Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 66%. We keep your highest score.

Next item →

1. Why do we need a stopping criterion when we are using the HAC?   **1 / 1 point**

   ○ The algorithm will turn our data into small clusters.

   ⦿ The algorithm will turn our data into just one cluster.

   > Correct! As the algorithm continue checking the distance between all the pairs of closest points, we can turn our points into just one cluster, that is why we need to come up with a stopping criterion when we are using HAC. You can find more information in the video *Hierarchical Agglomerative Clustering*.

   ○ The algorithm will not start working if we don't assign a number of clusters.

   ○ The stopping criterion ensures centroids are calculated correctly.

2. According to the DBSCAN required inputs, which statement describes the n_clu input?   **1 / 1 point**

   ○ It's the function to calculate distance.

   ○ It's the radius of local neighborhood.

   ⦿ It determines density threshold (for fixed Ɛ) (The minimum amount of points for a particular point to be consider a core point of a cluster).

   > Correct! The following statement describes the n_clu. You can find more information in the video *DBSCAN*.

   ○ It's the maximum amount of observations for a particular point to be consider a core point of a cluster.

3. How is a core point defined in the DBSCAN algorithm?   **1 / 1 point**

   ○ A point that has the same amount of n_clu neighbors within and outside the Ɛ-neighborhood.

   ○ A point that has no points in its Ɛ-neighborhood.

   ○ An Ɛ-neighbor point than has fewer than n_clu neighbors itself.

   ⦿ A point that has more than n_clu neighbors in their Ɛ-neighborhood.

   > Correct! Core points are those which have more than n_clu neighbors in their local neighborhood, including itself ("Ɛ-neighborhood"). For example: n_clu = 3 (means that, that point has a least 2 other neighbors that are withing the epsilon distance) You can find more information in the video *DBSCAN*.

👍 Like   👎 Dislike   🚩 Report an issue