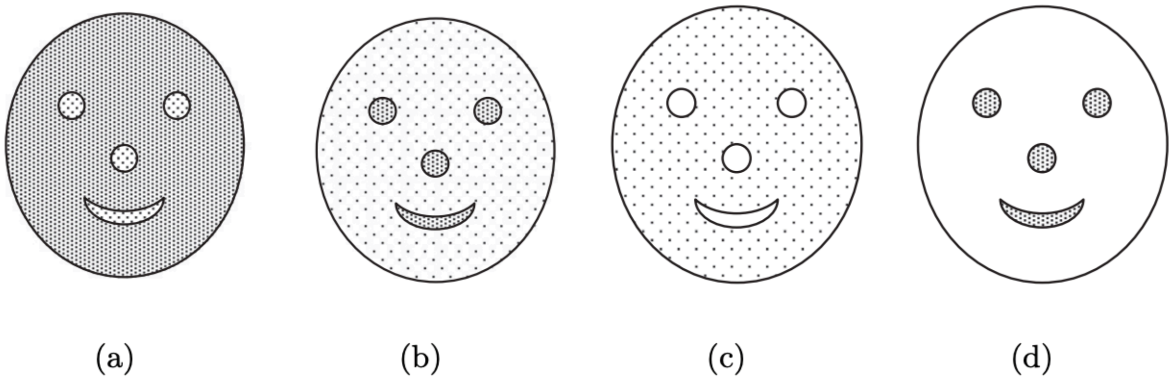
Point p3:
a = distance (p3 to p4) = 0.1
b = min (average distance of p3 to p1 and p2) = (0.35+0.3)/2= 0.32
SC = (0.32-0.1)/0.32 = 0.69

Point p4:
a = 0.1
b = min(average distance of p4 to p1 and p2) = (0.45+0.4)/2 = 0.425
SC = 0.325/0.425 = 0.76

Cluster 1 average SC = (0.5+0.43)/2 = 0.46
Cluster 2 average SC = (0.69+0.76)/2 = 0.725
Overall average SC = (0.46+0.725)/2 = 0.5925

Question 4 (20 points)
Given the below four faces, darkness or number of dots represents density. Lines are used only to distinguish regions and do not represent points.

For each figure, could you use partition, hierarchical, density, and other algorithms we learned in class to find the patterns represented by the nose, eyes, and mouth? Please list at least 3 different types of algorithms and explain the pros and cons of each.



Partition: k-mean: only for b and d. the points in the nose, eyes, and mouth are much closer together than the points between these areas. For (d) there is only space between these regions.

Hierarchical: single link / min: only for b and d. K-means would find the nose, eyes, and mouth, but the lower density points would also be included. For (d), K- means would find the nose, eyes, and mouth straightforwardly as long as the number of clusters was set to 4.

Density: DBSCAN: all

Question 5 (30 points)

You are given two sets of 100 points that fall within the unit square. One set of points is arranged so that the points are uniformly spaced. The other set of points is generated from a uniform distribution over the unit square.

(a). Is there a difference between the two sets of points?

(b). If so, which set of points will typically have a smaller SSE for K = 10 clusters?

(c). What will be the behavior of DBSCAN on the uniform dataset? The random dataset?

Answer:

(a) Yes. The random points will have regions of lesser or greater density, while the uniformly distributed points will, of course, have uniform density throughout the unit square.

(b) The random set of points will have a lower SSE

(c) DBSCAN will merge all points in the uniform data set into one cluster or classify them all as noise, depending on the threshold. There might be some boundary issues for points at the edge of the region. However, DBSCAN can often find clusters in the random data, since it does have some variation in density.