7&8

Cluster Analysis

Contents

True-False and Short Answer

1. For the following questions, give an answer and a short (1 or 2 sentences) explanation. For the rest of this question, "agglomerative hierarchical clustering" refers to procedures such as single link, complete link, and group average, while "k-means clustering" refers to k-means with random initialization of centroids and Euclidean distance.

   a) Agglomerative hierarchical clustering procedures are better able to handle outliers than k-means.
   True. When using agglomerative clustering, outliers will tend to remain as singleton clusters (or small clusters) that do not merge with other clusters until high in the tree. This makes it easy for agglomerative clustering techniques to identify and eliminate outliers. The standard K-means clustering algorithm, by contrast, assigns each outlier to some cluster, and thus can distort the centroid of the cluster.

   b) For any given data set, different runs of k-means can produce different clusterings, but agglomerative hierarchical clustering procedures will always produce the same clustering.
   True. There is no random element in the algorithms for agglomerative hierarchical techniques unless there are ties in the proximity values.

   c) K-means take less time and memory than agglomerative hierarchical clustering and is the most efficient clustering algorithm possible.
   False. Although k-means is more computationally efficient than agglomerative hierarchical clustering, there are more efficient algorithms possible, e.g., the leader algorithm. (See Exercise 12, Chapter 7.)

   d) During a post-processing step for K-means, a cluster is split by picking one of the points of the cluster as a new centroid and then reassigning the points in the cluster either to the original centroid or the new centroid. What happens to the SSE of the clustering?
   Splitting decreases SSE because we have two centroids for the same set of points, which will reduce the distance of points to the nearest centroid.

   e) When clustering a dataset using K-means, whenever SSE decreases, cohesion increases.
   True. For K-means, SSE is an inverse measure of the cohesion of clusters, and thus, as SSE decreases, cohesion increases and vice-versa.

   f) When clustering a dataset using K-means, whenever SSB (the between sum of squares) increases, separation increases.
   True. For K-means SSB (the between sum of squares) is a direct measure of the separation of clusters, and thus, as SSB increases, separation increases and vice-versa.

   g) Cohesion and separation are independent for K-Means, i.e., improving cohesion (smaller SSE) doesn't necessarily improve separation (larger between sum of squares (SSB)).
   False. SSE and SSB are related because the total sum of squares (TSS) is a constant and TSS = SSE + SSB. So, improving cohesion (reducing SSE) will also improve separation (increasing SSB, and vice-versa. See the book, page 577.

   h) When clustering a dataset using K-means, SSE + BSS is a constant.
   True. For K-means, the total sum of squares (TSS) is the sum of SSE (or within cluster sum of squares) and the between sum of squares (SSB), TSS is constant during the K-means clustering process. See the book, page 577

i) When clustering a dataset using K-means, whenever cohesion increases, separation increases.
True. For K-means, the total sum of squares (TSS) is the sum of SSE (the within cluster sum of squares) and the between sum of squares (SSB). Note TSS is constant at every step of the K-means clustering process. See the book, page 577. SSE is an inverse measure of cluster cohesion, while SSB is a direct measure of cluster separation. Thus, as cohesion increases, SSE decreases, and SSB (separation) increases since TSS = SSE + SSB is a constant. When SSE is at a local minima, BSS is at a local maxima.

j) K-means only optimizes cluster cohesion (SSE). Hence, improving cohesion (smaller SSE) doesn't necessarily improve cluster separation.
False. For K-means, the total sum of squares (TSS) is the sum of SSE (the within cluster sum of squares) and the between sum of squares (SSB). Note TSS is constant at every step of the K-means clustering process. See the book, page 577. SSE is an inverse measure of cluster cohesion, while SSB is a direct measure of cluster separation. Thus, as cohesion increases, SSE decreases, and SSB (separation) increases since TSS = SSE + SSB is a constant.

k) When clustering a dataset using K-means, SSE is guaranteed to monotonically decrease as the number of clusters increases.
False. SSE will generally decrease as k, the number of clusters, increases, but because of random initialization and local minima, it can happen that the SSE for a particular run of K-means with a particular k will be lower than for some other run of K-means with a smaller value of k.

l) Hierarchical clustering approaches can only handle low-dimensional data.
False. Hierarchical clustering algorithms use a proximity matrix. Thus, if a given similarity or dissimilarity is appropriate for high dimensions, then they can cluster high dimensional data.

m) Agglomerative hierarchical clustering doesn't optimize any global objective function.
True. The algorithms operate in a greedy manner.

n) For density-based clusters, it is difficult to define a natural measurement of cohesion.
True, since the natural way to define a measure of cohesion would have a low value if all the points of a cluster are relatively farther from points in other clusters than from points in their cluster, but this is often not the case for density-based clusters.

o) The silhouette coefficient is a good evaluation measure for density-based clusters.
False. The silhouette coefficient combines measures of cohesion and separation, but these measures may not work well for density-based clustering. For example, the density-based clusters found by DBSCAN can have widely different densities, i.e., different cohesions and may be intertwined, i.e., poor separation.

p) Gaussian mixture model clustering cannot handle varying densities.
False. GMM clustering can even handle embedded clusters (where one cluster sits inside another), where one cluster is denser than another.

q) The clusters for Gaussian mixture model clustering are fully described by their cluster centroids.
False. A full description of GMM clusters also needs to include the covariance matrix of each cluster.

r) Like SNN and DBSCAN, Chameleon is a partitional clustering algorithm.
False. Chameleon is a hierarchical clustering scheme.

Note: A hierarchical clustering does produce a partitional clustering at any given level. However, SNN and DBSCAN produce only a single partitional clustering, and should not be confused with a hierarchical clustering, which is a sequence of nested partitional clusterings.

s) Shared Nearest-Neighbor Clustering is a complete clustering algorithm
False. SNN clustering discards the noise points.

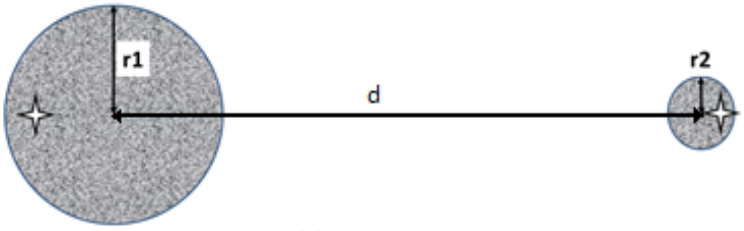
t) On high dimensional data, DBSCAN is likely to work better than Shared Nearest-Neighbor Clustering.
False, density is not defined well in higher dimensions.

2. To answer the following true/false question about how k-means operates, refer to figure (a) below. Note that we are referring to the very basic k-means algorithm presented in class and not to any of it more sophisticated variants, such as bisecting k-means or k-means++.

Note that for the figure, the initial centroids are given by the symbol: 

For figure (a), assume the shaded areas represent points with the same uniform density.

True or False: For Figure (a) and the given initial centroid: When the k-means algorithm completes, each shaded circle will have one cluster centroid at its center.



(a) d >> r1 >> r2

True. The clusters are too far away for one centroid to attract points from another.

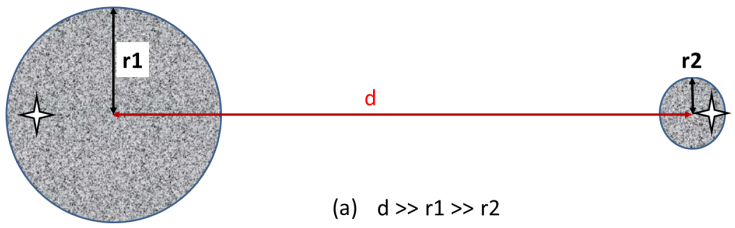
Instructor Note: We can easily make multiple versions of this question by placing stars in different initial locations. For example, place them both in the large cluster or both in the small cluster. The final answer will depend on the relative difference in the number of points in these clusters and the distance between them. For example, for the current setup, if both initial points are in the left circle, then if d is small enough, both centroids will stay in the larger cluster.

3. To answer the following true/false questions about how k-means operates, refer to figures (a), (b), and (c), below. Note that we are referring to the very basic k-means algorithm presented in class and not to any of it more sophisticated variants, such as bisecting k-means or k-means++.

Note that for all three figures, the initial centroids are given by the symbol: 

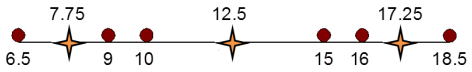
For figures (a) and (b), assume the shaded areas represent points with the same uniform density. For Figure (c), the data points are given as red dots, and their values are indicated under the dots. No explanation for your answer is necessary unless you feel there is some ambiguity in the figure or the

question.



(a) d >> r1 >> r2

(b)



(c)

a) **True or False:** For Figure (a) and the given initial centroid: When the k-means algorithm completes, each shaded circle will have one cluster centroid at its center.

Answer: True. The clusters are too far away for one centroid to attract points from another.

b) **True or False:** For Figure (b) and the given initial centroids: When the k-means algorithms completes, there will be one cluster centroid in the center of each of the two shaded regions, and each of the two final clusters will consist only of points from one of the shaded regions. In other words, none of the two final clusters will have points from both shaded regions.

Answer: False. The final clusters will have points from both of the two shaded regions since they are close to each other and not of circular shape.

c) **True or False:** For Figure (c) and the given initial centroids, the final clustering for k-means contains an empty cluster.

Answer. True. The centroid at 12.5 is farther away from all points than any other clusters and will become empty.

4. What are two key differences and two key similarities between clustering and association analysis?

   Similarities:

   Clustering and association analysis are both unsupervised methods.
   Both produce groupings of objects. (The items that support a frequent itemset.)

   Differences:

   The main focus of association analysis is on rules.

   For clustering, grouping of objects are based on similarity, which typically considers all attributes. In contrast, objects (transactions) are grouped into frequent itemsets based on a relatively small number of attributes. Thus, clustering finds global patterns, while association analysis finds local patterns.

Cluster Types

1. For each of the described data sets, decide what type of clustering should be used (hierarchical or partitional, exclusive or overlapping or fuzzy, complete or partial (incomplete)). Briefly explain your reasoning if you feel there may be several possible answers. **Note:** we are using partitional and hierarchical in the more relaxed use of the terms to mean un-nested or nested, respectively.

An example: Clustering library books based on their literary genre. The genre/topic can have several subtopics, as well.

Answer: hierarchical, overlapping, complete

a) Proteins perform different biological functions that are organized into a hierarchical taxonomy (GO) defined by biologists. Some proteins can be multi-functional as well. You want to group them based on those functions. Some proteins may also be missing functional annotation.

Hierarchical, overlapping, partial

b) A nutritionist asks you several questions (e.g., your calorie intake, types of food you eat, your physical activity labels, and so on) to assess your risks for diabetes in three different groups: low, medium, and high.

Partitional, exclusive, complete

c) An international grad student can work on campus only at most for 20 hours. You want to assign each student to different job categories (e.g., TA, RA, another on-campus job, jobless). Hint: the sum of these categories should sum up to 20 hours.

Partitional, fuzzy, complete

d) Grouping of students in a university-based on the organization (department, college, institute, etc.) to which they belong. A student may belong to multiple organizations. Also, some students don't have declared majors and hence may not belong to any organization.

Hierarchical, overlapping, partial.

e) Grouping of all the students in the Computer Science department based on the letter grade they get in the data mining (CSci 5523) class.

Partitional, exclusive, partial (some students in the CS dept wouldn't have taken the DM class and thus can't be grouped).

f) The objects are the registered students in a class. There are groups for each official grade that students received for the class. We want to cluster the students in those groups.

partitional, exclusive, complete

g) The data objects are the applicants to a college. The admissions personnel view the students as belonging to two groups: those that will be accepted and those that will be rejected.

partitional, exclusive, complete

h) The objects are the applicants to a college. Each applicant is assigned a score from 0 to 10, indicating the likelihood/desirability of their admission. Even before any decisions have been made, the admissions personnel view the students as belonging to two groups: those that will be accepted and those that will be rejected.

Partitional, fuzzy, complete (the score of every applicant makes a soft assignment to the two groups, and hence the clustering is fuzzy).

i) Each group is the set of all people in the military who have the same rank.

partitional, exclusive, complete
You can answer partial if you take the view that the set of objects was the entire population (even those outside military).

j) Grouping of employees in an organization based on the position that a person holds according to the organization's job classification. A person can hold multiple positions.

Partitional, overlapping, complete. If we consider that the total effort for each position held by a person in an organization sums to 100%, then this can be considered as fuzzy as well. If we consider that the job classification may be hierarchical, the clustering can be hierarchical as well.

k) A supermarket manager wants to group all goods into several categories, each of which has multiple subcategories.

hierarchical, exclusive, complete.

l) Now the supermarket manager wants to group all goods according to brand.

partitional, exclusive, complete

m) You want to group all the videos on YouTube into several genres/topics, each of which can have several subtopics.

hierarchical, overlapping, complete (can be partial if a video may belong to no existing category)

n) The objects are movies, and the groups are the types of movie: action, drama, kids, science-fiction, etc.

partitional, overlapping, complete (can be partial if a movie may belong to no existing category)

o) Grouping of movie actors based on the genres of the movies (comedy, drama, sci-fi, etc.) in which they have acted.

Partitional, overlapping (most actors have done movies of multiple genres), complete. (can be partial if a movie belongs to no existing category)

p) You want to group all locations on Earth as to whether they belong to a tropical rainforest, a deciduous forest, or an evergreen forest. Here, each location corresponds to a grid cell of surface area 1km by 1km, and a location can have more than one variety of forest.

partitional, overlapping, partial

q) Grouping of cities based on the various locations, namely county (local region), state or province, and country.

Hierarchical, exclusive, complete (overlapping can also be a correct answer since cities can fall in more than one county)

r) The objects are items, and the groups are the lattice of all items, i.e., the set of all subsets of the items.

hierarchical, overlapping, complete

s) The groups are all possible sets of 10 items drawn from a set of 100 items.

partitional, overlapping, complete

2. You are given the task of grouping the users of a movie recommendation website, e.g., Netflix, based on whether they like a particular comedy movie from a set of 10 comedy movies in the movie database. Every group consists of users that like a common comedy movie, and hence, there are 10 groups of users, each corresponding to one of the 10 comedy movies. Describe the characteristics of this grouping as to whether it is overlapping or exclusive or fuzzy, and complete or partial (incomplete) in the table below. If you feel there may be ambiguity, provide a short justification for your answer in the optional additional comments section of the table.

	Characteristic of the Grouping	Optional Additional Comments
Overlapping or non-overlapping	Overlapping	A user can like multiple comedy movies and thus belong to multiple groups.
Complete or partial	Partial	There may be users that do not like any of the 10 comedy movies and hence will not belong to any of the groups.

3. Short answer clustering questions.

a. A clustering algorithm A produces partial clusterings (i.e., A does not cluster all points). For what kind of problems is this feature of algorithm A particularly useful?

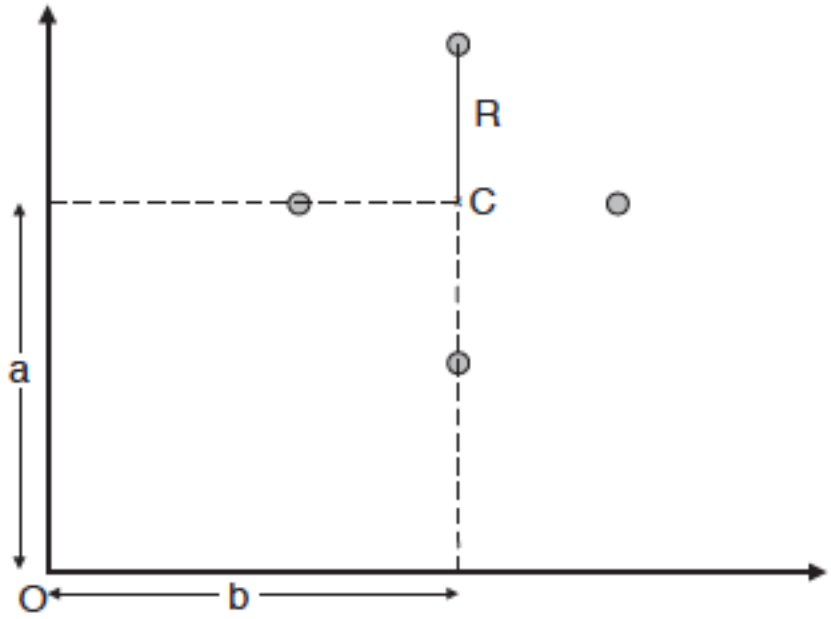
Clustering algorithms that produce partial clusterings are particularly useful in situations where the data set has noise and outliers since the clustering algorithm can discard such points.

b. The objects of a clustering problem are stocks listed on the New York Stock Exchange. Each stock is represented by a time series of its daily closing price. Stocks with similar market behavior (i.e., similar time series) are required to be in the same group. Which type of clustering is more natural for this problem: fuzzy or exclusive or overlapping? Briefly justify your answer.

Stocks of companies in similar businesses tend to have similar market behavior, and some companies are in multiple businesses. Therefore, fuzzy clustering is more natural for this problem.

K-means

1. Consider the 4 data points shown in the following Figure. The distance between each data point to the center C is R.



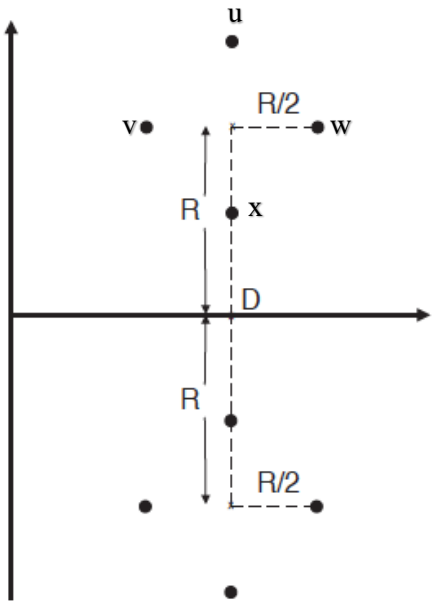
Example of 4 data points in 2-dimensional space.

a) Compute the total SSE of the data points to the centroid, C.
4R^2

b) Compute the total SSE of the data points to the origin, O.

4(a^2 + b^2 + R^2)

c) Using parts (a) and (b), compute the SSE for the 8 data points shown below with respect to the centroid, D. Note that points u, v, w, and x lie on a circle of radius R/2. Also, the figure is symmetric with respect to the horizontal line running through D.

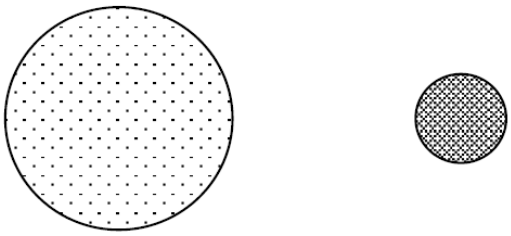


Example of 8 data points in 2-dimensional space.

using part (a), the SSE of u, v, w, and x with respect to their midpoint is $4(R/2)^2$.
From part (b), with a=R and b=0, the SSE of u, v, w, and x with respect to D is
$4(R^2 + 0^2 + (R/2)^{2)} = 5R^2$.
By symmetry, the SSE of the eight points with respect to D is 10*R^2.

2. K-means. Consider the following set of two-dimensional points. Darker areas indicate higher density. Assume a uniform density within each shaded area and that each circle has the same number of points. Also, assume that we are using the squared error objective function.

   We will distribute 5 centroids among the two circles as indicated in the problems below. Describe whether the centroids will move and how many centroids will end up in each circle.



a) All of the initial centroids are located in one of the circles.
At least one of them will move to the other circle.

b) At least one centroid in each circle.
The centroids will not move from one circle to another.

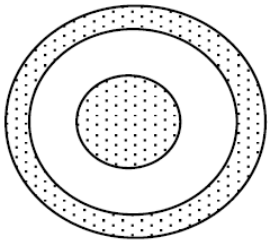
c) Now assume that the number of points in the larger circle is far greater than the number of points in the smaller circle.

Depending on the ratio of points, now the larger cluster may end up with all of the centroids

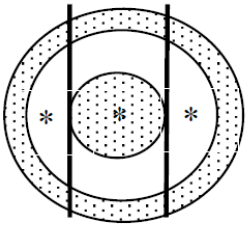
Instructor's note: this problem can be used to create multiple problems by specifying the positions of initial centroids, changing the number of points in each cluster, and specifying the positions of initial centroids.

3. K-means. For the following sets of two-dimensional points, (1) provide a sketch of how they would be split into clusters by K-means for the given number of clusters and (2) indicate approximately where the resulting centroids would be. Assume that we are using the squared error objective function. If you believe that there is more than one possible solution, then please indicate whether each solution is a global or local minimum. Darker areas indicate higher density. Assume a uniform density within each shaded area.

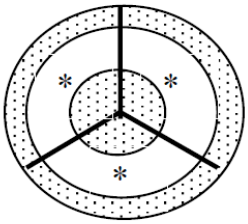
(a) k = 3.



Solution: * denotes the centroid of each cluster and lines represent boundaries of each cluster

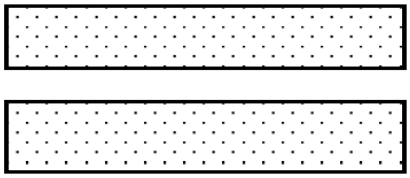


Local minima
(Centroids aligned in 1D-space)

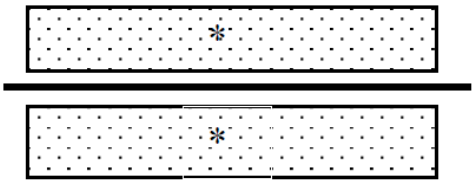


Global minimum
(Centroids aligned in 2D-space)

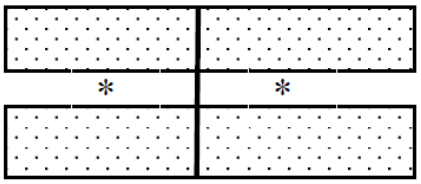
(b) k = 2



Solution:

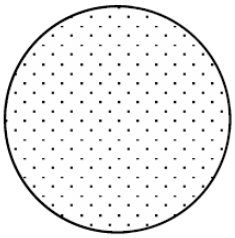


Local minima
(The clusters are non-globular)

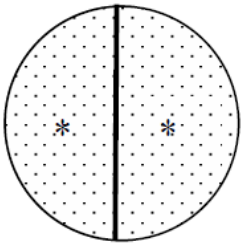


Global minimum
(The clusters are globular)

(c) k = 2



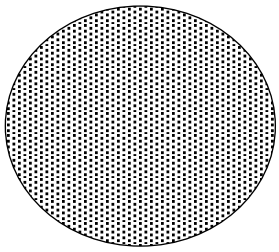
Solution:



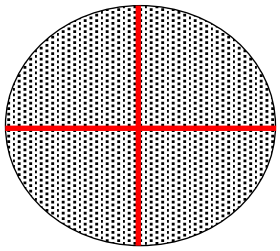
Or some rotation of this solution.

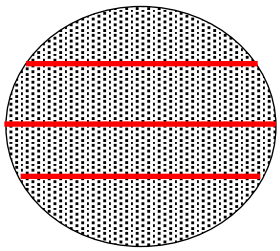
4. For the following sets of two-dimensional points, (1) draw a sketch of how they would be split into clusters by K-means for the given number of clusters and (2) indicate approximately where the resulting centroids would be. Assume that the minimization of the sum of squares error (SSE) is being considered as the optimization criterion. If you think that there can be multiple solutions for clustering depending on the initial choice of centroids, then please indicate whether each solution is a global or local minimum. Assume that the shaded regions in each of the figures have a uniform density of points.

   a) Use K = 4 on the dataset described in the figure below. How many possible ways are there (in theory) to partition this dataset into 4 clusters using the K-means algorithm?



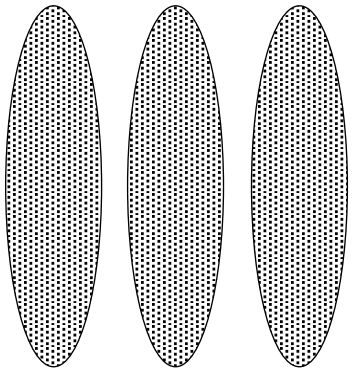
The circle will be split into 4 equal pieces. Think of cutting a pie into 4 slices. This will be the global minima. But there can be a local minima for which all centroids are on a diameter.



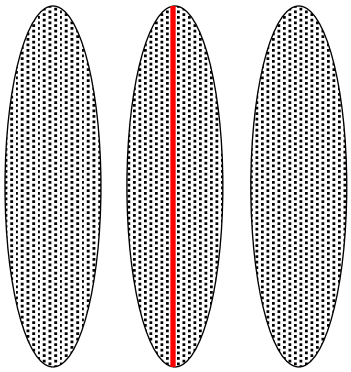


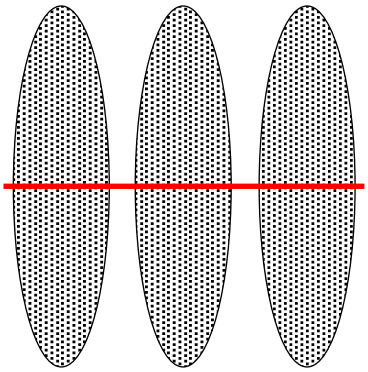
Any rotation of the above is also a solution.

b) Use K = 2 on the dataset described in the figure below.

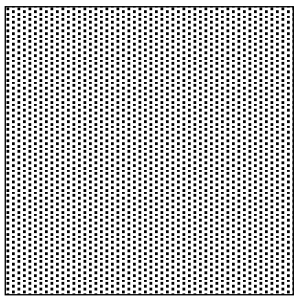


There are two possible solutions. In one, the two centroids are located in the middle of the figure and lie between ellipses 1 and 2 and 2 and 3, respectively. In the second solution, one centroid lies in the middle ellipse and is located about 1/3 of the distance from the top, centered horizontally in the ellipse. The second centroid is symmetrically placed. Which configuration is a local or global minimum will depend on the exact dimensions of the ellipses.

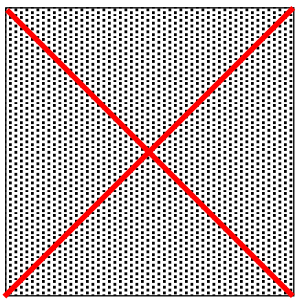


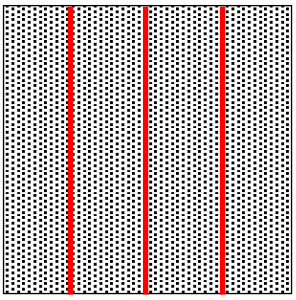


c) Use K = 4 on the dataset described in the figure below.



The square will most likely be split into four wedges. Imagine a square pie split across the diagonals. **The square could be split into four equal layers, either horizontally or vertically.** The square could also be split into four wedges. Imagine a square pie split across the diagonals. The wedges would give the globally minimal SSE since they are more compact than the wedges, while the horizontal or vertical layer split would give be a local minima.





5. Traditional K-means has a number of limitations, such as sensitivity to outliers and difficulty in handling clusters of different sizes and densities, or with non-globular shapes. Comment on the ability of fuzzy c-means to handle these situations.

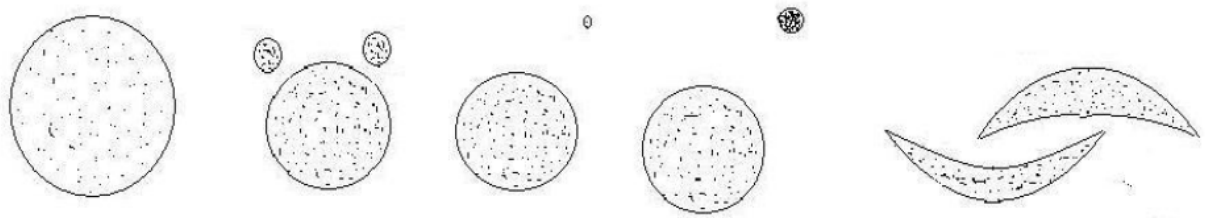
Answer: The fuzzy c-means algorithm is based on an objective function that is very similar to that of k-means. As such, fuzzy c-means has all the limitations of traditional K-means, except that it does not make

a hard assignment of an object to a cluster. More specifically, it tends to favor globular clusters, i.e., doesn't handle different shapes well, prefers clusters of the same size, i.e., tends to split bigger clusters, and is sensitive to differences in density since a small number of low-density points can have a higher SSE than a large number of points in a dense region.

6. Suppose you want to group videos on YouTube. Remember that videos are posted on YouTube continuously every day. You modify the standard k-means algorithm such that it can be applied in an online way. In particular, you allow the centroids to be updated incrementally after each assignment of a point to a cluster. Describe two advantages and one disadvantage of this online version as compared to standard k-means.

   One advantage is that you will never get an empty cluster. Another advantage is that it is memory efficient since you do not have to store all the points in memory. Disadvantages: the order of the videos may affect cluster assignments. Also, it is somewhat more computationally intensive since you are recomputing a cluster centroid for each assignment.

7. For the following sets of two-dimensional points, (1) draw a sketch of how they would be split into clusters by K-means for the given number of clusters and (2) indicate approximately where the resulting centroids would be. Assume that we are using the squared error objective function. If you think that there is more than one possible solution, then please indicate whether each solution is a global or local minimum. Note that the label of each diagram in Figure 1 matches the corresponding part of this question, e.g., Figure 1(a) goes with part (a). Assume that the points in the cluster are of equal density unless explicitly mentioned.

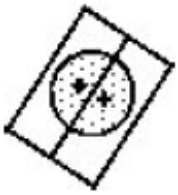


a) b) c) d) e)

a) K=2. How many possible ways are there (in theory) to partition the points into two clusters?

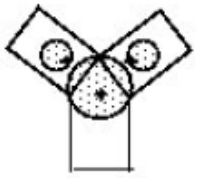
What can you say about the positions of the two centroids? (Again, you don't need to provide exact centroid locations, just a qualitative description.)

In theory, there are a very large number of ways to split the circle into two clusters - just take any line that bisects the circle. This line can make any angle between [0◦, 180◦] with the x-axis. The centroids will lie on the perpendicular bisector of the line that splits the circle into two clusters and will be symmetrically positioned. All these solutions will have the same, globally minimal, error.

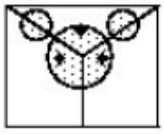


b) K= 3. Hint: Use the symmetry of the situation

For the solution shown in the top figure, the two top clusters are enclosed in two boxes, while the third cluster is enclosed by the regions defined by a triangle and a rectangle. (The two smaller clusters in the drawing are supposed to be symmetrical.) I believe that the second solution— suggested by a student—is also possible, although it is a local minimum and might rarely be seen in practice for this configuration of points. Note that while the two pie-shaped cuts out of the larger circle are shown as meeting at a point, this is not necessarily the case—it depends on the exact positions and sizes of the circles. There could be a gap between the two pie-shaped cuts, which is filled by the third (larger) cluster. (Imagine the small circles on opposite sides.) Or the boundary between the two pie-shaped cuts could be a line segment.



Global minimum



Local minimum

c) K = 2. Here is an outlier with distance d from the center of the circle and d > 2R, where R is the radius of the circle.

There will be two clusters, one having a centroid in the original circle and another with a centroid which may be in the original circle or between the outlier and the circle if the larger circle does not have too many points and/or the outlier is much farther away than 2R.

d) K = 2. There are two circles. The bigger circle has only 100 points, while the smaller circle is much denser with 100,000 points. The distance between the two centers is greater than 2R, where

R is the radius of the larger circle. Hint: the two centroids are more likely to be initialized with the points of the denser circle.

The denser circle will be divided into two clusters. One cluster will also contain the larger circle, and its centroid will be slightly shifted towards the larger circle.

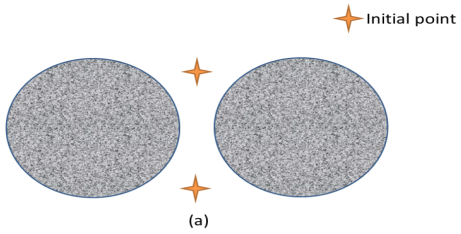
e) K = 2.

Since K-Means only can find globular clusters, it will split each ellipse between the two clusters.

8. To answer the following questions, refer to Figure (a) below. Assume the shaded areas represent points with the same uniform density. Note that the two initial centroids are indicated by the symbol: Initial point

Assume that after k-means runs, the final centroids are the same as the initial centroids.

1. Will this final clustering be a local or a global minimum with respect to SSE?
   local
2. Draw the boundary between the two clusters.
   Boundary: horizontal straight line halfway between the two centroids



9. In the figures below, assume that all the three circles have 50 points each that are uniformly distributed inside the circles. Also, assume that their sizes and distances between them are to scale. The 'X's in Figures 1 and 2 show two possible sets of centroid initializations for k-means for finding 3 clusters.

   Instructor note: By changing the number of points in each circle and the spacing and orientation of the circles, many similar problems can be created.

Fig.1

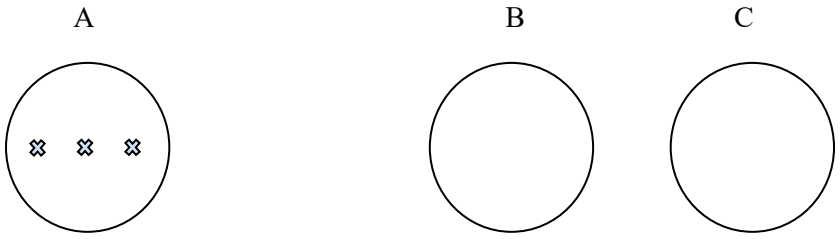
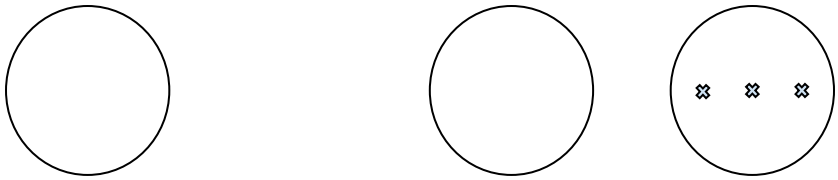
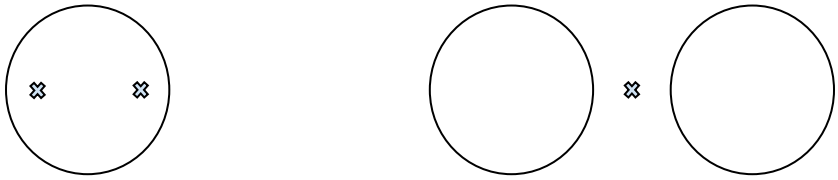


Fig 2



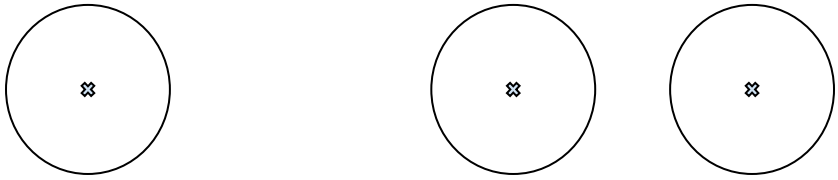
a. Describe the resulting cluster structure if k-means is used to cluster the points in Fig. 1 into 3 clusters. Also, mark final centroid locations in the figures above.

Solution:



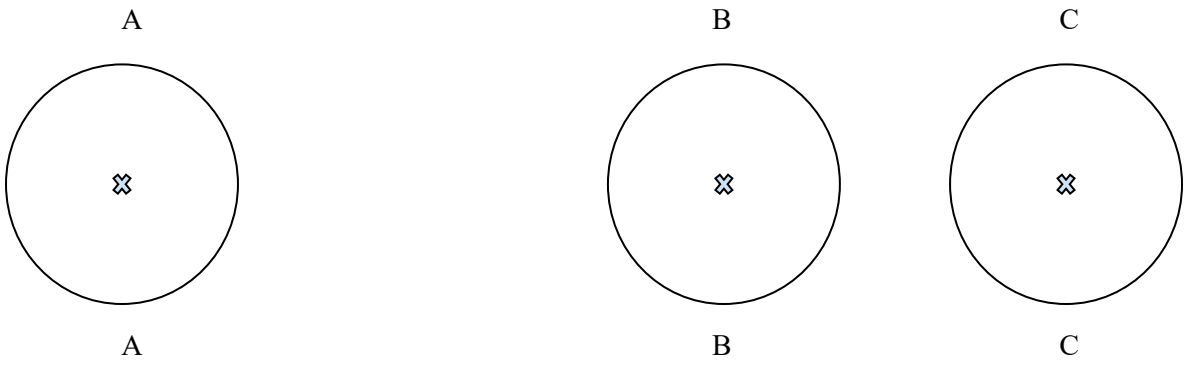
b. Describe the resulting cluster structure if k-means is used to cluster the points in Fig. 2 into 3 clusters. Also, mark final centroid locations in the figures above.

Solution:

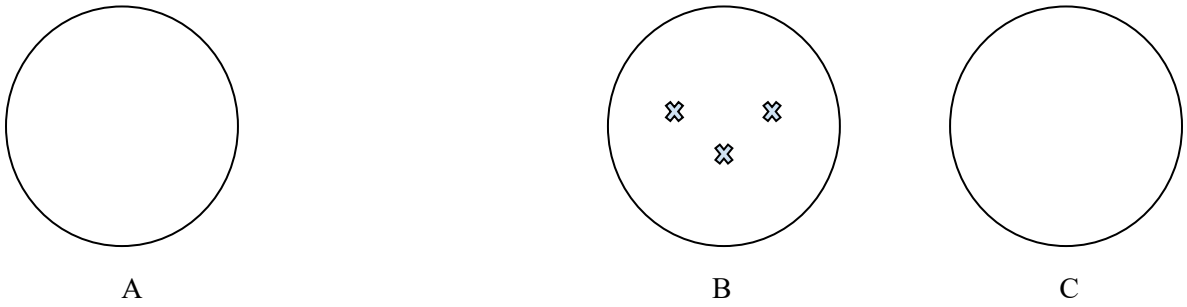


10. In the three sets of figures below, assume that the central circle (B) has 50,000 points, and the other two circles (A and C) have 50 points each. The X's are the centroid initializations for each run of K-means clustering. In the figures, mark the centroid location after convergence of K-means clustering. Assume a uniform distribution of points within each circle.

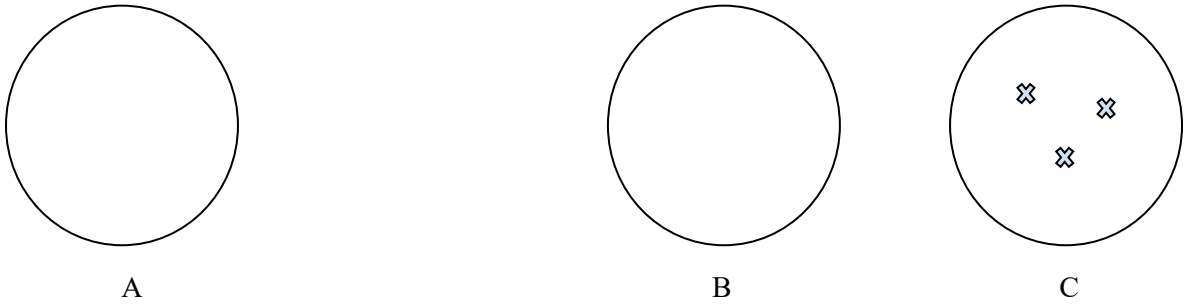
    Instructor note: By changing the number of points in each circle and the spacing and orientation of the circles, many similar problems can be created.



Solution: Final centroids the same as initial centroids.



Solution: All centroids are in circle B, but the top left initial centroid is on the left, towards circle A, while the right centroid is on the right towards circle C.

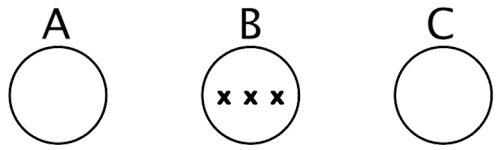


Solution: One centroid in circle C. Two centroids in circle B, with one slightly closer to circle A.

11. In each of the three sets of figures below, assume that circles A and B contain 100 points each, and circle C contains 100,000 points. The X's are the centroid initializations for each run of K-means clustering. Assume a uniform distribution of points within each circle. Each circle is the same size, and the distances between the circles is to scale.

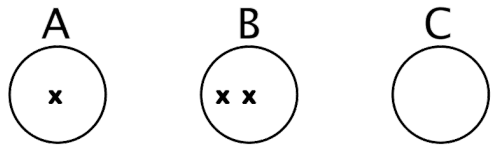
For each figure, you should tell how many centroids should end up in each circle after convergence of K-means clustering. Your answer should be 0, 1, 2, or 3. You should provide a brief justification for each case.

a) The distance between circles A and B is the same as the distance between B and C.



Number of Centroids in Circle (a): 1
Number of Centroids in Circle (b): 1
Number of Centroids in Circle (c): 1
Brief explanation: All of circle A's points will be assigned to the centroid in A. About ⅓ of circle B's points (the ones in the left third of circle B) will be assigned to the centroid on the left in circle B. The remaining ⅔ of the points in B and all the points in C will be assigned to the centroid in the center of B. This will cause the right centroid in B to move to circle C since C has many more points than B. In the next iteration, all points in A,B, and C will be assigned to the centroids located in their own circles and K-means will converge.

b) The distance between circles A and B is the same as the distance between B and C.



Number of Centroids in Circle (a): 1
Number of Centroids in Circle (b): 1
Number of Centroids in Circle (c): 1
Brief explanation: Since circles A and B are close together and quite far away from circle C, the points from both A and B will be assigned to the centroid that is in A. The points in C will be split between the two centroids in C, with each centroid having 50,000 points.
Since A and B have the same number of points each, the centroid in A will move between A and B. The centroids in C will move apart slightly but both will remain in C, each having half of C's points.

c) Circles A and B are much closer than B and C.

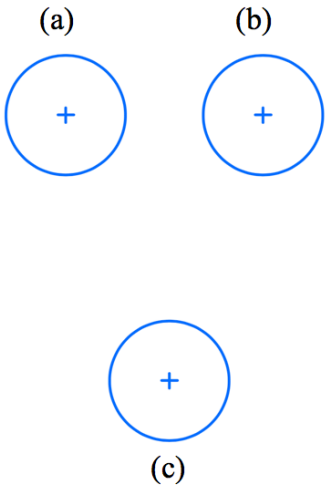
Number of Centroids in Circle (a): 0
Number of Centroids in Circle (b): 0
Number of Centroids in Circle (c): 2
Brief explanation:
Since circles A and B are close together and quite far away from circle C, the points from both A and B will be assigned to the centroid that is in A. The points in C will be split between the two centroids in C, with each centroid having 50,000 points.
Since A and B have the same number of points each, the centroid in A will move between A and B. The centroids in C will move apart slightly but both will remain in C, each having half of C's points.

12. In the three sets of figures below, assume that circle (a) has 100 points, and circles (b) and (c) have 100,000 points each. The +'s are the centroid initializations for each run of K-means clustering. Assume a uniform distribution of points within each circle.

    For each figure, you should tell how many centroids should end up in each circle after convergence of K-means clustering. Your answer should be 0, 1, 2, or 3. You should provide a brief justification for each case.

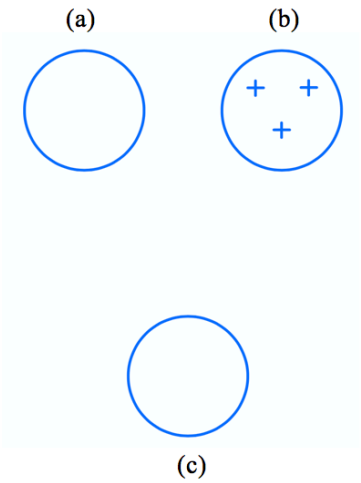
    Instructor note: By changing the number of points in each circle and the spacing and orientation of the circles, many similar problems can be created.

(i) Figure 1:



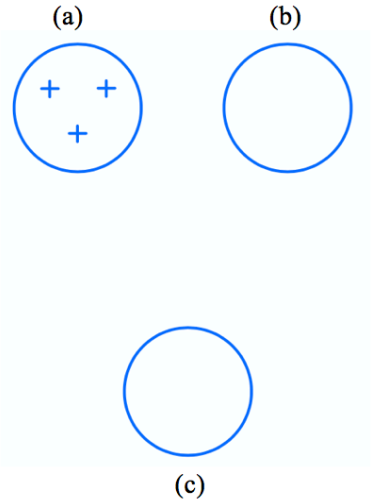
Number of Centroids in Circle (a): 1
Number of Centroids in Circle (b): 1
Number of Centroids in Circle (c): 1
Brief explanation: All points in all three circles are closer to the centroid in their circle than to centroids in other circles, so no points will move during the first iteration. Indeed, since the initial centroids are centrally located in each circle, and the distribution of points is uniform, the initial centroids will be the final centroids.

(ii) Figure 2:
Circle (a) have 100 points, circles (b) and (c) have 100,000 points each



Number of Centroids in Circle (a): 0
Number of Centroids in Circle (b): 2
Number of Centroids in Circle (c): 1
Brief explanation: Since both circles have the same number of points (100,000), the points in circle (c) will pull the bottom initial centroid of circle (b) towards circle (c), a process that will eventually move the centroid to circle (c). The top left centroid of circle (b) will be pulled slightly towards the points in circle (a), but since there are only 100 points in circle (a), that centroid will remain in circle (b). The top right centroid of circle (b) will remain in circle (b) but move more toward the middle of the circle.

(iii) Figure 3:
Circle (a) have 100 points, circles (b) and (c) have 100,000 points each

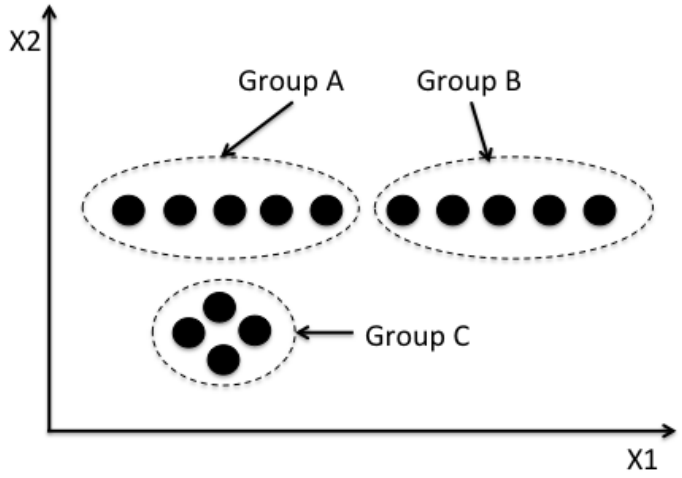


Number of Centroids in Circle (a): 1
Number of Centroids in Circle (b): 1
Number of Centroids in Circle (c): 1

Brief explanation: Circle (a) The top right initial centroid of circle (a) will be pulled towards circle (b) and since circle (b) has 100,000 points vs. 100 in circle (a), the centroid will move to the center of circle (b). A similar process will move the bottom initial centroid of circle (a) to the center of circle (c). One centroid will remain in circle (a) and will move to the center of circle (a).

Hierarchical Clustering

1. At an intermediate stage of some agglomerative clustering algorithm, you are given three groups of points, as shown in the figure below, which need to be considered for merging. Note that every circle in the figure represents a two-dimensional point, and the Euclidean distance in two dimensions is being used as the distance measure.



   a) Using the single link (MIN) hierarchical clustering technique, which pair of groups would you consider for merging? Provide a one-sentence justification.

      Groups A and B will be merged since they have the smallest single link distance (between the right-most point of A and left-most point of B), as compared to Groups A and C, and Groups B and C.

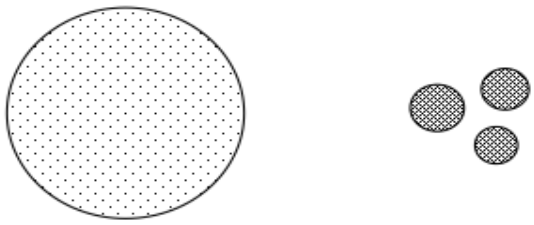
   b) Using the complete link (MAX) hierarchical clustering technique, which pair of groups would you consider for merging? Provide a one-sentence justification.

      Groups A and C will be merged since they have the smallest complete link distance (between the right-most point of A and the farthest point in C), as compared to the complete link distance of Groups A and B (between the left-most point in A and right-most point in B), and Groups B and C (between right-most-point in B and the farthest point in C).

2. Given the four clusters shown in the figure below, answer the following questions.

Assume that: (1) Data points within the clusters are uniformly distributed; (2) Between cluster distances of the three small clusters (on the right) are larger than the distances between points in the large cluster (on the left); (3) The small clusters have a higher density than the big cluster.

   a) Will MAX agglomerative clustering identify the four clusters? Explain briefly.
   b) Will MIN agglomerative clustering identify the four clusters? Explain briefly.



a) No, MAX will merge three smaller clusters before it finds the large cluster.

b) Yes, MIN will find four clusters.

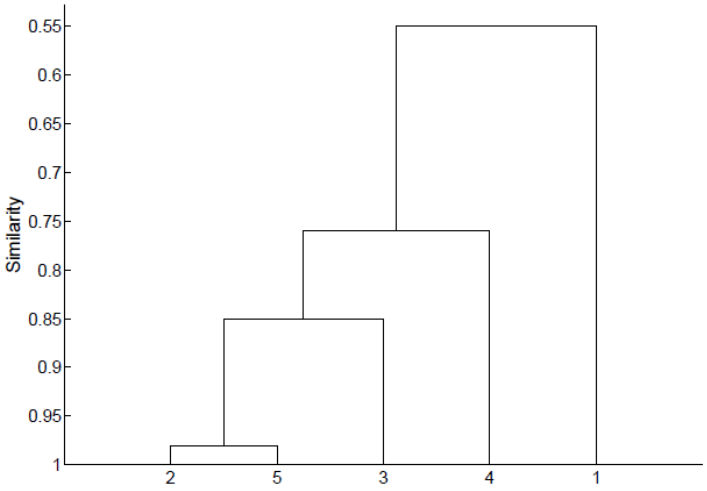
3. Use the similarity matrix in Table 2 to perform single and complete link hierarchical clustering. Show your results by drawing a dendrogram. The dendrogram should clearly show the order in which the points are merged.

	P1	P2	P3	P4	P5
P1	1.00	0.10	0.41	0.55	0.35
P2	0.10	1.00	0.64	0.47	0.98
P3	0.41	0.64	1.00	0.44	0.85
P4	0.55	0.47	0.44	1.00	0.76
P5	0.35	0.98	0.85	0.76	1.00

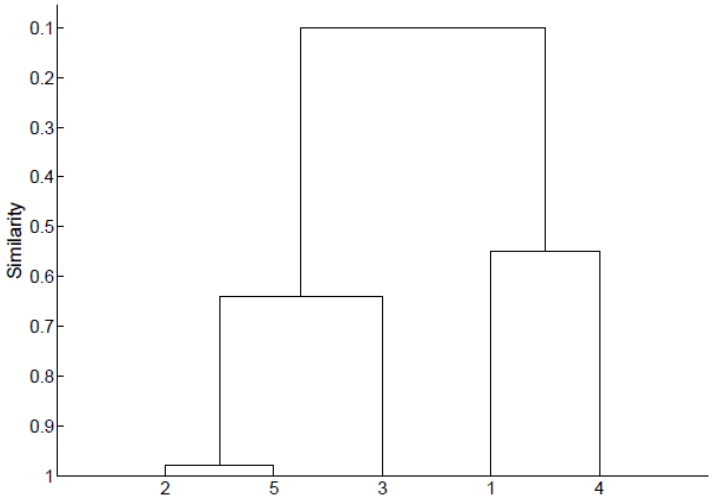
Table 2: Similarity matrix.

Since the proximity matrix is given in terms of the similarity matrix, you can either convert it to a distance matrix before building clusters, as shown in the example in the book, or construct the clusters by reversing the min to max relation and vice versa.

The single-link algorithm will yield to the following sequence of clustering:
<<<<P2, P5>,P3>, P4, >, P1>.



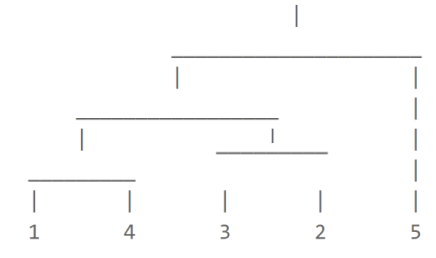
The complete-link will yield to the following sequence of clustering: <<<P2, P5>,P3>, <P4, P1>>.



4. Given the Similarity Matrix below, as well as a three dendrograms, indicate whether that dendrogram was produced by a MIN clustering, MAX clustering, or neither.

Pairwise Similarities	Point 1	Point 2	Point 3	Point 4	Point 5
Point 1	10	8	4	9	1
Point 2	8	10	7	5	3
Point 3	4	7	10	5	6
Point 4	9	5	5	10	2
Point 5	1	3	6	2	10

Dendrogram 1

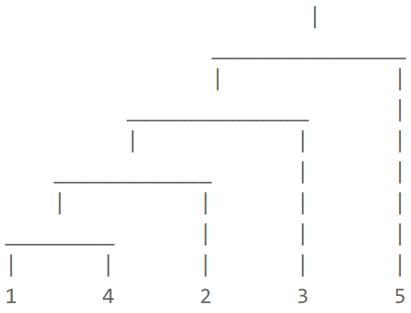


MAX

Dendrogram 2

Neither

Dendrogram 3



MIN

5. Describe the complete link clustering technique with respect to the following characteristics.

a. Ability to cluster large data sets.

Because complete link has a time complexity of O(nlog(n)) and space complexity of O(n2), its ability to cluster large data sets is limited unless sampling or some other approach is used.

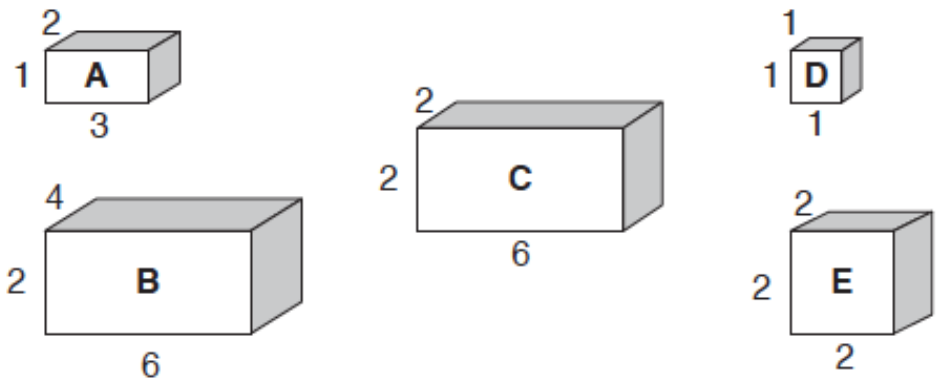
b. Ability to find clusters of differing densities.

Complete link defines a cluster in terms of the maximum distance between two points in a cluster. Thus, complete link can find clusters of differing densities provided that other factors, e.g., differing sizes and non-globular shapes, do not interfere.

c. Objective function that is optimized and whether the optimization is local or global.
Complete link greedily minimizes the distance between clusters in terms of the maximum distance between two points in different clusters. As such, it is not guaranteed to find a global optimum.

d. Manner in which the resulting clusters are described. (Assume that k clusters are found by slicing the dendrogram at the appropriate level.)
The clusters are described via cluster labels or equivalently, a list of the points in each cluster. There is nothing equivalent to a cluster centroid or medoid.

6. Consider the five objects (A, B, C, D, and E) shown in the figure below. Each object has three features: length, width, and height. For example, the features of object A are (3,2,1).



(a) Suppose we apply the single link (MIN) algorithm to cluster the objects. Draw the dendrogram for the clusters assuming the similarity measure is Euclidean distance.
Ans: Proximity matrix

	A	B	C	D	E
A					
B	√14				
C	√10	√4			
D	√5	√34	√27		
E	√2	√20	√16	√3	

Join A and E
Join A and E and D
Join B and C
Join (A, E, D) and (B, C)

(b) Repeat the question in part (a) assuming that the similarity measure is correlation.

Proximity matrix

	A	B	C	D	E
A					
B	1				
C	0.87	0.87			
D	NaN	NaN	NaN		
E	NaN	NaN	NaN	NaN	

Since many of the correlations are not defined, we won't be able to cluster all the points (e.g., D and E).

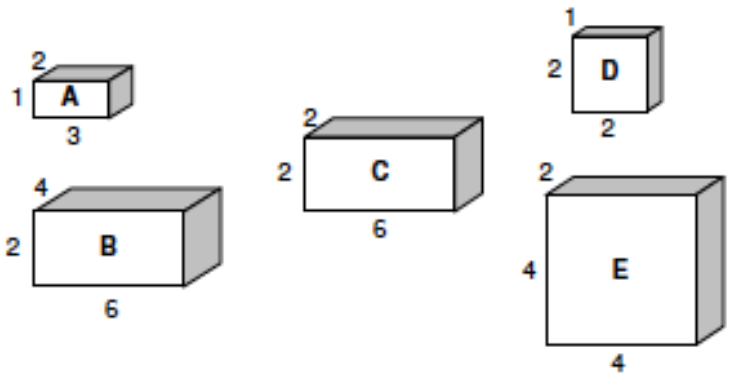
(c) Suppose we apply the complete link (MAX) algorithm to cluster the objects. Draw the dendrogram for the clusters assuming the similarity measure is Euclidean distance.

Join A and E

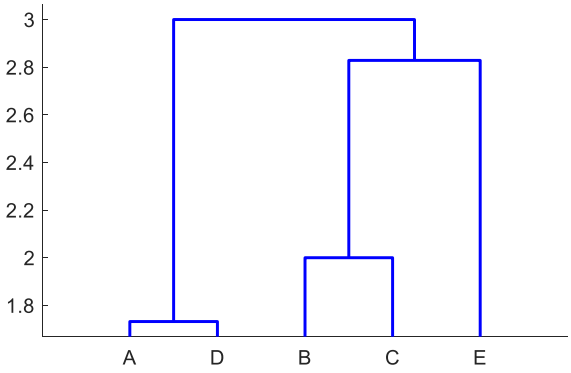
Join B and C

Join (A, E) and (B, C)

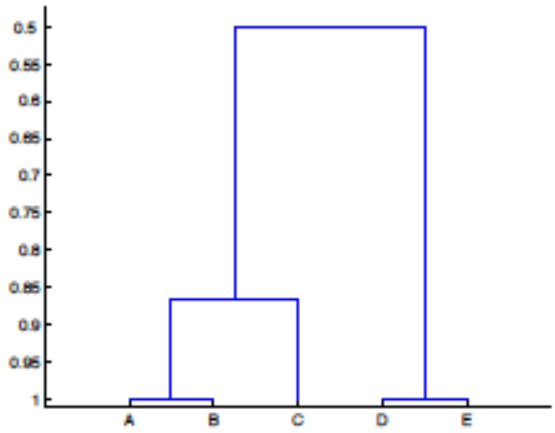
7. Consider the five objects (A, B, C, D, and E) shown in the figure below. Suppose we apply the single link (MIN) algorithm to cluster the objects.



(a) Draw the dendrogram for the clusters assuming the proximity measure is Euclidean distance.



(b) Draw the dendrogram for the clusters assuming the proximity measure is correlation.



8. Consider a set of 5 points in two-dimensional space, shown in the following table:

Point ID	X	Y
1	9	8
2	6	8
3	6	4
4	10	6
5	3	1

Assuming Euclidean distance as the distance measure, answer the following questions:

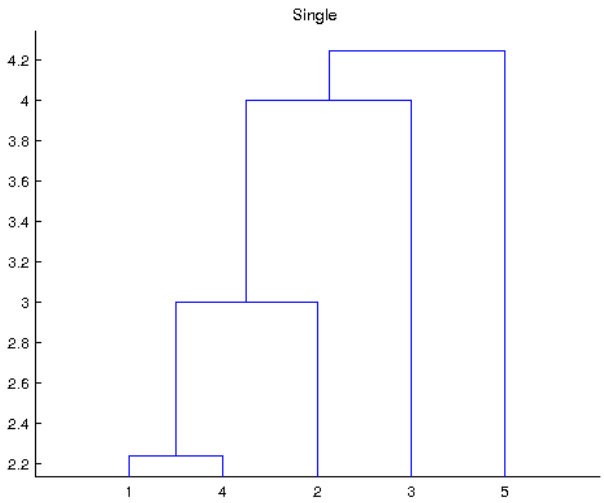
a) Compute the matrix of pairwise distances between the 5 points, where the ijth entry in the matrix corresponds to the distance between point i and point j.

Pairwise Distances	Point 1	Point 2	Point 3	Point 4	Point 5
Point 1	0	3	5	2.23	9.2
Point 2	3	0	4	4.47	7.6
Point 3	5	4	0	4.47	4.2
Point 4	2.23	4.47	4.47	0	8.6

Point 5	9.2	7.6	4.2	8.6	0

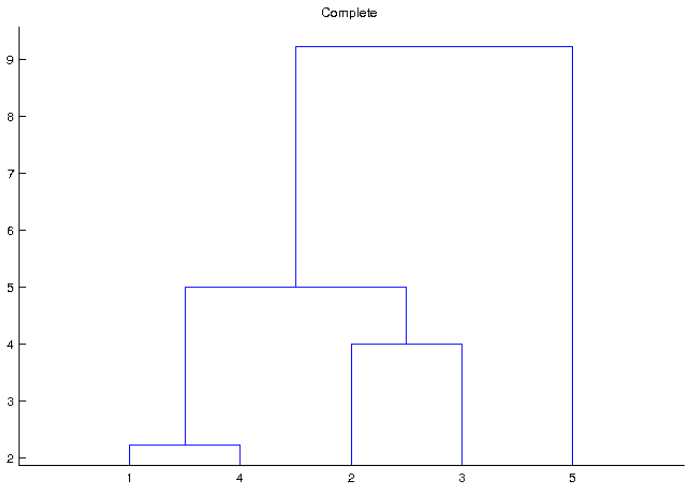
b) Use the single link (MIN) hierarchical clustering technique for clustering these 5 points and show the dendrogram of the clustering.

Answer:



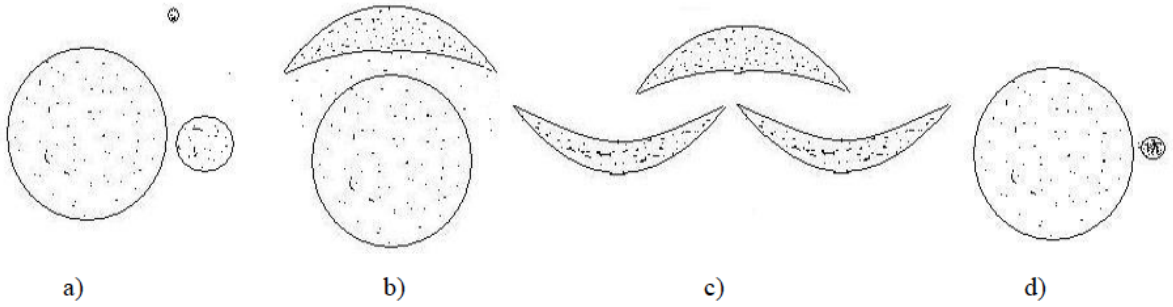
c) Use the complete link (MAX) hierarchical clustering technique for clustering these 5 points and show the dendrogram of the clustering.

Answer:



9. How will single-link and complete-link perform for the following cases? The points are evenly

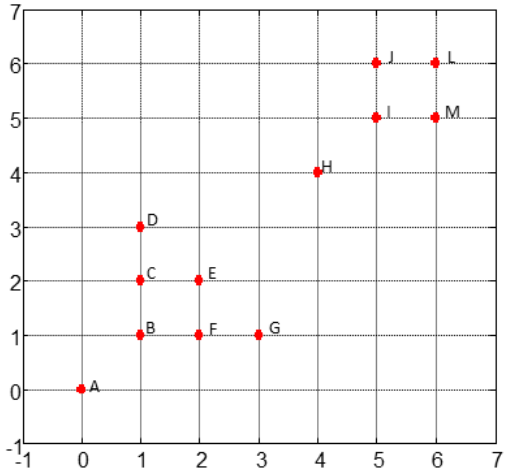
distributed for the first three cases(a-c), while the last case(d) has one dense cluster with 50000 points and one relatively sparse cluster with 50 points only without any noisy data points in between. Assume that the points inside a boundary are denser than the points outside the boundary, which represent the noise points (case b).



Ans: a) MIN will perform well, but MAX may break the larger cluster and merge it with the smaller cluster. b) MIN is sensitive to noise, MAX is not sensitive to the noise but can break the crescent-shaped cluster. c) MIN will perform well; MAX will also break the clusters into globular shape. d) MIN will perform well if the distance between the two circles is greater than the distance between the nearest points in the sparser cluster. MAX will tend to break the larger cluster and merge it with the smaller cluster.

DBSCAN

1. Suppose we apply DBSCAN to cluster the following dataset using Euclidean distance.



Given that MinPts = 3 and EPS = 1, answer the following questions.

a) Label all points as "core points," "boundary points," and "noise."

Core points: B, C, E, F, I, J, L, M

Border points: D, G.

Noise: A, H.

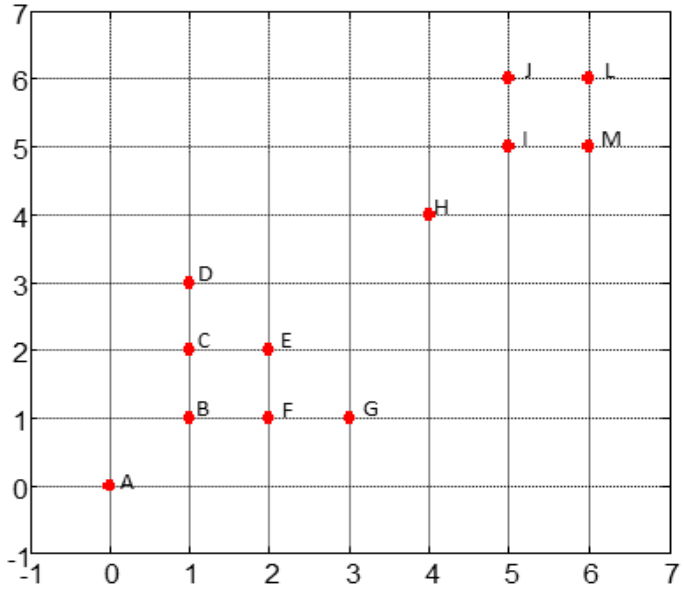
b) What is the clustering result?

There will be two clusters: B, C, D, E, F, G, will form a cluster, and I, J, L, M will form the other cluster.

c) Repeat the above two questions when epsilon = $\sqrt{10}$.

Core points: B, C, E, F, I, J, L, M, D, G, A, H, and all points are clustered as a single cluster.

2. Suppose we apply DBSCAN to cluster the following dataset using Euclidean distance.



A point is a core point if its density (num point within EPS) is ≥ MinPts. Given that MinPts = 3 and EPS = 1, answer the following questions.

a) Label all points as "core points," "boundary points," and "noise."

A core point, as defined here, has its distance to its two nearest neighbors ≤ 1.

Core points: B, C, E, F, I, J, L, M

Border points: D, G

Noise: A, H

b) What is the clustering result?

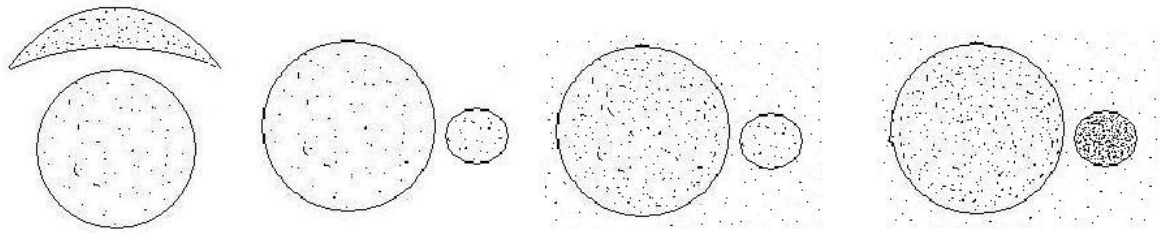
There will be two clusters. B, C, D, E, F, G will form a cluster, and I, J, L, M, will form the other cluster.

c) Repeat the above two questions when epsilon = $\sqrt{2}$.

Now the core points are defined as points with distance to its 2-NN <= 2.
Core points: B, C, E, F, I, J, L, M, D, G,
Border points: A, H

All points are clustered as a single cluster.

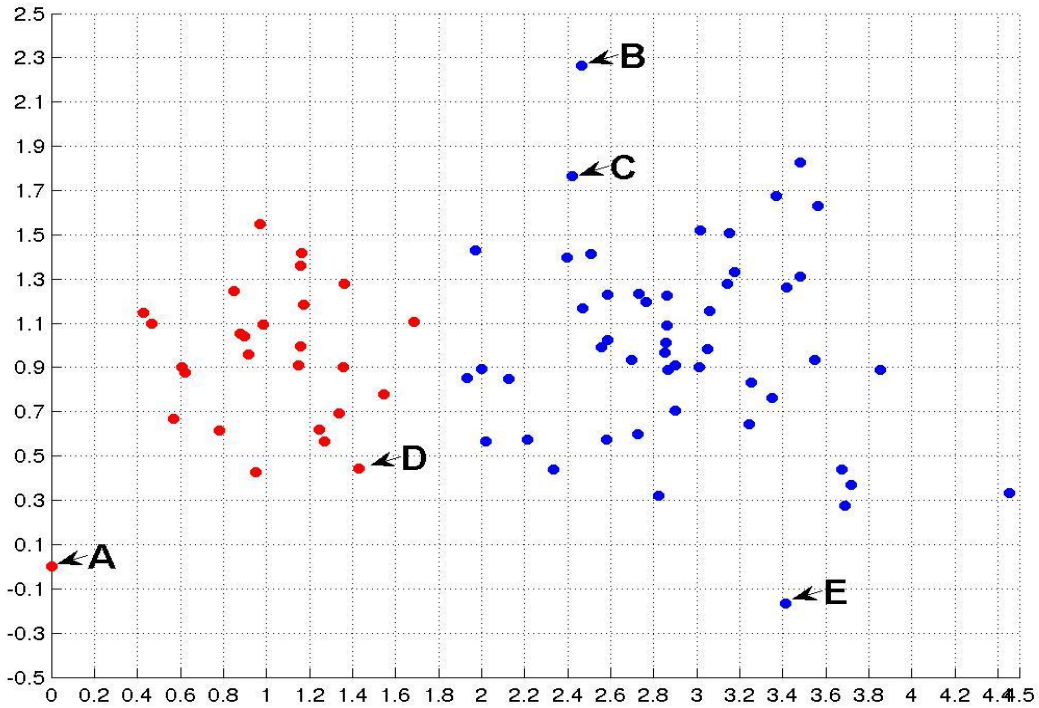
3. How will DBSCAN perform for the following cases? Describe the effect of the EPS parameter on clustering them with a fixed MinPts. Assume that the points inside a boundary are denser than the points outside the boundary, which represent the noise points (case c and d).



a) b) c) d)

Ans: For all cases except d), DBSCAN will perform well. If eps is too large, then the noise points will also be considered as the points of the cluster. If eps is too small, then only the denser cluster will be detected, but the other one will not be detected.

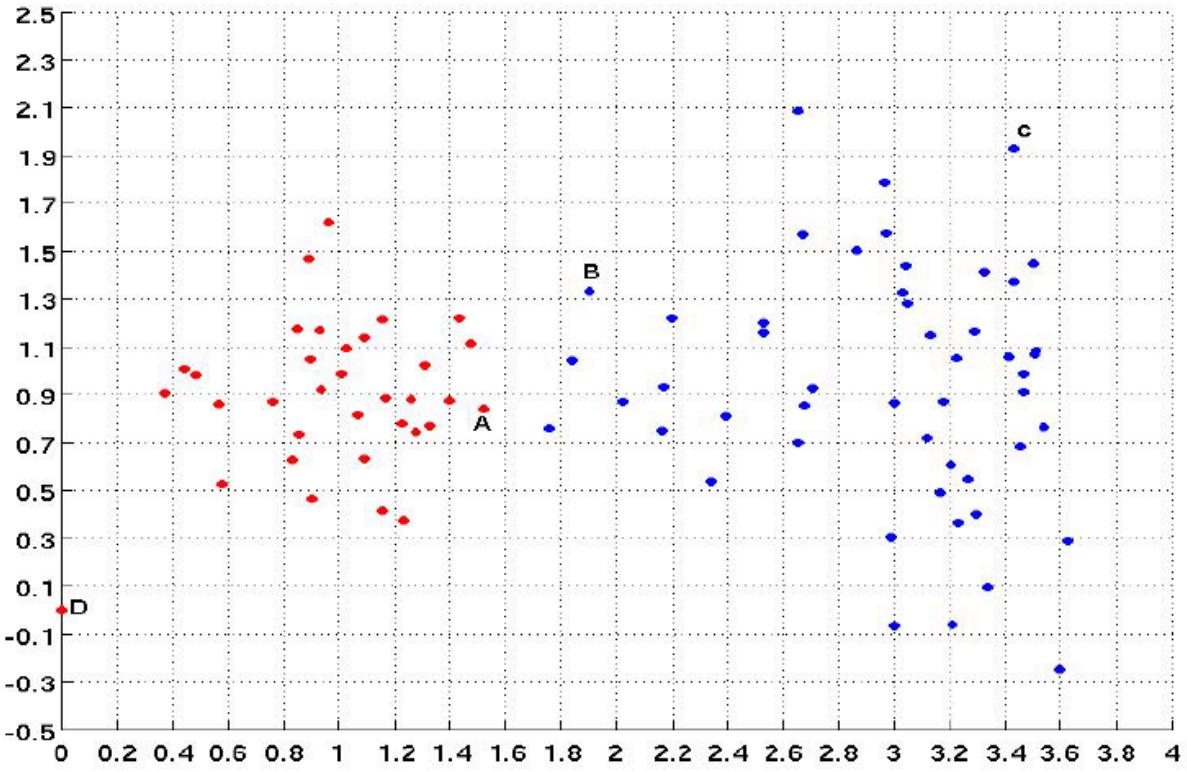
4. In the figure below, mark whether the points A, B, C, D, E are border, core and noise point for a) EPS=0.4 and b) EPS=0.6 with MinPts=5 for both cases. You don't need to compute the actual distance between the points but should roughly estimate the neighborhood of each point to find the border, core, and noise points. A point is a core point if its density (num point within EPS) is ≥ MinPts.



Ans: a) EPS = 0.4: core point – D, border point – C, noise - A, B, E

a) EPS = 0.6: Core – C, D, border point – B, E, Noise – A

5. Using the DBSCAN clustering algorithm, mark whether the points A, B, C, and D shown in the figure below are border, core or noise point for a) EPS = 0.5 and b) EPS = 1, with MinPts = 4 for both the cases. You don't need to compute the actual distance between the points but should roughly guess the neighborhood of each point to find the border, core, and noise points. How many clusters do you think DBSCAN will find for both cases? For density, count the point itself. A point is a core point if its density (num point within EPS) is ≥ MinPts.



a) EPS = 0.5: core- A, border- B, noise- C, D

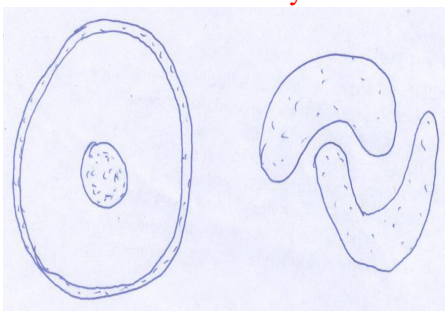
b) EPS=1: core- A, B, C, border- D.

There will be 2 and 1 cluster(s) found for EPS=0.5 and EPS=1, respectively.

Cluster Evaluation

1. To obtain a measure of cluster goodness, the silhouette coefficient combines a measure of cluster coherence and a measure of cluster separation. Give an example of a set of clusters for which the clusters are well defined according to one of the standard definitions (e.g., center-based, contiguity-based, density-based), but the silhouette coefficient doesn't work well, i.e., the value of the silhouette coefficient does not indicate a good clustering.

   The silhouette coefficient is first computed for individual points of a cluster, and then they are averaged to get the final score for a cluster. Here, we consider the score for **clusters,** not the individual points. In the case of contiguity-based clusters, such as those in the figure below, the silhouette coefficient may not work well.



2. The following table (confusion matrix) shows the clustering results in a land cover classification dataset that consists of many pieces of land. The number provided in the table is the number of objects (pieces of land) that are clustered into each cluster that belongs to each category. For example, the number in the forest column and cluster 1 row means that 10 forest items are clustered into cluster 1. Answer the following questions based on the given table. No calculations are necessary.

Table: Clustering results for land cover classification dataset.

	Forest	Farm	Shrubland	Urban	Water
Cluster 1	20	10	10	10	950
Cluster 2	400	100	400	50	50
Cluster 3	50	50	500	200	200
Cluster 4	200	250	150	200	200

   1. Which cluster has the smallest entropy?
      Cluster 1

   2. Which cluster has the biggest entropy?
      Cluster 4

3. The following table (confusion matrix) shows the k-means clustering results for a land cover classification dataset that consists of many pieces of land. The number provided in the table is the number of objects (pieces of land) that are clustered into each cluster that belongs to each category. For example, the number in the forest column and cluster 1 row means that 10 forest items are clustered into cluster 1. Answer the following questions based on the given table. No calculations are necessary.

Table: k-means clustering results for land cover classification dataset

	Forest	Farm	Shrubland	Urban	Water
Cluster 1	10	100	20	10	30000
Cluster 2	3000	10	1000	10	0
Cluster 3	10	3000	500	150	200
Cluster 4	2000	2500	1500	3000	1400

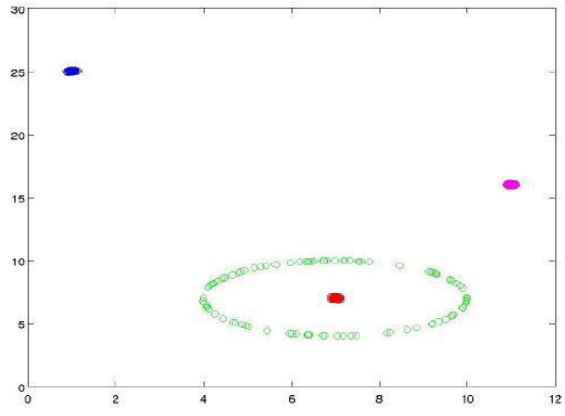
a) Which cluster has the largest clustering entropy? Cluster 4

b) Which cluster has the smallest clustering entropy? Cluster 1

4. Assume you are given a data set of objects, each of which is assigned to one of two classes, and suppose that C1 and C2 are two clusterings produced from this data set. If entropy judges C1 to be a more accurate clustering than C2, is it necessary that SSE will also judge C1 to be a more accurate clustering than C2?

No, since entropy is a supervised measure, while SSE is unsupervised, and thus they provide very different perspectives of cluster validity. It is possible that while C1 captures the class structure of the data set closely, it may not produce very tight clusters in terms of their SSE scores. On the other hand, a clustering judged to be the best by SSE may not capture the class structure very well, since SSE doesn't take the knowledge of class labels into account.

5. As shown in Fig. A and Fig. B, there are two datasets of 400 two-dimensional data points in 4 different clusters colored by four different colors – blue, red, green, and magenta. Also given are the distance matrices (400 x400) for the two datasets in Fig. C and Fig. D. The colorbar shows the distances between points, with the color blue indicating the lowest distances and the color red indicating the highest.



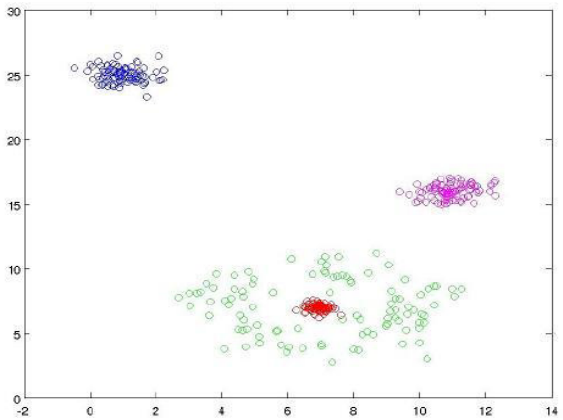


Fig. A

Fig. B

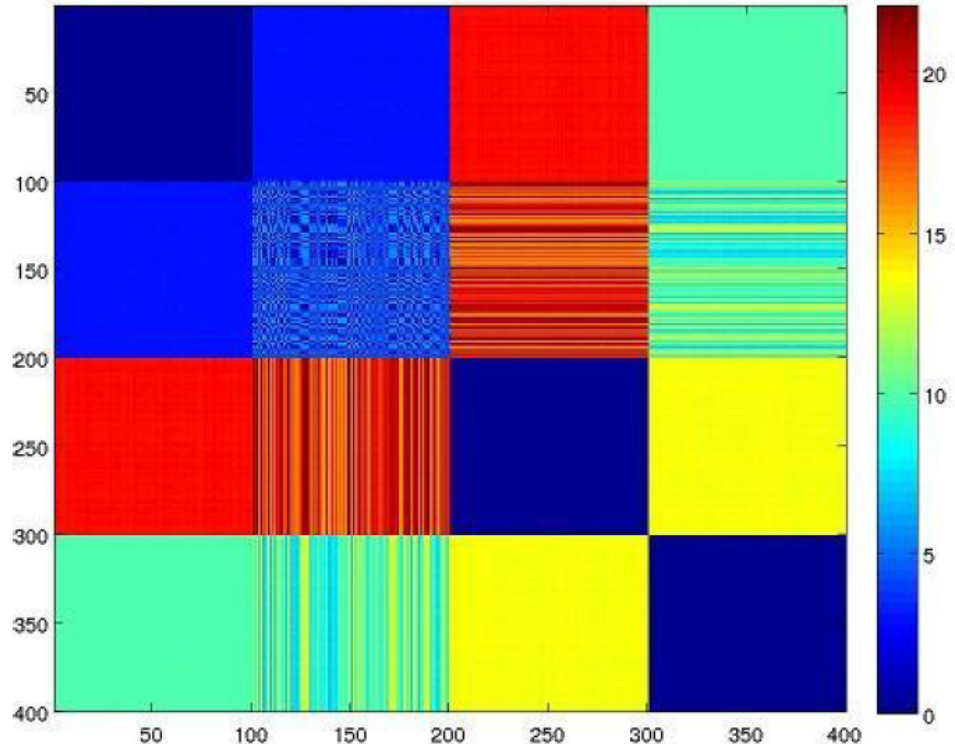


Fig. C

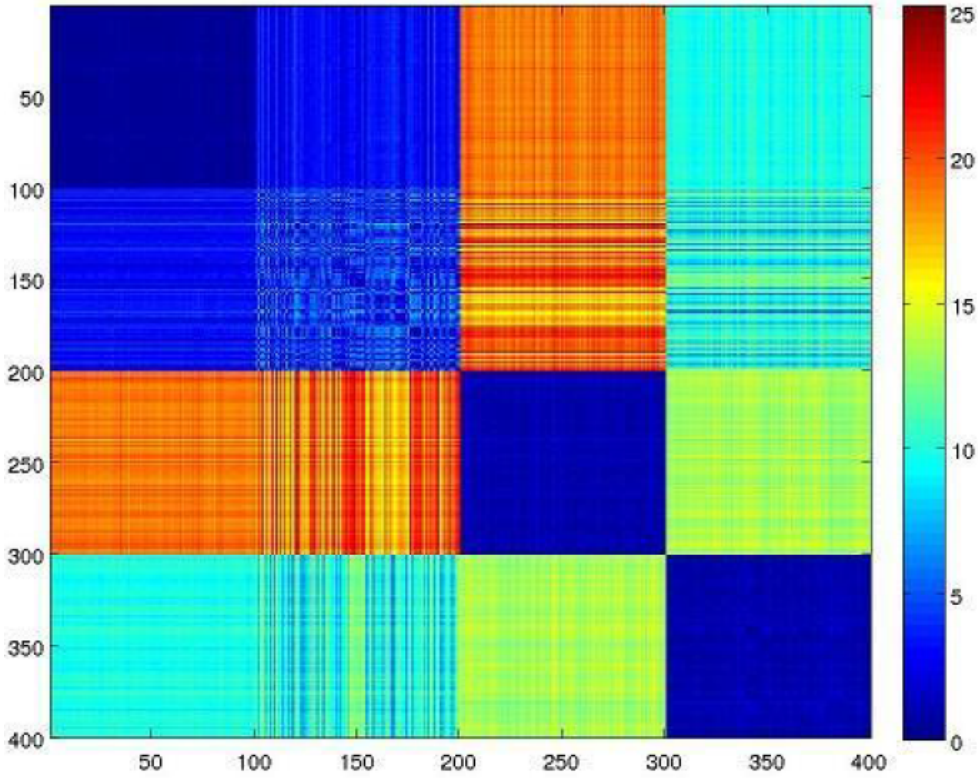


Fig. D

(a) Match the matrices with their corresponding datasets and provide a brief justification.
The distance matrix in Fig. C corresponds to the dataset in Fig. A and the one in Fig. D corresponds to the dataset in Fig. B. The clusters in A are very compact as compared to the ones in B. Therefore, except for the green cluster, which is not globular, the distances between any two other clusters are going to be very similar. Therefore, the squares are expected to be more uniformly colored in the distance matrix for A than for B. Clearly, the one in C satisfies this criterion and thus corresponds to A.

(b) Also, figure out the corresponding rows (or columns, doesn't matter which as the matrix is symmetric) representing the points of each cluster. Assume that the rows corresponding to the points of a cluster are consecutive. Therefore, Rows 1-100 belong to the same cluster, Rows 101-200 belong to the same cluster, and so on.
The distance between any two points within a globular cluster would have low variance as compared to the ring cluster. The diagonal squares represent the distances between the points of the same cluster. Clearly, the square in the second row or column is showing higher variance compared to all other squares in the diagonal. Therefore, Column 2 belongs to the green cluster.

The blue cluster is farthest from all other three clusters, and thus its corresponding row or column is expected to be reddish (higher distances), which is column 3. Column 3 belongs to the blue cluster.
As the red cluster is closer to the green cluster than the magenta cluster, therefore, the column which is more bluish with 2nd row (which corresponds to the green cluster) is expected to belong to the red cluster. Column 1 intersects with Row 2 with a deeper blue shade than Column 4. Therefore, column 1 corresponds to the red cluster.

The remaining column 4 corresponds to the magenta cluster.

6. The distance matrices below (Figures A, B, and C) are sorted according to cluster labels, and correspond, in some order, to the sets of points (Figures D, E, and F). Differences in color distinguish between clusters, and each set of points contains 100 points and four clusters, each of equal size. In the distance matrix, blue indicates the lowest distances, and red indicates the highest distances.

(a) Match the distance matrices (Figures A, B, and C) with the sets of points (Figures D, E, and F).

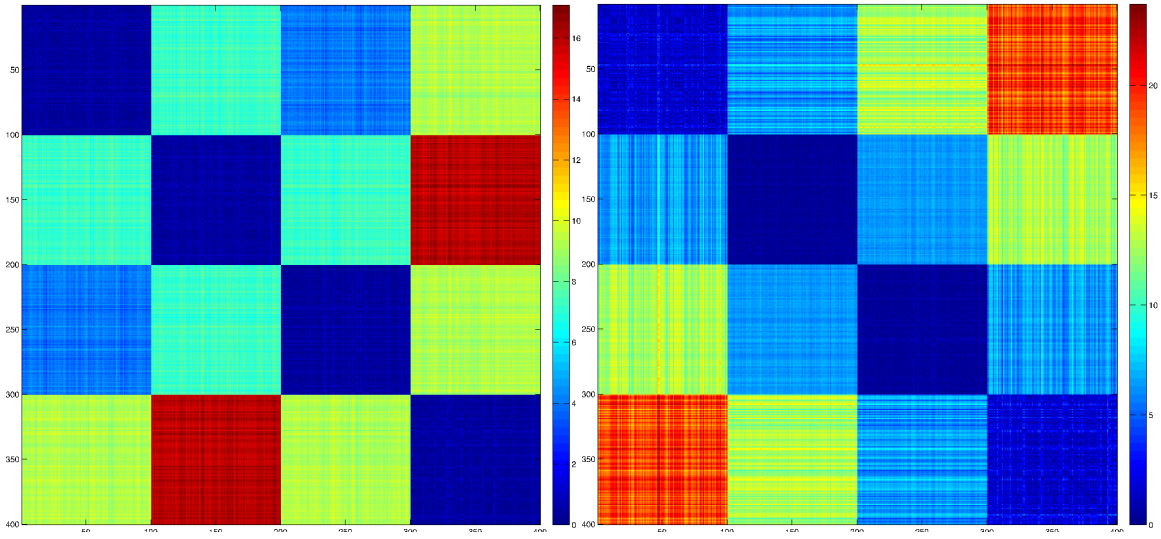


Figure A **Figure B**

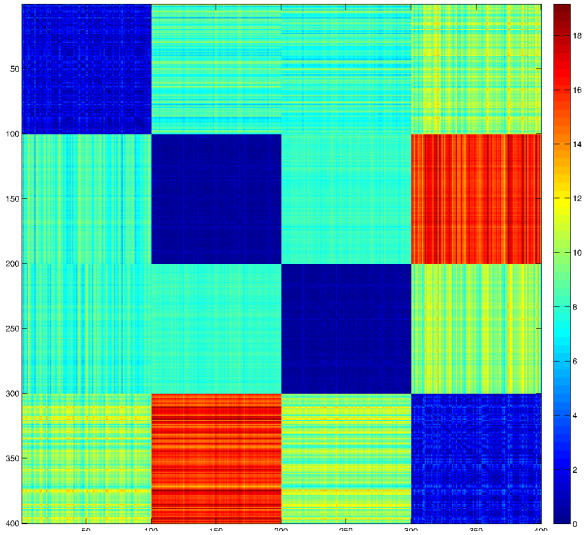


Figure C

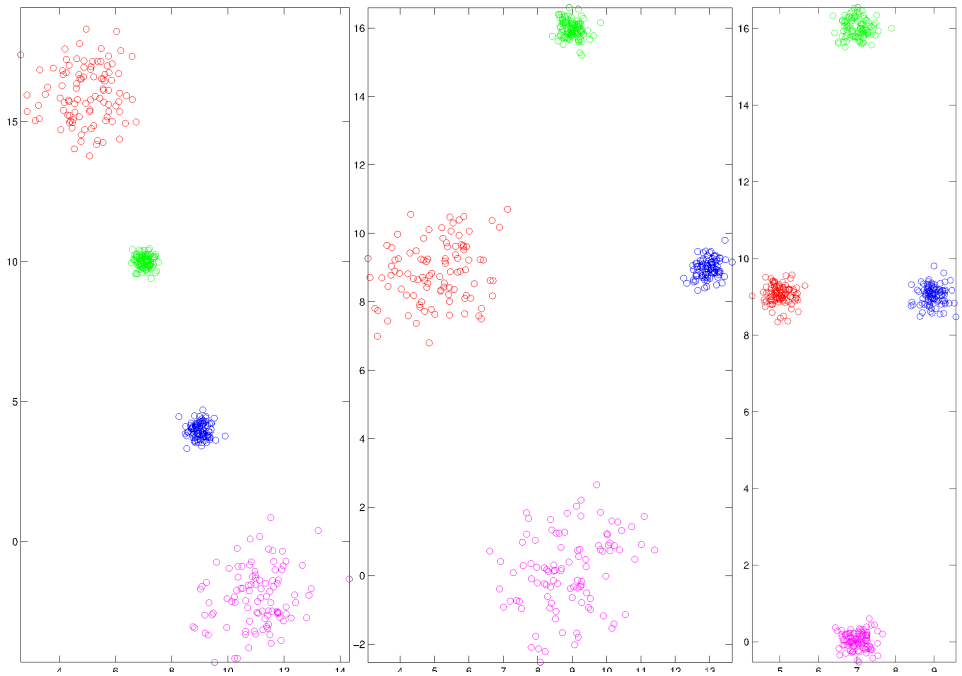


Figure D **Figure E** **Figure F**

Matrices	Datasets
1(a)	
1(b)	
1(c)	

Matrices	Datasets
1(a)	(f)
1(b)	(d)
1(c)	(e)

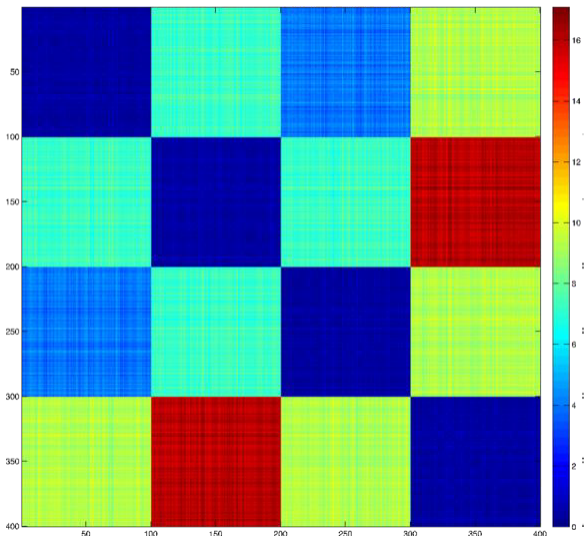
(b) For the symmetric matrix given in **Figure B,** match the four rows to the corresponding clusters (characterized by four colors – red, blue, green, and magenta) in the dataset that you match with it in the previous question.

Matrix rows	Cluster (represented by colors)
1st (1-100)	
2nd (101-200)	
3rd (201-300)	
4th (301-400)	

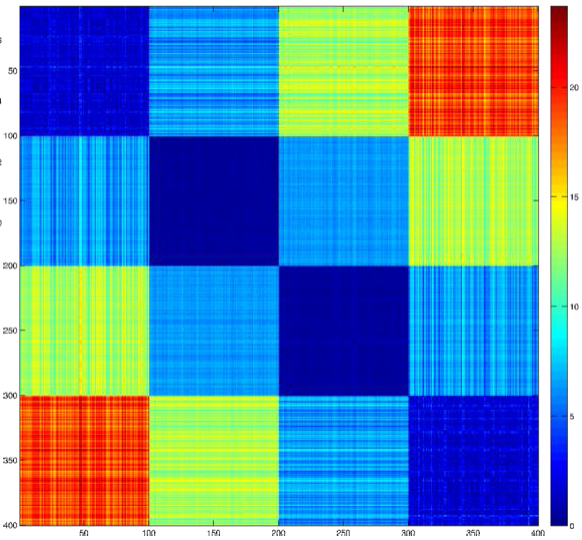
Matrix rows	Cluster (represented by colors)
1st	Red
2nd	Green
3rd	Blue
4th	Magenta (pink)

7. The figures below are sorted according to cluster labels, and corresponds to the sets of points (Dataset X, Dataset Y, and Dataset Z). Differences in color distinguish between clusters, and each set of points contains 100 points and four clusters each of equal size. In the distance matrix, blue indicates the lowest distances and red indicates the highest distances.

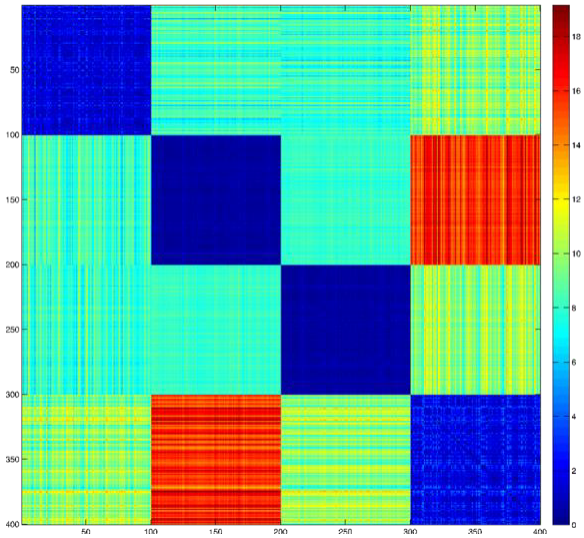
(a) Match the distance matrices (Matrix 1, Matrix 2, Matrix 3) with the sets of points (Dataset X, Dataset Y, and Dataset Z). **Provide a brief explanation**



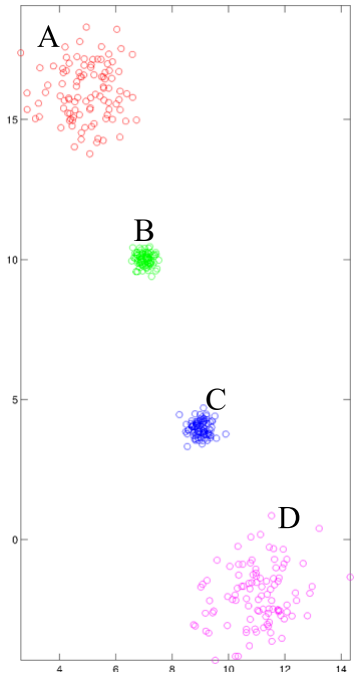
Matrix 1



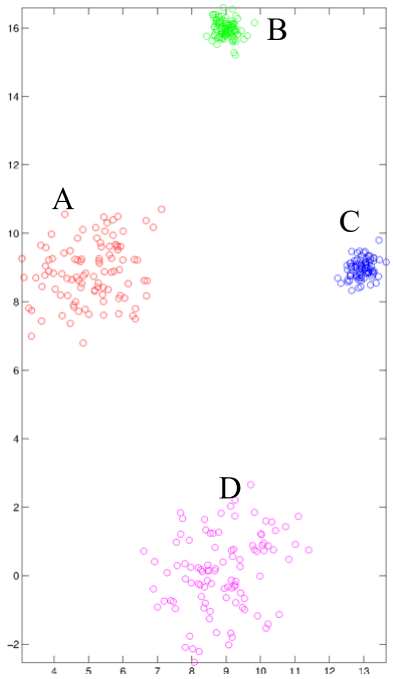
Matrix 2



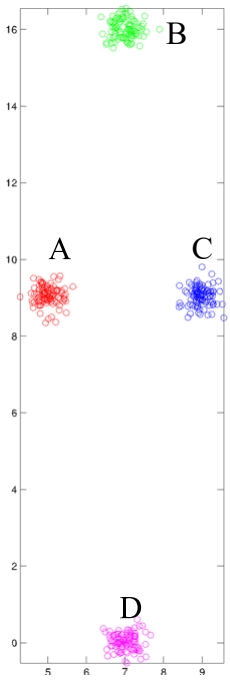
Matrix 3



Dataset X



Dataset Y



Dataset Z

i. Matrix 1
Dataset (X/Y/Z): Z
Explanation:

Diagonal entries explanation:

- Diagonal entries are all similarly crisp and blus, the most crisp compared to other confusion matrices, indicating that points that belong to the same cluster are closest to each other, and cluster cohesion is similar for all clusters

Off-diagonal entries explanation:

1. Rows 1 and 3 correspond to clusters A and C. This is because the colors of the off-diagonal entries for these two rows are all different, indicating the different distances between cluster A (or C)' s distances to all other clusters (i.e: A is closest to C (blue off-diagonal); followed by B (green off-diagonal); and is the furthest from D (yellow off-diagonal); similar explanation for C).
2. Row 2 correspond to cluster B. Same distances to A and C (green off-diagonal), furthest distance from A (red off-diagonal)
3. Row 4 correspond to cluster D. Same distances to A and C (yellow off-diagonal), furthest distance from B (red off-diagonal).

ii. Matrix 2
Dataset (X/Y/Z): X
Explanation

Diagonal entries explanation:

- 2 diagonal entries are more blue and crisp compared to the other 2, indicating 2 clusters have better cohesion (B and C) than the other 2 (A and D)

Off-diagonal entries explanation:

1. Rows with less crisp diagonal entries (rows 1 and 4) have all different colors, indicating that all other clusters have different distances from these clusters (e.g: Cluster A is the nearest to B, followed by C and then D, no 2 clusters have same distance to cluster A)
2. Rows with more crisp diagonal entries have 2 same colors (other than the diagonal), indicating that it has same distance to 2 clusters, and is the furthest from 1 cluster (e.g: B's distance to A and C is similar, but is the furtherst from D)

iii. Matrix 3
Dataset (X/Y/Z): Y
Explanation:

Diagonal entries explanation:

- 2 diagonal entries are more blue and crisp compared to the other 2, indicating 2 clusters have better cohesion (B and C) than the other 2 (A and D)

Off-diagonal entries explanation:

1. All rows have 2 similar and 1 different colors off diagonals entries. This indicates each cluster has 2 other clusters relatively closer to it than the remaining 1 cluster (e.g: B is similarly close to A and C compared to with D)
2.

(b) For the symmetric matrix given in **Matrix 2** match the four rows to the corresponding clusters (characterized the nearest alphabet in each cluster (e.g: A, B, C, D)) in the dataset that you match with it in previous question. **Provide a brief explanation**

i. Matrix rows: 1st (1-100) Cluster (A/B/C/D): A Explanation:
Diagonal entry is less crisp, meaning the cluster is less cohesive. All off- diagonal entries have different colors, indicating all other clusters have different distances from is (closest to B, fllowed by C, and furthest from A)

ii. Matrix rows: 2nd (101-200)
Cluster (A/B/C/D): B
Explanation:

Diagonal entry is more crisp, indicating the cluster is cohesive. 2/3 off-diagonal entries have the same color, indicating 2 other clusters are closer to it (A and C, eventhough the off-diagonal indicating distances with A is less crisp), and is the furthest from 1 other cluster (D)

iii. Matrix rows: 3rd (201-300)
Cluster (A/B/C/D): C
Explanation: Similar to (ii)

iv. Matrix rows: 4th (301-400)
Cluster (A/B/C/D): D
Explanation: Similar to (i) in inverted order

8. To obtain a measure of cluster goodness, the silhouette coefficient combines a measure of cluster

coherence and a measure of cluster separation. Give an example of a set of clusters for which the clusters are well defined according to one of the standard definitions (e.g., center-based, contiguity-based, density-based), but the silhouette coefficient doesn't work well, i.e., the value of the silhouette coefficient does not indicate a good clustering

The silhouette coefficient is first computed for individual points of a cluster, and then they are averaged to get the final score for a cluster. Here, we consider the score for **clusters,** not the individual points (although partial points were given for those cases). In the case of contiguity-based clusters, the silhouette coefficient may not work well.

9. For the figures below, please match the distance matrices (Fig. (a), (b), and (c)) with their corresponding datasets (Fig. (d), (e), and (f)).

Each dataset contains four clusters, and each cluster has 80 points. In the distance matrices, points are sorted according to the four cluster labels. Different darkness indicates differences in the distance: black indicates the lowest distances, and white indicates the highest distances.

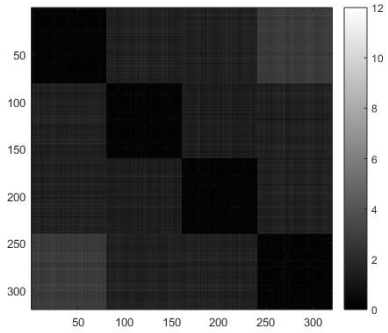


Fig. (a)

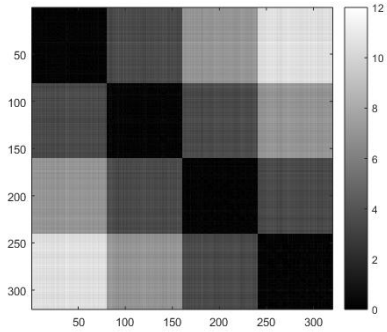


Fig. (b)

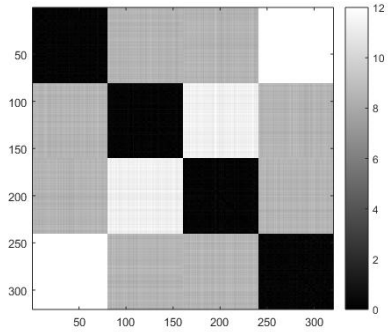


Fig. (c)

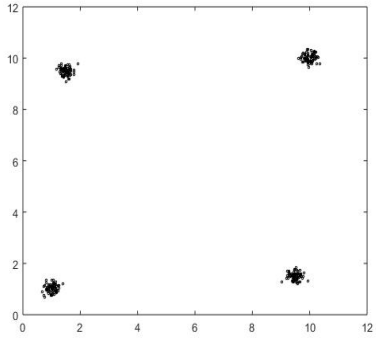


Fig. (d)

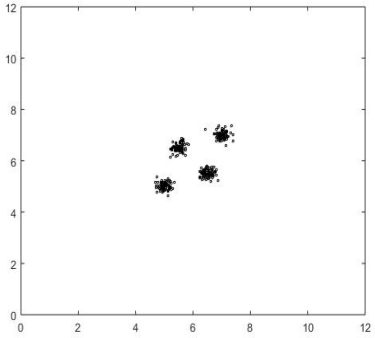


Fig. (e)

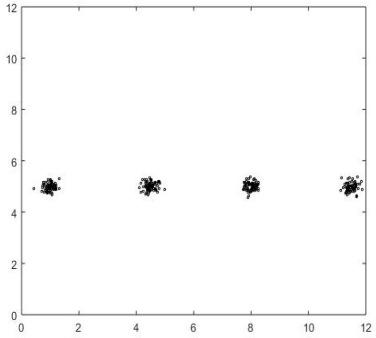


Fig. (f)

The distance matrix	Corresponding dataset ((d), (e) or (f))
Fig. (a)	(e)
Fig. (b)	(f)
Fig. (c)	(d)

10. In the figure below, match the similarity matrices, which are sorted according to cluster labels, with the sets of points. Differences in shading and marker shape distinguish between clusters, and each set of points contains 1500 points and four clusters, each of equal size.

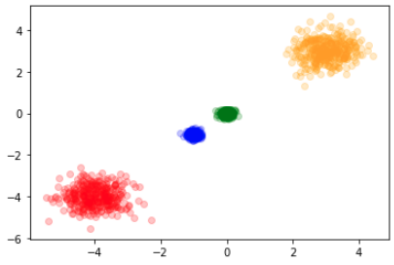


Fig (a)

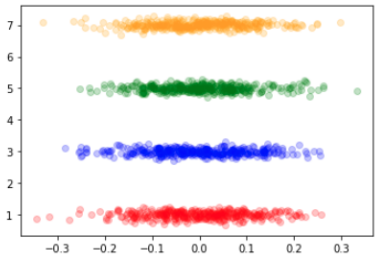


Fig (b)

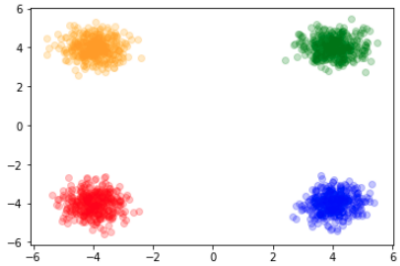


Fig (c)

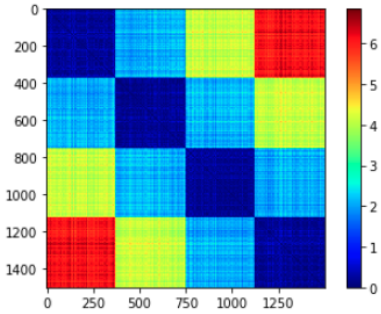


Fig (1)

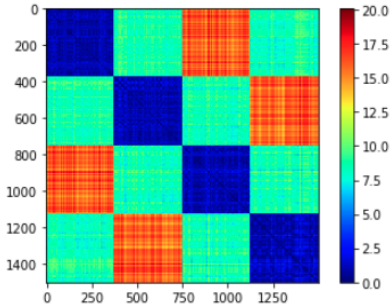


Fig (2)

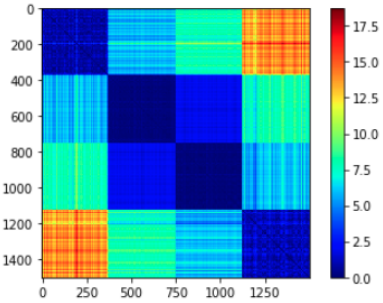
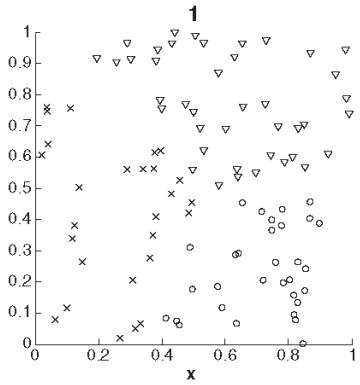


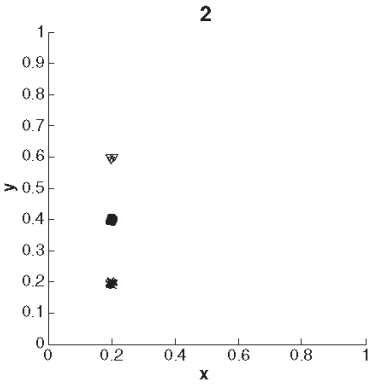
Fig (3)

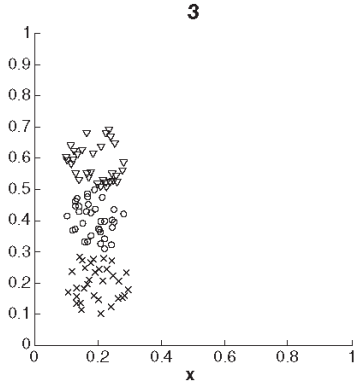
The distance matrix	Corresponding dataset ((d), (e) or (f))
Fig. (a)	3
Fig. (b)	1
Fig. (c)	2

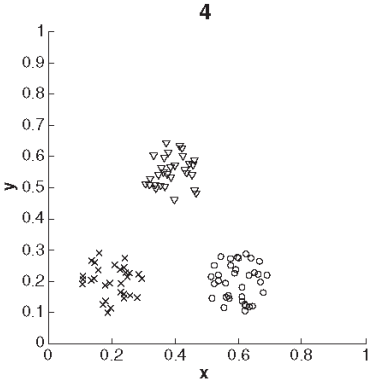
11. In the figure below, match the similarity matrices, which are sorted according to cluster labels, with the sets of points. Differences in shading and marker shape distinguish between clusters, and each set of points contains 100 points and three clusters, each of equal size.

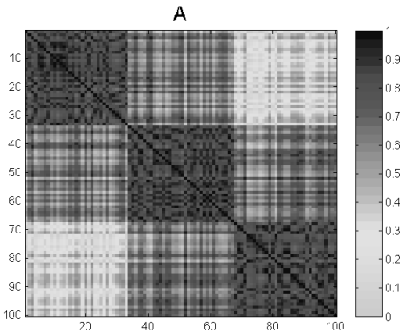
Set of Points	Similarity Matrix
1	D
2	C
3	A
4	B

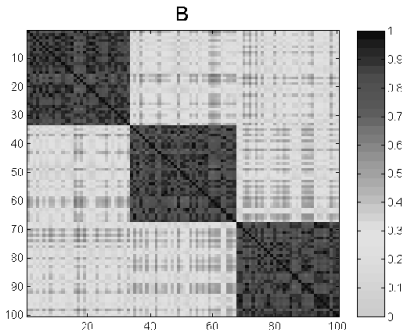


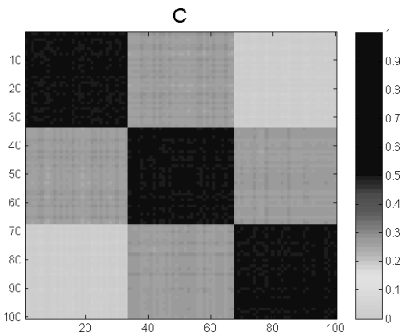


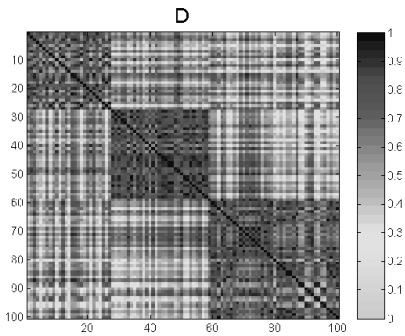












Prototype-Based Clustering

1. For this question, please refer to Fig 8.1 in the book. This figure shows the membership scores computed by fuzzy c-means clusters of a two-dimensional data set. Suppose you defined the core and border points based on the membership assignment of Fuzzy C-Means clustering using the following function:

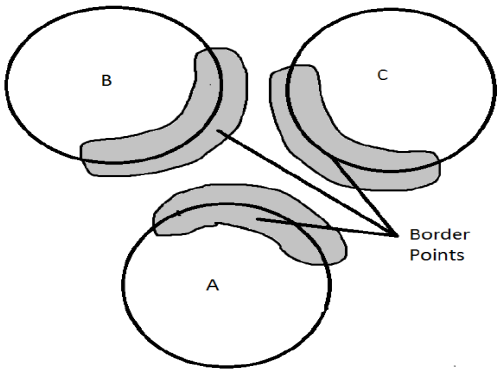
   if $\max_{i \epsilon C}(m_i) >=$ theta, then core point
   else border point.

Here m_i is the membership function of a point to the i^{th} cluster, and C denotes the total number of clusters. For example, a point with membership scores (0.6,0.3,0.1) will be a border point for theta=0.8, and a point with membership scores (0.8,0.1,0.1) will be a core point.

a) Is the above scheme able to find all the core points that are found by DBSCAN for some EPS and MinPts? You can vary theta to find them.
b) Is the above scheme able to find all the border points that are found by DBSCAN for a particular EPS, MinPts? You can vary theta to find them.

Justify your answer in both cases.

First, we choose a reasonable theta and MinPts so, all the points that are closer to the centroids of the three circles will be identified as core points, and all the points toward the periphery will be identified as border points. If you vary the parameters, the number of core and border points will vary. However, the question is whether you can identify the core and border points by this naïve scheme for any theta, no matter how you choose your parameters for DBSCAN.



a) Core points will be identified by reasonable theta.

b) Some border points will be identified, but the rest will not be. For example, in the figure, among all the border points, you can identify only the points that are close to another cluster (marked by shaded regions). The reason is that the membership scores for these points are distributed among three clusters, so the max of them will be less. But, for other points, the max is 1. So, they cannot be identified.

Density-Based Clustering

1. Here is a summary of the point counts for cells of a 6-by-6 grid. Note that the letters at the left and top are used to name rows and columns and are not part of the array. Assume that cells are considered adjacent only if they touch horizontally or vertically – not if they are diagonally connected.

	a	b	c	d	e	f
A	0	0	1	2	1	1
B	1	7	10	3	5	2
C	0	4	6	8	1	0
D	2	5	2	9	7	0
E	1	8	1	7	6	2
F	0	1	2	3	2	1

DENCLUE finds the local density maxima and then assigns each cell to the local maximum that it can reach by following the direction of maximum increase in density. Use this approach to partition the cells into clusters, and then answer the following questions:

Answer for partitioning the cells into groups.

	a	b	c	d	e	f
A	0	0	1	2	1	1
B	1	7	10	3	5	2
C	0	4	6	8	1	0
D	2	5	2	9	7	0
E	1	8	1	7	6	2
F	0	1	2	3	2	1

i) List all the local density maxima. (Use row letter + column letter to refer the cell, such as Ab, Ce, Ec, etc.)

Bc, Be, Dd, and Eb

ii) In the list below, which pairs of cells belong to the same density maxima?
a) Cb and Da
b) Bd and Be

c) Cd and Ff
d) Be and Ce
e) Ca and Ad

c) and e)

iii) To find clusters from the groups of points associated with a density maxima, we need to specify a density threshold, T. The number of clusters depends on the threshold T for the point count of a "dense" cell. A density maximum becomes a cluster if it is equal to or above the threshold. Two density clusters are merged if there is a path connecting the density maxima of each cluster where the points in the path have a density greater than or equal to the density Threshold.

How many clusters are there for T= 3 and T = 9?

There is one cluster for T = 3 and two clusters for T= 9.

Graph-Based Clustering

1. SNN Similarity Calculation

Point	First Neighbor	Second Neighbor
1	4	3
2	3	4
3	4	2
4	3	1

Table 1: Two nearest neighbors of four points

Calculate the SNN similarity between each pair of points using the definition of SNN similarity defined in Algorithm 8.10 in the book.

Pairs	SNN Similarity
1,2	0
1,3	0
1,4	1
2,3	1
2,4	0
3,4	0.

2. Use k = 3 in the definition of SNN similarity to answer the following questions:

a) What is the minimum possible value of SNN similarity between two points? 0

b) What is the maximum possible value of SNN similarity between two points?
2. You can't get 3 shared nearest neighbors because each point must appear on each other's nearest neighbor list, or else the similarity is 0.

Point	First Neighbor	Second Neighbor	Third neighbor
1	4	3	2
2	3	4	6
3	4	2	5
4	3	1	2
5	1	6	3
6	1	2	3

Table of three nearest neighbors of six points

c) The above table lists the three nearest neighbors of six points. Calculate the SNN similarity for the following pairs of points using the definition of SNN similarity defined in Algorithm 9.10 in the book (Use k = 3)

i. 1 and 4

2

d) 3 and 6

0

3. For the definition of SNN similarity provided in Algorithm 8.11 in the book, the calculation of SNN distance does not take into account the weights of the edges connecting the two points to their shared neighbors. In other words, it might be more desirable to give higher similarity to two points that are connected to their shared neighbors by edges having higher weights, as compared to two points that are connected to their shared neighbors by edges having lower weights.

(a) Describe how you might modify the definition of SNN similarity to give higher similarity to points whose shared neighbors are connected to them by edges having higher weights.
Answer: This can be done by considering the weights of the edges into the similarity computation. For example, instead of just adding 1 for each common neighbor, we can add a function, say min or mean, of the weights of the edges that connect the two points to this common neighbor. Adding up the value of this function for each common neighbor will give an edge weight incorporated SNN similarity measure between the two points.

(b) Discuss the advantages and disadvantages of such a modification.
Answer: Such an approach is more complex. Also, it is not easy to determine which function to use to combine the weights of the edges between the points and their common neighbors. However, this modification is advantageous if all the edges in the similarity graph don't have the same reliabilities, and using their weights gives a more accurate measurement of the similarities of points. It also reduces the

dependence of the results on the number of neighbors or similarity threshold chosen to construct the similarity graph.

4. List one advantage and one disadvantage of SNN similarity over direct similarity.

Advantage: good for high-dimensional data

Disadvantage: SNN similarity takes longer to compute, depends on parameter k

Comparisons of Clustering Algorithms

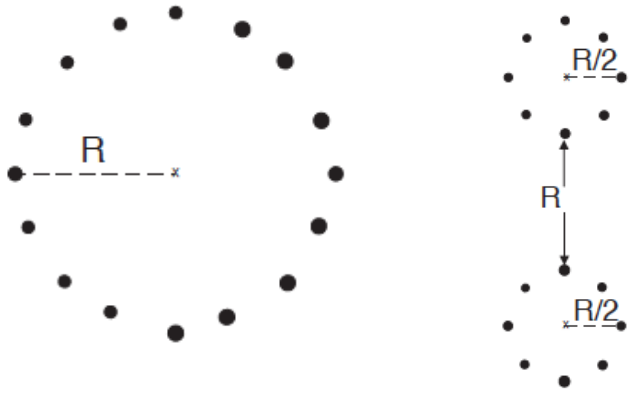
1. Discuss one advantage and disadvantage of fuzzy c-means with respect to the Gaussian mixture model (GMM) clustering algorithm.

   Advantage: Fuzzy c-means only estimates means, and thus it has fewer parameters than GMM, which estimates both means and the covariance matrix. This means that GMM becomes impractical for higher-dimensional data since the covariance matrix grows as n^2 where n is the number of variables.
   Disadvantage: GMM allows for a more general model that may fit the data better.

2. State one advantage and one disadvantage of the Gaussian mixture model clustering algorithm in comparison to the k-means clustering algorithm.

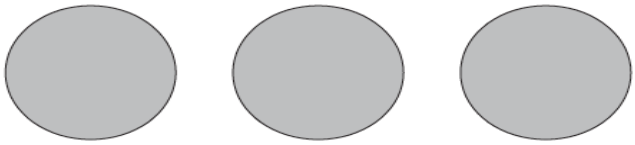
Advantage: GMM has a more general model which could fit the data better
Disadvantage: GMM estimates both the means and the covariance matrix. This means that GMM becomes impractical for higher-dimensional data since the covariance matrix grows as N^2, where N is the number of variables.

3. For the following data set, state which method will perform better, k-means, or bisecting k-means with k =3.



Bisecting K-means would perform better as it is more likely to separate the bigger cluster from the other two small clusters in the first iteration. And, in the second iteration, the cluster containing the two small clusters would be likely to be bisected into two small globular clusters of radius R/2. In contrast, K-means would face more initialization problems in getting the correct initialization of three centroids.

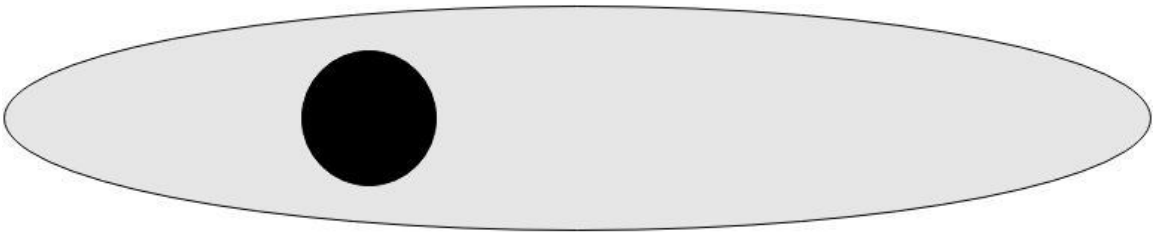
4. For the following data set, state which method will perform better (i.e., identify the three ellipses as three different clusters), k-means, or bisecting k-means (where k = 3).



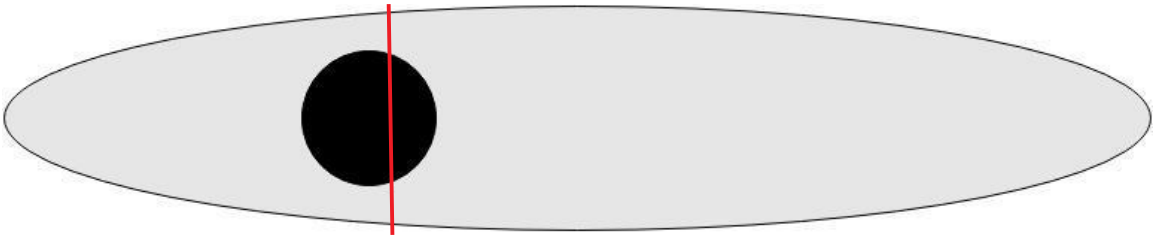
K-means would perform better because bisecting K-means would tend to split the middle cluster in the first iteration, especially if the distance between the two clusters is small compared to the radius of each cluster. Once the middle cluster is split, it can never be rejoined in successive iterations.

5. Comparison of K-means and Gaussian mixture models

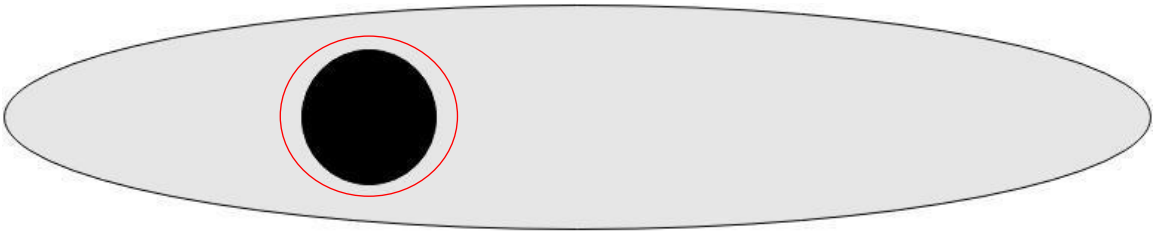
a) For the following set of two-dimensional points, sketch how they would be split into two clusters by K-means (when a global minimum of SSE is achieved) and by Gaussian mixture model clustering. Draw your answers on the figures below. You can assume the density of points in the darker area is much higher than the density of points in the lighter area.



K-means would split this cluster somewhat to one side of the middle of the denser cluster – see the red line. Mixture model clustering would probably do a reasonably good job of finding the two clusters containing the dense and less dense clusters. There is a red circle around the dense cluster, and the rest will be the less dense cluster.



K-means

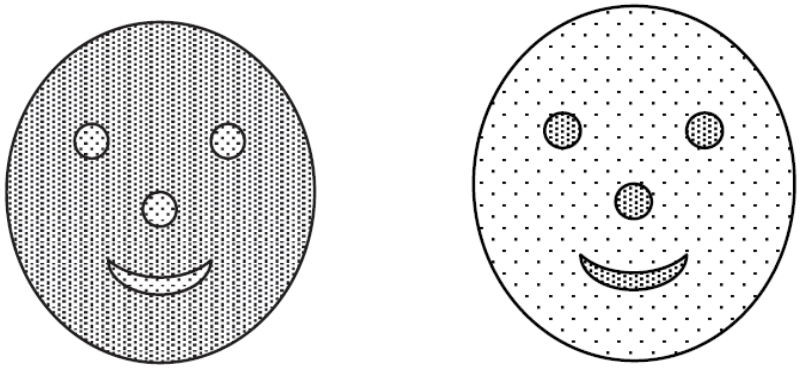


Gaussian Mixture Model

b) Name one other clustering method that might be able to accurately capture the two clusters.

SNN could likely find the two clusters, as is indicated by the "fish" dataset in the Chapter 7 slides. Chameleon should also perform well since it can handle differing densities.

6. Consider the following two sets of points (faces) shown in figures (a) and (b). The darker shading indicates a denser point distribution.



(a) (b)

a) For each figure, could you use DBSCAN to find clusters corresponding to the patterns represented by the nose, eyes, and mouth? Explain.
DBSCAN can work only for (b) because in (b) the points in the nose, eyes, and mouth are much closer together than the points between these areas, and DBSCAN could distinguish these areas. For (a), the noise is much denser than the interest patterns, so the nose, eyes, and mouth will be eliminated by DBSCAN.

b) For each figure, could you use K-means to find the patterns represented by the nose, eyes, and mouth? Explain.
K-means can work for (b) as long as the number of clusters was set to 4, although the lower density points would also be included. K-means does not work for (a).

a) For (a), could you figure out a clustering method, which can find the patterns represented by the nose, eyes, and mouth?

Take the reciprocal of the density as the new density and use DBSCAN.

7. Cluster comparison and choice of clustering approach.

(a) State two similarities and two differences between SNN density-based clustering and DENCLUE.

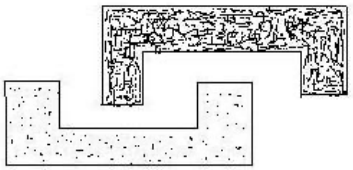
Similarities:
- Both rely on a density-based notion of clustering.
- Both are partitional clustering techniques.
- Both can handle clusters of different sizes and shapes.

Differences:
- SNN uses a graph-based view of the data, as well as density-based notions.
- SNN density-based clustering uses SNN density instead of regular density.
- DENCLUE uses kernel density functions to define density, SNN uses the DBSCAN notion
- DENCLUE assigns each point to the nearest density attractor (peak), while SNN links core points together to form clusters as in DBSCAN

(b) Algorithms such as SNN produce partial clusterings (i.e., they do not cluster all points). For what kind of problems is this feature of SNN particularly useful?

SNN is particularly useful in situations where the data set has noise and outliers since the SNN algorithm discards such points.

8. Consider the two blocks shown below. The upper block is denser than the lower block.



(a) Comment on whether the following clustering algorithms will be able to detect the two blocks as separate clusters:

i. single link

ii. complete link

iii. DBSCAN

iv. Chameleon

No explanation is required.

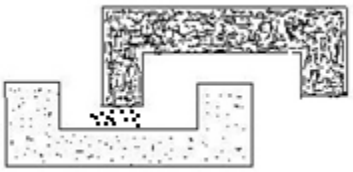
i. Yes (In the figure, the two blocks are well separated. Therefore, single link would be able to detect the two clusters)

ii. No (Would fail because of the non-globular shape of the blocks)

iii. Yes (Would work for a suitable choice of NumPts and Eps).

iv. Yes (The Relative closeness and Relative Interconnectivity of two blocks are very different. Therefore, blocks would not get merged.)

(b) For the following figure, assume that the points inside a boundary are denser than the points outside the boundary, which represent the noise points.



Between DBSCAN and Chameleon, which is likely to perform better at detecting the two blocks and why? Provide a brief explanation.
Chameleon because it can handle noise and differences in density. DBSCAN would work only if there is a choice of NumPts and Eps such that all noise points get eliminated and do not link the two blocks.

9. Comparison of clustering algorithms.

a. Compare EM (Gaussian mixture model clustering) and fuzzy c-means, and list one advantage for each algorithm.
Fuzzy c-means only estimates means and thus has fewer parameters than EM, which estimates both means and the covariance matrix. This means that EM becomes impractical for higher-dimensional data since the covariance matrix grows as N^2, where N is the number of variables. EM allows for a more general model that may fit the data better.

b. Discuss MAX and MIN agglomerative clustering for the following cases:
i) presence of noise

MIN is more affected here since it might merge clusters in the presence of noise

ii) different densities of clusters

Both MIN and MAX are not affected by different densities of clusters

iii) high dimensional data

Both MIN and MAX will not be affected by high dimensional data if the proximity function used can handle high dimensional data, e.g., cosine similarity for document data. If the proximity function does not handle high dimensional data very well, then both will perform poorly.

10. Similarities and differences of clustering algorithms.

a) a) List one similarity and one difference between SOM and k-means.

Similarities:
- K-means and SOM are center-based, partitional clustering techniques.
- Both have assignment, and update steps are the key actions of their algorithms.

Differences:
- SOM enforces a relationship between adjacent cluster centroids. This is useful for visualization of the clusters and understanding their relationships to one another. K-means does not.

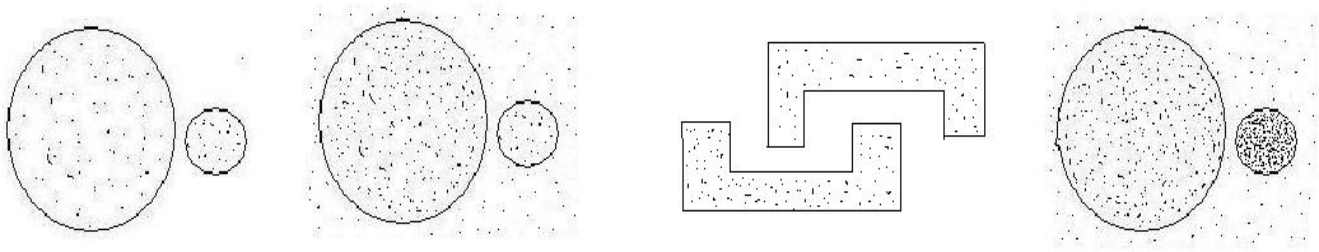
- SOM updates centroids incrementally. Most versions of K-means do not.
- The effect of a centroid update decreases with each iteration for SOM, but not for traditional K-means.

- SOM is not guaranteed to converge, i.e., does not converge to any local or global minimum necessarily.

b) List one similarity and one difference between bisecting k-means and hierarchical clustering using group average.

Similarity: both hierarchical, both minimize MSE
Difference: agglomerative vs. divisive/partitional

11. How will single-link, complete-link, and DBSCAN perform for the following cases? The cluster density is the same for the first three cases (a-c), while the last case (d) has one dense cluster with 50000 points and one relatively sparse cluster with 50 points with noise data points in between. Assume that the points inside a boundary are denser than the points outside the boundary, which represent the noise points (cases b and d).



(a) (b) (c) (d)

a) Single-link and DBSCAN will work well for this purpose, but complete-link doesn't because the two

clusters have very different sizes, and the distance between them is small.
b) DBSCAN will perform well, but single-link will not perform well, since it is susceptible to noise, and complete-link doesn't work well because of the size difference and nearness of the clusters.
c) Single-link and DBSCAN will perform well for this purpose, but complete-link will break the clusters into globular shapes.
d) DBSCAN will perform well if noise is less dense than the left cluster. MIN has problems with noise. MAX will put most of the points in the dense cluster into one cluster but will pick up noise points and points from the less dense cluster. MAX will include a lot of noise points in the less dense cluster, and this cluster will likely be broken since some points will be put with the dense cluster.

12. Clustering algorithm comparison.

(a) Explain one advantage and one disadvantage of DBSCAN over the K-means clustering algorithm.

Advantages: It is not necessary to specify the number of clusters in advance for DBSCAN. Also, DBSCAN can remove noise and outliers; thus, its clustering is generally less affected by such points.
Disadvantages: The computational efficiency of DBSCAN is worse than that of K-means, particularly for high-dimensional data. Also, it is sometimes not very easy to set the Eps and MinPts parameters.

(b) List one similarity and one difference between EM and fuzzy c-means.

Both are prototype-based approaches, and both estimate the mean. Fuzzy c-means only estimates means and thus has fewer parameters than EM (Gaussian mixture model clustering), which estimates both means and the covariance matrix. This means that EM become impractical for higher-dimensional data since the covariance matrix grows as n^2 where n is the number of variables. EM allows for a more general model that may fit the data better.

(c) List one similarity and one difference between SNN density-based clustering and DBSCAN.
Similarities:
- Both rely on a density-based notion of clustering.
- Both are partitional clustering techniques.
- Both can handle clusters of different sizes and shapes.
- Both are incomplete clustering techniques.
Differences:
- SNN uses a graph-based view of the data, as well as density-based notions.
- SNN density-based clustering uses SNN density instead of regular density.
- DBSCAN cannot handle data with differing densities.
- DBSCAN cannot handle high-dimensional data

(d) Discuss MAX and MIN agglomerative clustering for the following cases:
- presence of noise
MAX would be better

- different densities of clusters
MIN will identify the two clusters, since it can separate the two clusters as it would first find the dense cluster, and then it would form the low-density cluster. For MAX, it depends a lot on the shortest distance between the boundaries of the two clusters and the relative ratios of the densities of the two clusters. Besides, MAX has problems when one cluster is bigger, and the other is smaller, and they are close.

(d) Discuss k-means and AVG agglomerative clustering for the following cases:
- presence of outliers
K-means would be more affected by the presence of outliers
AVG can work with categorical data with the appropriate similarity measure for categorical data, but k-means cannot.

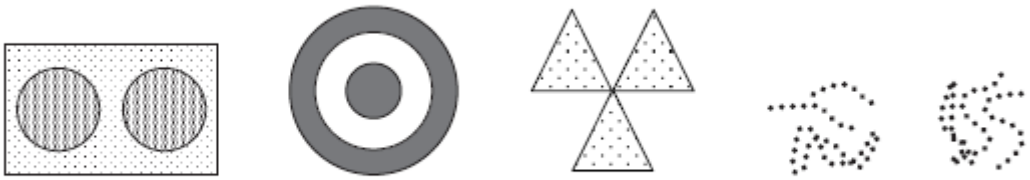
4. Identify the clusters in the following Figure using the center-, contiguity-, and density-based definitions. Also, indicate the number of clusters for each case and give a brief indication of your reasoning. Note that darkness or the number of dots indicates density. If it helps, assume center-based means K-means, contiguity-based means single link, and density-based means DBSCAN.

(a) (b)

(a) Center-based 2 clusters. The rectangular region will be split in half. Note that the noise is included in the two clusters.
Contiguity-based 1 cluster because the two circular regions will be joined by noise.
Density-based 2 clusters, one for each circular region. Noise will be eliminated.

(b) Center-based 2 clusters. The two groups of lines will be split in two.
Contiguity-based 5 clusters. Each set of lines that intertwines becomes a cluster.
Density-based 2 clusters. The two groups of lines define two regions of high density separated by a region of low density

13. Identify the clusters in the following Figure using the center-, contiguity-, and density-based definitions. Also, indicate the number of clusters for each case and give a brief indication of your reasoning. Note that darkness or the number of dots indicates density. If it helps, assume center-based means K-means, contiguity-based means single link, and density-based means DBSCAN.



(a) (b) (c) (d)

Solution:
(a)

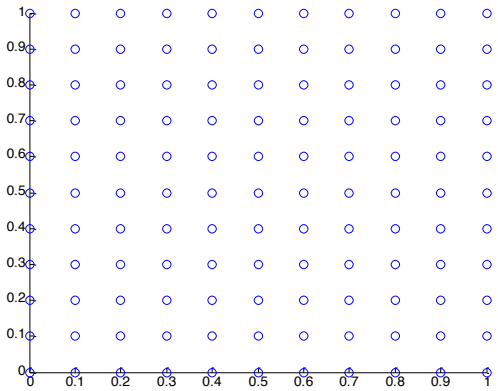
center-based: 2 clusters. The rectangular region will be split in half. Note that the noise is included in the two clusters.
contiguity-based: 1 cluster because the two circular regions will be joined by noise.
density-based 2 clusters, one for each circular region. Noise will be eliminated.

(b)
center-based: 1 cluster that includes both rings.
contiguity-based: 2 clusters, one for the ring and the other for the middle circle.
density-based: 2 clusters, one for the ring and the other for the middle circle.

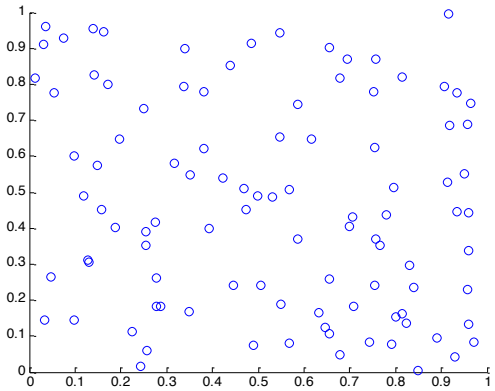
(c)
center-based: 3 clusters, one for each triangular region. One cluster is also an acceptable answer.
contiguity-based: 1 cluster. The three triangular regions will be joined together because they touch.
density-based: 3 clusters, one for each triangular region. Even though the three triangles touch, the density in the region where they touch is lower than throughout the interior of the triangles.

(d)
center-based 2 clusters: The two groups of lines will be split in two.
contiguity-based: 5 clusters. Each set of lines that intertwines becomes a cluster.
density-based: 2 clusters. The two groups of lines define two regions of high density separated by a region of low density.

14. Suppose you are given two sets of 100 points that fall within the unit square. One set of points (a) is arranged so that the points are uniformly spaced. The other set of points (b) is randomly generated from a uniform distribution over the unit square.



a.



b.

(i) How does the behavior of DBSCAN differ on (a) and (b)?
On (a), DBSCAN will either place all the points in one cluster or will produce no clusters, i.e., all points

are noise, depending on its parameter settings. On (b), DBSCAN will produce the dense regions, if any, as clusters.

(ii) How does the behavior of DBSCAN and K-means differ on (b)?
K-means will find K clusters. DBSCAN will produce the dense regions, if any, as clusters, depending on the parameter settings.

15. Answer the following questions.

a) State two similarities and two differences between SNN density-based clustering and DENCLUE.

b) Explain two major advantages of Chameleon over hierarchical clustering using group average.

c) Whenever two clusters are merged, the resulting SSE after merging is always greater than or equal to the SSE before merging. True or False? Explain briefly.

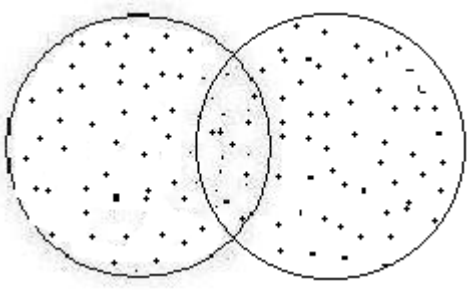
a)
Similarities:
- Both rely on a density-based notion of clustering.
- Both are partitional clustering techniques.
- Both can handle clusters of different sizes and shapes.

Differences:
- SNN uses a graph-based view of the data, as well as density-based notions.
- SNN density-based clustering uses SNN density instead of regular density.
- DENCLUE uses kernel density functions to define density, while SNN uses the DBSCAN notion but with SNN similarity.
- DENCLUE assigns each point to the nearest density attractor (peak), while SNN links core points together to form clusters as in DBSCAN

b) 1st Advantage: Chameleon can find clusters of varying densities.
2nd Advantage: Chameleon can find clusters of varying sizes and shapes. Group average favors globular clusters.

c) True, under the assumption that the new centroid of the resulting cluster is determined with the objective to minimize SSE. False, otherwise.

16. Among the three clustering algorithms-K-Means, fuzzy c-Means, and Single-link, which one will perform the best and worst to find two clusters from the following case and why?



Single link is worst since it will produce just one cluster. Fuzzy c-means is best since it will produce two clusters and give roughly equal cluster membership weights for the points in the overlapping region. K-means also finds two clusters but splits the boundary points.

17. Differences and similarities of clustering methods.

a) State one similarity and one difference between K-means and SOM.

Similarities:

- K-means and SOM are center-based, partitional clustering techniques.
- Both have assignment and update steps that are the key actions of their algorithms.

Differences:

- SOM enforces a relationship between adjacent cluster centroids. K-means does not.
- SOM updates centroids incrementally. Most versions of K-means do not.
- The effect of a centroid update decreases with each iteration for SOM, but not for traditional K-means.
- While SOM is often used to visualize the data, K-means is not.

b) State two similarities and two differences between Fuzzy c-means and EM clustering based on Gaussian distributions.

Similarities:

- Both are partitional clustering schemes.
- Both find centroids.
- Both use weights, i.e., points are assigned to all clusters with some weight.
- Both use an “assign” and “update” methodology.
- Both require an estimate of the number of clusters.

Differences:

- Fuzzy c-means is based on fuzzy set theory, EM on probability theory.
- EM can find elliptical clusters of differing spread. Fuzzy c-means finds clusters much like those of traditional K-means and prefers globular clusters.
- EM is better at finding clusters of different density because of the covariance parameter.
- The covariance matrix parameter of EM provides additional descriptive information about the clusters found.

- The number of parameters grows as n^2 where n is the number of dimensions, for EM.

c) State two similarities and two differences between SNN density-based clustering and DENCLUE.

Similarities:

- Both rely on a density-based notion of clustering.

- Both are partitional clustering techniques.

- Both can handle clusters of different sizes and shapes.

Differences:

- SNN uses a graph-based view of the data, as well as density-based notions.

- SNN density-based clustering uses SNN density instead of regular density.

- DENCLUE uses kernel density functions to define density, SNN uses the DBSCAN notion

- DENCLUE assigns each point to the nearest density attractor (peak), while SNN links core points together to form clusters as in DBSCAN

18. Clustering algorithms comparison.

a) State one advantage and one disadvantage of SOM over the K-means clustering algorithm
b) State one advantage and one disadvantage of K-means clustering over hierarchical clustering with group average.

a) Advantages of SOM:
- SOM clustering takes into account the relationship between the clusters
- SOM allows for visualization of the clusters if the SOM grid is 2D
- SOM can be regarded as a dimensionality reduction

Disadvantages of SOM:
- SOM requires more computation
- Doesn't represent a local minimum of SSE

b) Advantage of k-means:
- Time complexity is O(n). For Hierarchical time complexity is $O(n^2)$
- K-means has a centroid (prototype) that provides a compact representation of a cluster
- K-means aims to optimize a well-defined notion of cluster goodness (i.e., SSE)

Disadvantages of K-means:
- K-means can produce different clusterings for different runs. Group average does not except for randomness due to tie-breaking when different pairs of points have the same similarity.

- K-means can only be applied to data with a well-defined notion of a centroid. Group average can be applied to any data for which a proximity function can be applied.
- K-means is impacted by outliers, but group average is not affected significantly since outliers will remain as separate clusters until the highest levels of the hierarchy.

19. Characterizing clustering methods.

a) Describe the complete link clustering technique with respect to the following characteristics.

i. Ability to cluster large data sets.

Because complete link has a time complexity of O($n$log(n)) and space complexity of O(n^2), its ability to cluster large data sets is limited unless sampling or some other data reduction approach is used.

ii. Ability to find clusters of differing densities.

Complete link defines a cluster in terms of the maximum distance between two points in a cluster. Thus, and thus, complete link can find clusters of differing densities provided that other factors, e.g., differing sizes and non-globular shapes, do not interfere.

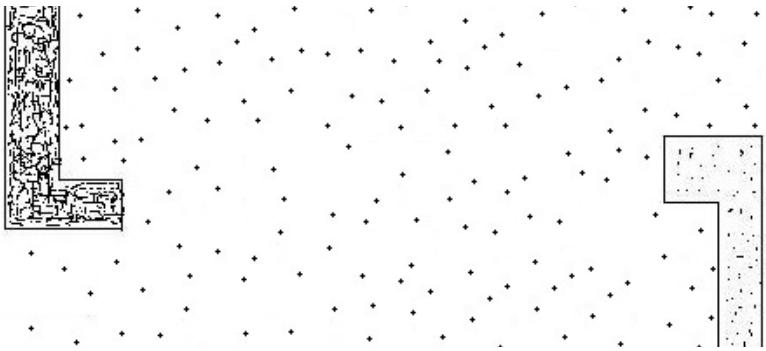
b) List one advantage and one disadvantage of SNN similarity over direct similarity.

Advantage: good for high-dimensional data

Disadvantage: SNN similarity takes longer to compute and depends on the parameter k, i.e., the number of nearest neighbors.

20. What challenges are faced by complete-link and DBSCAN, respectively, for the following case? Between the two algorithms, which one will you prefer for this specific data set? And why?

The upper block is denser than the lower block. Also, assume that the points inside a block are denser than the points outside the block, which represent the noise points.



Challenge faced by complete link:	Challenge faced by DBSCAN:
Since the two blocks are far from each other, complete link can separate them very well. But due to complete-link's bias toward globular shapes, it will assign some surrounding noise points to the clusters. In addition, some noise points will form one or more clusters well before all of the points in the two blocks become part of their respective clusters	It's hard to determine MinPts initially, i.e., the value for k when looking at the behavior of k-dist (refer to textbook Page 529). If k is too small, some noise points located closely to the blocks would be labeled as clusters; If it is too large, there is a risk that the two clusters will be merged via noise points between them. Note: The different densities of the two blocks do not create any trouble for DBSCAN, and the light block on the right side will not be considered as noise since it's denser than noise.
Your preference and the reason: For this specific case, DBSCAN should be preferred. Because with appropriate parameters (Eps & MinPts), DBSCAN could distinguish the two clusters clearly, whereas complete link will always involve noise in the clusters. To determine a good value for Eps, we can change the value of k and look at the behavior of corresponding k-dist. The k-th nearest neighbor distance for noise points will change dramatically as k changes, but it doesn't for cluster points.	

21. Different clustering methods have different characteristics. In this question, you are asked to identify the characteristics of each of the following clustering methods: DBSCAN, MAX, CLIQUE (subspace clustering), and SNN.

a) Are the given clustering methods complete or not?

	The clustering method is **complete (Yes/No)**
DBSCAN	No
MAX	Yes
CLIQUE	No
SNN	No

b) Is the given clustering method capable of handling clusters of varying densities in the data? Assume that in addition to meaningful clusters, the data also has noise points. This noise is of lower density relative to all clusters near it. Note that the noise in a region of high-density clusters could have a greater density relative to low-density clusters in a different region.

	The clustering method can handle the presence of clusters of varying densities **(Yes/No)**
DBSCAN	No
MAX	No
CLIQUE	No
SNN	Yes

c) Can the given clustering method handle/tolerate noise? (i.e., can it detect clusters in the presence of noise). Assume that all clusters in the data are of the same density, which is higher than the density of noise points.

	The clustering method **can** handle/ is tolerant of noise? **(Yes/No)**
DBSCAN	Yes
MAX	Yes
CLIQUE	Yes
SNN	Yes

22. Different clustering methods have different characteristics. In this question, you are asked to identify the characteristics of each of the various clustering methods. Answer Yes or No. **No explanation needed**.

a) Can the given clustering method handle/tolerate noise? (i.e., can it detect clusters in the presence of noise). (yes or no)

K-means yes
MAX yes
MIN no
SNN yes

b) Is the given clustering method capable of handling clusters of varying densities in the data? (Yes or No)

(Assume that in addition to meaningful clusters, the data also has noise points. This noise is of lower density relative to all clusters near it. Note that the noise in a region of high-density clusters could have a greater density relative to low-density clusters in a different region.)

Chameleon yes
EM (Gaussian mixture model) yes
SNN yes
DBSCAN no

c) Is the given clustering algorithm suitable for finding clusters of shapes that are not globular?

MIN yes
Chameleon yes
Group Average no
DENCLUE yes

d) Is the clustering method complete? (Yes/No)

SNN no
Fuzzy c-mean yes
Chameleon yes
CLIQUE (subspace clustering) no

23. Different clustering methods have different characteristics. In this question, you are asked to identify the characteristics of various clustering methods. Answer Yes or No. No explanation needed.

a) Is the clustering method complete? (Yes/No)
SNN no
Fuzzy c-mean yes
Chameleon no
CLIQUE (subspace clustering) no

b) Is the given clustering method capable of handling clusters of varying densities in the data? (Yes or No)
(Assume that in addition to meaningful clusters, the data also has noise points. This noise is of lower density relative to all clusters near it. Note that the noise in a region of high-density clusters could have a greater density relative to low-density clusters in a different region.)
Chameleon yes
EM (Gaussian mixture model) yes
SNN yes
DBSCAN no

c) Can the given clustering method handle/tolerate noise? (i.e., can they detect clusters in the presence of noise). (yes or no)
K-means yes
MAX yes
MIN no
SNN yes

d) Is the given clustering algorithm suitable for finding clusters of shapes that are not globular?
MIN yes
Chameleon yes
Group Average no
DENCLUE yes