By Akash,

1. Discuss the strengths and weaknesses of hierarchical clustering algorithms.

Answer:

Strengths:

- Do not have to assume any particular number of clusters – Any desired number of clusters can be obtained by 'cutting' the dendogram at the proper level
- They may correspond to meaningful taxonomies – Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction,)

Weaknesses:

- do not scale well: time complexity of at least O(n2), where n is the total number of objects n ..
- Once a merging or splitting decision has been made, there exists no facility to rectify a mistake at a later stage.
- Different schemes have problems with one or more of the following: –
  1. Sensitivity to noise and outliers –
  2. Difficulty handling different sized clusters

2. Discuss the strengths and weaknesses of partitioning clustering algorithms.

Answer:

Strengths:

- Fast, easy to implement.
- "Good enough" in a wide variety of tasks and domains.
- The global optimum may be found using techniques such as: deterministic annealing and genetic algorithms

Weaknesses:

- Must pre-specify number of clusters k.
- Highly sensitive to choice of initial clusters.
- Assumes that each cluster is spherical in shape and data examples are largely concentrated near its centroid.
- Unable to handle noisy data and outliers.
- Not suitable to discover clusters with non-convex shapes.

3. Are the *k-means* and *k-medoids* algorithms part of partitioning or hierarchical clustering techniques? Justify your answer.

Answer:

The k-means and k-medoids algorithms are a part of partitioning clustering technique. Partitioning clustering is defined as a division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset. Both the *k*-means and *k*-medoids algorithms are partitional (breaking the dataset up into groups) and both attempt to minimize the distance between points labeled to be in a cluster and a point designated as the center of that cluster. K-means and k-medoids have a similar concept where the end result provides two definite clusters clearly differentiable from the other. k-medoid is based on centroids (or medoids) calculating by minimizing the absolute distance between the points and the selected centroid, rather than minimizing the square distance. As a result, it's more robust to noise and outliers than k-means.

4. How do k-means and k-medoids compare to the AGNES algorithm?
Answer:

K-means and k-medoids are partitioning algorithms whereas AGNES is a category of the hierarchical partitioning algorithm

Agglomerative: Begin with each item assigned to its own cluster. Apply a bottom-up strategy where, at each step, the most similar pair of clusters are merged.

Both k-means and k-medoids algorithms are breaking the dataset up into k groups. Also, they are both trying to minimize the distance between points of the same cluster and a particular point which is the center of that cluster

Time Complexity
K-means and PAM is linear in the number of data objects i.e. $O(n)$, where n is the number of data objects. The time complexity of AGNES algorithm is quadratic i.e. $O(n2)$.Therefore, for the same amount of data, AGNES will take quadratic amount of time.

Shape of Clusters
K-means and PAM work well when the shape of clusters are hyper-spherical (or circular in 2 dimensions). If the natural clusters occurring in the dataset are non-spherical then probably K-means is not a good choice.

Repeatability
K-means starts with a random choice of cluster centers, therefore it may yield different clustering results on different runs of the algorithm. Thus, the results may not be repeatable and lack consistency. However, with AGNES algorithm, you will most definitely get the same clustering results.

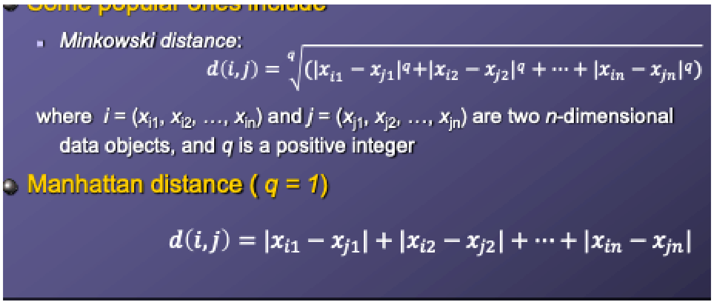
5. How the similarity measure is defined in the k-means algorithm? (5 marks)
Answer:

The similarity measure can be found out by using distance as a measure for finding out the similarity between the cluster points. Distance can be measured by

1- Manhattan distance

2- Euclidean distance

3-Minkowski distance



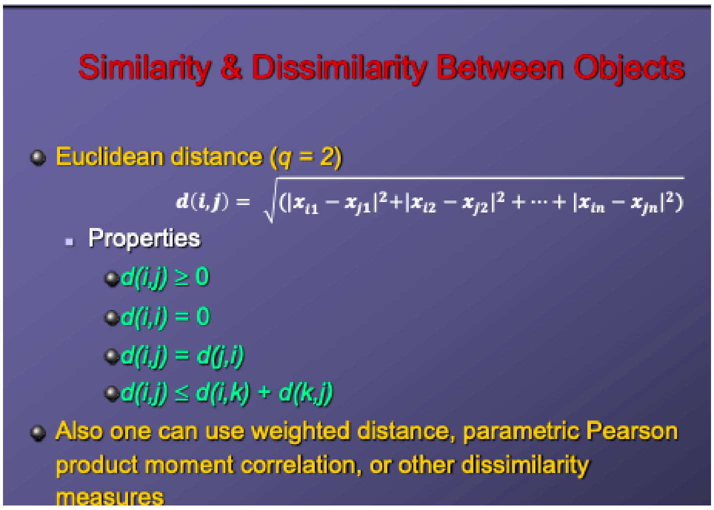
- Some popular ones include
  - *Minkowski distance*:

$$d(i,j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \cdots + |x_{in} - x_{jn}|^q)}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{in})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jn})$ are two n-dimensional data objects, and q is a positive integer

- Manhattan distance ($q = 1$)

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{in} - x_{jn}|$$



Similarity & Dissimilarity Between Objects

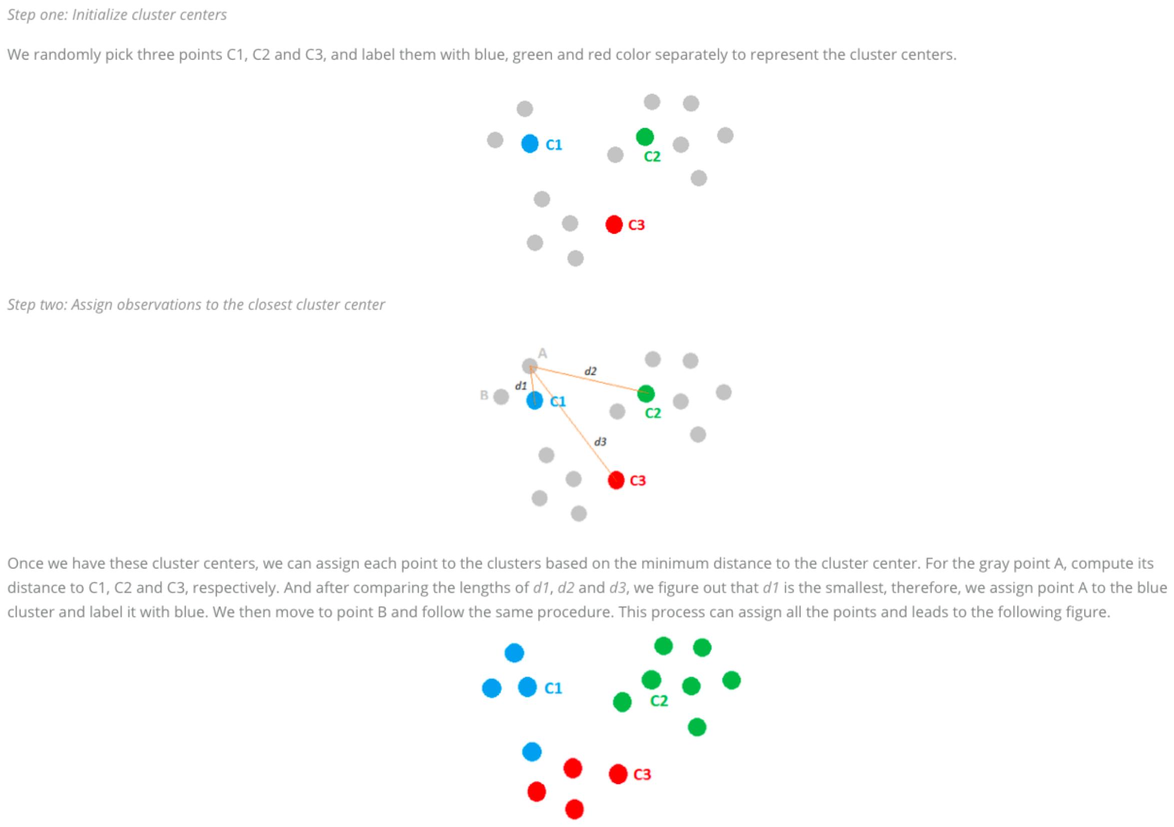
- Euclidean distance ($q = 2$)

$$d(i,j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{in} - x_{jn}|^2)}$$

  - Properties
    - $d(i,j) \geq 0$
    - $d(i,i) = 0$
    - $d(i,j) = d(j,i)$
    - $d(i,j) \leq d(i,k) + d(k,j)$
- Also one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

6. Explain each of the four steps of the k- means algorithm. (10 marks)
Answer:

1. Initialisation: Select k initial cluster centroids (e.g. at random)
2. Assignment step: Assign every item to its nearest cluster centroid (e.g. using Euclidean distance).
3. Update step: Recompute the centroids of the clusters based on the new cluster assignments, where a centroid is the mean point of its cluster.
4. Go back to Step 2, until when no reassignments occur (or until a maximum number of iterations is reached).

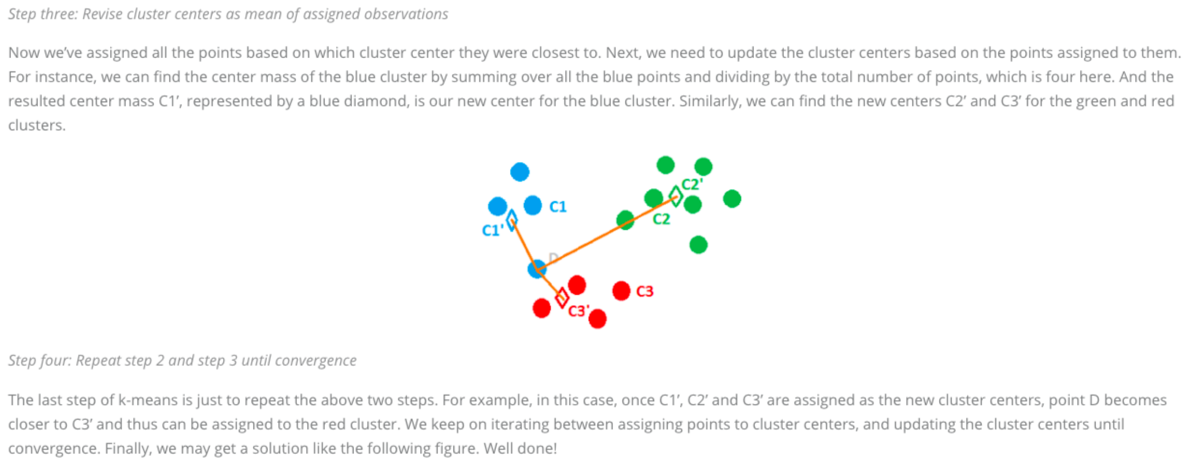


Step one: Initialize cluster centers

We randomly pick three points C1, C2 and C3, and label them with blue, green and red color separately to represent the cluster centers.

Step two: Assign observations to the closest cluster center

Once we have these cluster centers, we can assign each point to the clusters based on the minimum distance to the cluster center. For the gray point A, compute its distance to C1, C2 and C3, respectively. And after comparing the lengths of *d1*, *d2* and *d3*, we figure out that *d1* is the smallest, therefore, we assign point A to the blue cluster and label it with blue. We then move to point B and follow the same procedure. This process can assign all the points and leads to the following figure.



Step three: Revise cluster centers as mean of assigned observations

Now we've assigned all the points based on which cluster center they were closest to. Next, we need to update the cluster centers based on the points assigned to them. For instance, we can find the center mass of the blue cluster by summing over all the blue points and dividing by the total number of points, which is four here. And the resulted center mass C1', represented by a blue diamond, is our new center for the blue cluster. Similarly, we can find the new centers C2' and C3' for the green and red clusters.

Step four: Repeat step 2 and step 3 until convergence

The last step of k-means is just to repeat the above two steps. For example, in this case, once C1', C2' and C3' are assigned as the new cluster centers, point D becomes closer to C3' and thus can be assigned to the red cluster. We keep on iterating between assigning points to cluster centers, and updating the cluster centers until convergence. Finally, we may get a solution like the following figure. Well done!

Some Additional Remarks about K-means

- The k-means algorithm converges to local optimum. Therefore, the result found by K-means is not necessarily the most optimal one.
- The initialization of the centers is critical to the quality of the solution found. There is a smarter initialization method called K-means++ that provides a more reliable solution for clustering.
- The user has to select the number of clusters ahead of time.

10 marks in your bag ☺

7. What are the stopping conditions of the k-means algorithm? (5 marks)
Answer:

The common stopping conditions I have seen:

1. Convergence. (No further changes)
2. Maximum number of iterations.
3. Variance did not improve by at least x
4. Variance did not improve by at least x * initial variance

If you use MiniBatch k-means, it will not converge, so you need one of the other criteria. The usual one is the number of iterations.

8. What are the strengths of the k-means algorithm? (5 marks)

Answer:

- If variables are huge, then K-Means most of the times computationally faster than hierarchical clustering, if we keep k small. Easy to use.
- K-Means produce tighter clusters than hierarchical clustering, especially if the clusters are globular.
- Relatively efficient: 0(tkn), where n, k, and t are the number of objects, number of clusters, and number iterations respectively. Normally, k,t << n
- Often terminates at a local optimum n.
- The global optimum may be found using techniques such as: deterministic annealing and genetic algorithms

9. What are the weaknesses of the k-means algorithm? (5 marks)
Answer:

- Must pre-specify number of clusters k.
- Highly sensitive to choice of initial clusters.
- Assumes that each cluster is spherical in shape and data examples are largely concentrated near its centroid.
- Unable to handle noisy data and outliers.
  example:
  • 1,3,5,7,9 MeanCenter: 5
  • 1,3,5,7,1009 MeanCenter: 205
- Not suitable to discover clusters with non-convex shapes.

10. How k-means differs from k-medoids algorithm? (5 marks)
Answer:

- The k-medoids algorithm is a clustering algorithm related to the k-means algorithm and the medoidshift algorithm. Both the k-means and k-medoids algorithms are partitional (breaking the dataset up into groups).

- K-means attempts to minimize the total squared error, while k-medoids minimizes the sum of dissimilarities between points labeled to be in a cluster and a point designated as the center of that cluster. In contrast to the k-means algorithm, k-medoids chooses datapoints as centers (medoids or exemplars).
- K-medoids is also a partitioning technique of clustering that clusters the data set of n objects into k clusters with k known a priori. A useful tool for determining k is the silhouette.
- It could be more robust to noise and outliers as compared to k-means because it minimizes a sum of general pairwise dissimilarities instead of a sum of squared Euclidean distances. The possible choice of the dissimilarity function is very rich but in our applet we used the Euclidean distance.
- K-means works well for large datasets but PAM does not.

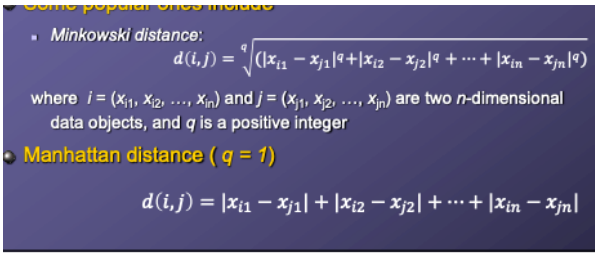
One of the key concept is any clustering algorithm is the similarity measure. Briefly outline how to compute the dissimilarity between objects described by the following types of variables(Questions 11 to 15)

11. Numerical (interval-scaled) variables. (5 marks)

Answer:

- Interval-scaled (numeric) variables are continuous measurements of a roughly linear scale.
- Examples – weight and height, latitude and longitude coordinates (e.g., when clustering houses), and weather temperature.
  Computing dissimilarity:

Distances are normally used to measure the similarity or dissimilarity between two data objects described by interval-scaled variables



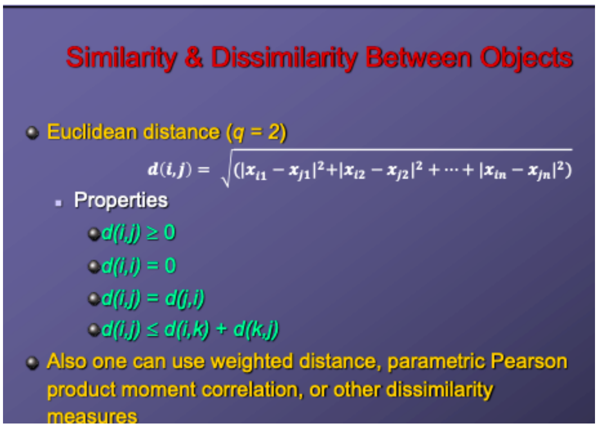
- Some popular ones include
  - *Minkowski distance*:

$$d(i,j) = \sqrt[q]{(|x_{i1}-x_{j1}|^q+|x_{i2}-x_{j2}|^q+\cdots+|x_{in}-x_{jn}|^q)}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{in})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jn})$ are two n-dimensional data objects, and q is a positive integer

- Manhattan distance ($q = 1$)

$$d(i,j) = |x_{i1}-x_{j1}| + |x_{i2}-x_{j2}| + \cdots + |x_{in}-x_{jn}|$$



Similarity & Dissimilarity Between Objects

- Euclidean distance ($q = 2$)

$$d(i,j) = \sqrt{(|x_{i1}-x_{j1}|^2+|x_{i2}-x_{j2}|^2+\cdots+|x_{in}-x_{jn}|^2)}$$

  - Properties
    - $d(i,j) \geq 0$
    - $d(i,i) = 0$
    - $d(i,j) = d(j,i)$
    - $d(i,j) \leq d(i,k) + d(k,j)$
- Also one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

12. Asymmetric binary variables. (5 marks)
Answer:

- A binary variable is asymmetric if the outcomes of the states are not equally important,
  – Example: the positive and negative outcomes of a HIV test.
  – we shall code the most important outcome, which is usually the rarest one, by 1 (HIV positive).
- Given two asymmetric binary variables, the agreement of two Types of Data in Cluster Analysis 1s (a positive match) is then considered more significant than that of two 0s (a negative match).
- The dissimilarity based on such variables is called asymmetric binary dissimilarity
- In asymmetric binary dissimilarity the number of negative matches, t, is considered unimportant and thus is ignored in the computation:

$$d(i, j) = \frac{r+s}{q+r+s}$$

13. Categorical variables/Nominal. (5 marks)
Answer:

- A categorical (nominal) variable is a generalization of the binary variable in that it can take on more than two states.
  – Example: map_color is a categorical variable that may have five states: red, yellow, green, pink, and blue.
- The states can be denoted by letters, symbols, or a set of integers.
- Dissimilarity between categorical variables:
- Method 1: Simple matching
  – The dissimilarity between two objects i and j can be computed based on the ratio of mismatches:

  $$d(i, j) = \frac{p-m}{p}$$

  – m is the number of matches (i.e., the number of variables for which i and j are in the same state)
  – p is the total number of variables.
  –Weights can be assigned to increase the effect of m or to assign greater weight to the matches in variables having a larger number of states.
- Method 2: use a large number of binary variables
  – creating a new asymmetric binary variable for each of the nominal states
  – For an object with a given state value, the binary variable representing that state is set to 1, while the remaining binary variables are set to 0.
  – For example, to encode the categorical variable map _color, a binary variable can be created for each of the five colors listed above.
  – For an object having the color yellow, the yellow variable is set to 1, while the remaining four variables are set to 0.

14. Ratio-scales variables. (5 marks)(Not sure)
Answers:

Ratio-scaled attributes are numeric attributes with an inherent zero-point. Measurements are ratio-scaled in that we can speak of values as being an order of magnitude larger than the unit of measurement.

- treat them like interval-scaled variables — not a good choice!
- apply logarithmic transformation yi = log(xi)
- treat them as continuous ordinal data and treat their rank as interval-scaled

15. Variables of mixed types. (5 marks)

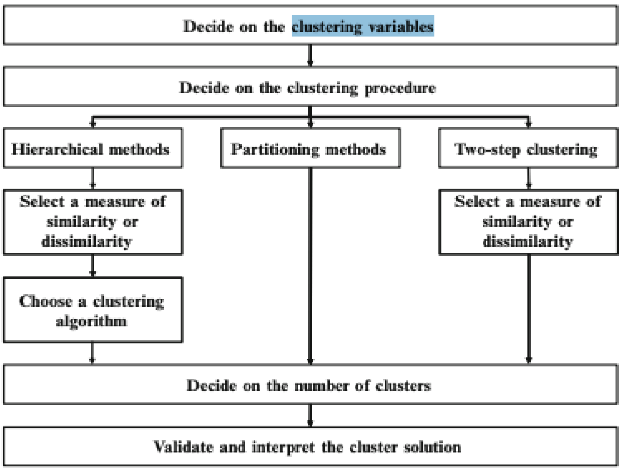
Answers:

- A database may contain different types of variables – interval-scaled, symmetric binary, asymmetric binary, nominal, and ordinal
- The contribution of variable f to the dissimilarity between i and j, that is, dij(f)
- If f is interval-based: – use the normalized distance so that the values map to the interval [0.0,1.0].
- If f is binary or categorical: Types of Data in Cluster Analysis – dij(f) = 0 if xif = xjf , or dij(f) = 1 otherwise
- If f is ordinal: – compute ranks rif

16. Discuss the three essential elements of a typical clustering solution. (5 marks)
Answers:

The 3 Essential Elements of a typical clustering solution are:

1. Which clustering variables should be included in the analysis.
2. To decide on the clustering procedure
3. Decide on the number of clusters
4. Ro interpret the solution by defining and labelling the obtained clusters.


Decide on the clustering variables
Decide on the clustering procedure
Hierarchical methods
Partitioning methods
Two-step clustering
Select a measure of similarity or dissimilarity
Select a measure of similarity or dissimilarity
Choose a clustering algorithm
Decide on the number of clusters
Validate and interpret the cluster solution

17. In the context of segmenting customers according to their behaviours, describe the roles that each element plays in such a clustering process. (10 marks)
Answer:

Let’s try to gain a basic understanding of the cluster analysis procedure by looking at a simple example. Imagine that you are interested in segmenting your customer base in order to better target them through, for example, pricing strategies.
The first step is to decide on the characteristics that you will use to segment your customers. In other words, you have to decide **which clustering variables will be included in the analysis**. For example, you may want to segment a market based on customers’ price consciousness (x) and brand loyalty (y). These two variables can be measured on a 7-point scale with higher values denoting a higher degree of price consciousness and brand loyalty. The objective of cluster analysis is to identify groups of objects (in this case, customers) that are very similar with regard to their price consciousness and brand loyalty and assign them into clusters.
After having decided on the clustering variables (brand loyalty and price consciousness), we need **to decide on the clustering procedure** to form our groups of objects. This step is crucial for the analysis, as different procedures require different decisions prior to analysis. There is an abundance of different approaches and little guidance on which one to use in
practice. We are going to discuss the most popular approaches in market research, as they can be easily computed using SPSS. These approaches are the following:
– Hierarchical methods,
– Partitioning methods (more precisely, k-means), and
– Two-step clustering.
In the final step, we need to **interpret the solution by defining and labeling** the obtained clusters. This can be done by examining the clustering variables’ mean values or by identifying explanatory variables to profile the clusters. Ultimately, managers should be able to identify customers in each segment on the basis of easily measurable variables. This final step also requires us to assess the clustering solution’s stability and validity.

18. What is meant by “scalable clustering algorithm”?(5 marks)
Answer:

Many clustering algorithms work well on small data sets containing fewer than several hundred data objects; however, a large database may contain millions or even billions of objects, particularly in Web search scenarios. Clustering on only a sample of a given large data set may lead to biased results. Therefore, highly scalable clustering algorithms are needed.

“How can we make the k*-means algorithm more scalable?”* One approach to making the *k*-means method more efficient on large data sets is to use a good-sized set of samples in clustering. Another is to employ a filtering approach that uses a spatial hierarchical data index to save costs when computing means. A third approach explores the micro clustering idea, which first groups nearby objects into “micro clusters” and then performs *k*-means clustering on the micro cluster.

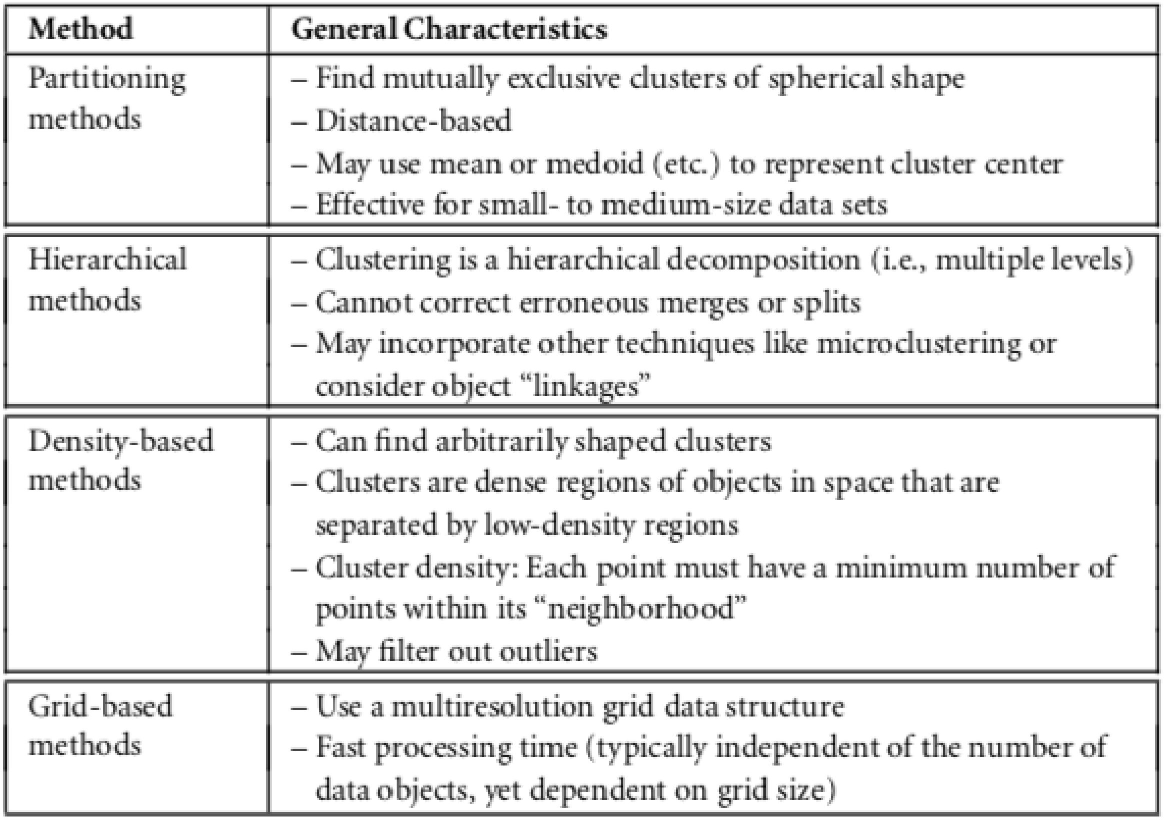
19. Why the types of variables are so important in a clustering algorithm? (5 marks)
Answer:

There are several types of clustering variables and these can be classified into general (independent of products, services or circumstances) and specific (related to both the customer and the product, service and/or particular circumstance), on the one hand, and observable (i.e., measured directly) and unobservable (i.e., inferred) on the other. The types of variables used for cluster analysis provide different segments and, thereby, influence segment-targeting strategies. However, faulty assumptions may lead to improper market segments and, consequently, to deficient marketing strategies. Thus, great care should be taken when selecting the clustering variables.

20. What are the main types of variables when dealing with clustering algorithms? (5 marks)
Answer:

Refer to questions 11,12,13,14,15 for this answer ☺

Extra:



Method	General Characteristics
Partitioning methods	– Find mutually exclusive clusters of spherical shape – Distance-based – May use mean or medoid (etc.) to represent cluster center – Effective for small- to medium-size data sets
Hierarchical methods	– Clustering is a hierarchical decomposition (i.e., multiple levels) – Cannot correct erroneous merges or splits – May incorporate other techniques like microclustering or consider object "linkages"
Density-based methods	– Can find arbitrarily shaped clusters – Clusters are dense regions of objects in space that are separated by low-density regions – Cluster density: Each point must have a minimum number of points within its "neighborhood" – May filter out outliers
Grid-based methods	– Use a multiresolution grid data structure – Fast processing time (typically independent of the number of data objects, yet dependent on grid size)