Clustering Interview questions

We believe that you have learned both theoritical and practical knowledge on Naive Bayes classification algorithm through your assignment.

So let's test your knowledge here. This will help you to be prepared for interviews too!

Best with Quest

1. What is K means Clustering Algorithm?

K Means algorithm is a centroid-based clustering (unsupervised) technique. This technique groups the dataset into k different clusters having an almost equal number of points. Each of the clusters has a centroid point which represents the mean of the data points lying in that cluster.

The idea of the K-Means algorithm is to find k-centroid points and every point in the dataset will belong to either of the k-sets having minimum Euclidean distance.

2. Is Feature Scaling required for the K means Algorithm?

Yes, K-Means typically needs to have some form of normalization done on the datasets to work properly since it is sensitive to both the mean and variance of the datasets.

For performing feature scaling, generally, StandardScaler is recommended, but depending on the specific use cases, other techniques might be more suitable as well.

For Example, let’s have 2 variables, named age and salary where age is in the range of 20 to 60 and salary is in the range of 100-150K, since scales of these variables are different so when these variables are substituted in the euclidean distance formula, then the variable which is on the large scale suppresses the variable which is on the smaller scale. So, the impact of age will not be captured very clearly. Hence, you have to scale the variables to the same range using Standard Scaler, Min-Max Scaler, etc.

3. Why do you prefer Euclidean distance over Manhattan distance in the K means Algorithm?

Euclidean distance is preferred over Manhattan distance since Manhattan distance calculates distance only vertically or horizontally due to which it has dimension restrictions.

On the contrary, Euclidean distance can be used in any space to calculate the distances between the data points. Since in K means algorithm the data points can be present in any dimension, so Euclidean distance is a more suitable option.

5. Why is the plot of the within-cluster sum of squares error (inertia) vs K in K means clustering algorithm elbow-shaped? Discuss if there exists any other possibility for the same with proper explanation.

Let’s understand this with an example,

Say, we have 10 different data points present, now consider the different cases:

k=10: For the max value of k, all points behave as one cluster. So, within the cluster sum of squares is zero since only one data point is present in each of the clusters. So, at the max value of k, this should tend to zero.
K=1: For the minimum value of k i.e, k=1, all these data points are present in the one cluster, and due to more points in the same cluster gives more variance i.e, more within-cluster sum of squares.
Between K=1 from K=10: When you increase the value of k from 1 to 10, more points will go to other clusters, and hence the total within the cluster sum of squares (inertia) will come down. So, mostly this forms an elbow curve instead of other complex curves.
Hence, we can conclude that there does not exist any other possibility for the plot.

6. What are the advantages and disadvantages of the K means Algorithm?

Advantages:
-Easy to understand and implement.
-Computationally efficient for both training and prediction.
-Guaranteed convergence.

Disadvantages:
-We need to provide the number of clusters as an input variable to the algorithm.
-It is very sensitive to the initialization process.
-Good at clustering when we are dealing with spherical cluster shapes, but it will perform poorly when dealing with more complicated shapes.
-Due to the leveraging of the euclidean distance function, it is sensitive to outliers.

7. How to decide the optimal number of K in the K means Algorithm?

Most of the people give answers to this question directly as the Elbow Method however the explanation is only partially correct.

In order to find the optimal value of k, we need to observe our business problem carefully, along with analyzing the business inputs as well as the person who works on that data so that a decent idea regarding the optimal number of clusters can be extracted.

For Example, If we consider the data of a shopkeeper selling a product in which he will observe that some people buy things in summer, some in winter while some in between these two. So, the shopkeeper divides the customers into three categories. Therefore, K=3.

In cases where we do not get inference from the data directly we often use the following mentioned techniques:

Elbow Method – This method finds the point of inflection on a graph of the percentage of variance explained to the number of K and finds the elbow point.
Silhouette method – The silhouette method calculates similarity/dissimilarity score between their assigned cluster and the next best (i.e, nearest) cluster for each of the data points.
Moreover, there are also other techniques along with the above-mentioned ones to find the optimal no of k.

8. How to perform K means on larger datasets to make it faster?

The idea behind this is mini-batch k means, which is an alternative to the traditional k means clustering algorithm that provides better performance for training on larger datasets.

It leverages the mini-batches of data, taken at random to update the cluster mean with a decreasing learning rate. For each data batch, the points are all first assigned to a cluster and then means are re-calculated. The cluster centres are then further re-calculated using gradient descent. This algorithm provides faster convergence than the typical k-means, but with a slightly different cluster output.

9. What are the possible stopping conditions in the K means Algorithm?

The following can be used as possible stopping conditions in K-Means clustering:

Max number of iterations has been reached: This condition limits the runtime of the clustering algorithm, but in some cases, the quality of the clustering will be poor because of an insufficient number of iterations.
When RSS(within-cluster sum of squares) falls below a threshold: This criterion ensures that the clustering is of the desired quality after termination. Practically in real-life problems, it’s a good practice to combine it with a bound on the number of iterations to guarantee convergence.
Convergence: Points stay in the same cluster i.e., the algorithm has converged at the minima.
Stability: Centroids of new clusters do not change.

10. What is the effect of the number of variables on the K means Algorithm?

The number of variables going into K means the algorithm has an impact on both the time(during training) and complexity(upon application) along with the behaviour of the algorithm as well.

This is also related to the “Curse of dimensionality”. As the dimensionality of the dataset increases, more and more examples become nearest neighbours of xt, until the choice of nearest neighbour is effectively random.

A key component of K means is that the distance-based computations are directly impacted by a large number of dimensions since the distances between a data point and its nearest and farthest neighbours can become equidistant in high dimension thereby resulting in reduced accuracy of distance-based analysis tools.

Therefore, we have to use the Dimensionality reduction techniques such as Principal component analysis (PCA), or Feature Selection Techniques.

11. What are the different methods of hierarchical clustering? What are the pros and cons of the hierarchical clustering?

-Agglomerative – Also called bottom-up approach. Each observation starts in its own cluster and pairs of clusters are merged as one moves up the hierarchy.

-Divisive – Also called top-down approach. All observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

Pros:
-Shows all possible linkages
-Give much better understanding of data
-No need to pre-define the number of clusters

Cons:
-Scalability
-Computationally expensive

12. What is Density-based Clustering? What is the DBSCAN Algorithm?

Density-Based Clustering is an unsupervised machine learning method that identifies different groups or clusters in the data space. These clustering techniques are based on the concept that a cluster in the data space is a contiguous region of high point density, separated from other such clusters by contiguous regions of low point density.

Partition-based(K-means) and Hierarchical clustering techniques are highly efficient with normal-shaped clusters while density-based techniques are efficient in arbitrary-shaped clusters or detecting outliers.

DBSCAN also known as Density-Based Spatial Clustering Application with Noise is an unsupervised machine learning algorithm that forms the clusters based upon the density of the data points or how close the data is.

As a result, the points which are outside the dense regions are excluded and considered as the noisy points or outliers. This characteristic of the DBSCAN algorithm makes it a perfect fit for outlier detection and making clusters of any random shapes and sizes.

This algorithm works on a parametric approach that uses two parameters i.e., eps(epsilon) and min_pts.

eps: It represents the radius of the neighbourhoods around a data point x.
min_pts: It is the minimum number of data points that we want in the neighbourhood of a particular point to define a cluster.

13. How does the epsilon value affect the DBSCAN Clustering Algorithm? Is the DBSCAN Algorithm sensitive to the values of the parameters?

The DBSCAN Algorithm is sensitive to the choice of epsilon. When we have clusters with varying densities, then two cases arise i.e.,

If epsilon is too small: In such cases, we define the sparser clusters as noise i.e, result in the elimination of sparse clusters as outliers.

If epsilon is too large: In such cases, the denser clusters may be merged together, which gives the incorrect clusters.

Yes, the DBSCAN algorithm is very sensitive to the values of epsilon and min_pts. Therefore, it is crucial to understand how to choose the values of epsilon and min_pts. A minor variation in these values can change the results significantly produced by the DBSCAN algorithm.

14. BIRCH Clustering Algorithm?

Balanced Iterative Reducing and Clustering using Hierarchies, or BIRCH for short, deals with large datasets by first generating a more compact summary that retains as much distribution information as possible, and then clustering the data summary instead of the original dataset. BIRCH actually complements other clustering algorithms by virtue if the fact that different clustering algorithms can be applied to the summary produced by BIRCH. BIRCH can only deal with metric attributes (similar to the kind of features KMEANS can handle).

15. Explain the Agglomerative Hierarchical Clustering algorithm with the help of an example.

Initially, each data point is considered as an individual cluster in this technique. After each iteration, the similar clusters merge with other clusters and the merging will stop until one cluster or K clusters are formed.

The steps of the agglomerative algorithm are as follows:

Compute the proximity matrix.
Let each data point be a cluster.
Repeat this step: Combine the two closest clusters and accordingly update the proximity matrix.
Until only a single cluster remains.
For Example,

Let’s say we have six observations named {A,B,C,D,E,F}.

Step- 1: In the first step, we compute the proximity of individual observations and consider all the six observations as individual clusters.

Step- 2: In this step, similar clusters are merged together and result in a single cluster.

For our example, we consider B, C, and D, E are similar clusters that are merged in this step. Now, we are remaining with four clusters named A, BC, DE, F.

Step- 3: We again compute the proximity of new clusters and merge the similar clusters to form new clusters A, BC, DEF.

Step- 4: Again, compute the proximity of the newly formed clusters. Now, the clusters named DEF and BC are similar and combine together to form a new cluster. Therefore, now we are left with two clusters named A, BCDEF.

Step- 5: Finally, all the clusters are combined together and form a single cluster and our procedure is completed for the given algorithm.

Therefore, the pictorial representation of the above example is shown below:

16. What is a dendrogram in Hierarchical Clustering Algorithm? How can you find the clusters to have upon looking at the dendrogram? What is Space and Time Complexity of the Hierarchical Clustering Algorithm?

A dendrogram is defined as a tree-like structure that is mainly used to store each step as a memory that the Hierarchical clustering algorithm performs. In the dendrogram plot, the Y-axis represents the Euclidean distances between the observations, and the X-axis represents all the observations present in the given dataset.

In a dendrogram, look for the largest vertical line which doesn’t cross any horizontal line.

With the help of this line, we can draw a horizontal line and then, the points where this horizontal line cross over the various vertical lines, we count all those intersecting points, and then count of intersecting points is the ideal answer for the number of clusters the dataset can have.

Space complexity: Hierarchical Clustering Technique requires very high space when the number of observations in our dataset is more since we need to store the similarity matrix in the RAM. So, the space complexity is the order of the square of n.

Space complexity = $O(n^2)$ where n is the number of observations.

Time complexity: Since we have to perform n iterations and in each iteration, we need to update the proximity matrix and also restore that matrix, therefore the time complexity is also very high. So, the time complexity is the order of the cube of n.

Time complexity = $O(n^3)$ where n is the number of observations.

17. What is spectral clustering?

Spectral clustering is an EDA technique that reduces complex multidimensional datasets into clusters of similar data in rarer dimensions. The main outline is to cluster the all spectrum of unorganized data points into multiple groups based upon their uniqueness "Spectral clustering is one of the most popular forms of multivariate statistical analysis" 'Spectral Clustering uses the connectivity approach to clustering', wherein communities of nodes (i.e. data points) that are connected or immediately next to each other are identified in a graph. The nodes are then mapped to a low-dimensional space that can be easily segregated to form clusters. Spectral Clustering uses information from the eigenvalues (spectrum) of special matrices (i.e. Affinity Matrix, Degree Matrix and Laplacian Matrix) derived from the graph or the data set.

Spectral clustering methods are attractive, easy to implement, reasonably fast especially for sparse data sets up to several thousand. Spectral clustering treats the data clustering as a graph partitioning problem without making any assumption on the form of the data clusters.

18. Difference between Spectral Clustering and Conventional Clustering Techniques

Spectral clustering is flexible and allows us to cluster non-graphical data as well. It makes no assumptions about the form of the clusters. Clustering techniques, like K-Means, assume that the points assigned to a cluster are spherical about the cluster centre. This is a strong assumption and may not always be relevant. In such cases, Spectral Clustering helps create more accurate clusters. It can correctly cluster observations that actually belong to the same cluster, but are farther off than observations in other clusters, due to dimension reduction.

The data points in Spectral Clustering should be connected, but may not necessarily have convex boundaries, as opposed to the conventional clustering techniques, where clustering is based on the compactness of data points. Although, it is computationally expensive for large datasets, since eigenvalues and eigenvectors need to be computed and clustering is performed on these vectors. Also,for large datasets, the complexity increases and accuracy decreases significantly.

Now Rest with this Quest :)