**10.4 For the *k*-means algorithm, it is interesting to note that by choosing the initial cluster**

**centers carefully, we may be able to not only speed up the algorithm’s convergence, but**

**also guarantee the quality of the final clustering. The *k*-meansCC algorithm is a variant**

**of *k*-means, which chooses the initial centers as follows. First, it selects one center**

**uniformly at random from the objects in the data set. Iteratively, for each object *p* other**

**than the chosen center, it chooses an object as the new center. This object is chosen at**

**random with probability proportional to *dist*.*p*/2, where *dist*.*p*/ is the distance from *p***

**to the closest center that has already been chosen. The iteration continues until *k* centers**

**are selected.**

**Explain why this method will not only speed up the convergence of the *k*-means**

**algorithm, but also guarantee the quality of the final clustering results.**

**10.12 Present conditions under which density-based clustering is more suitable than partitioning-based clustering and hierarchical clustering. Give application examples to support your argument.**

Density-based clustering can identify outliers and abnormal (not spherical) cluster shapes.

Density-based clustering also identifies mutually exclusive clusters, as opposed to hierarchical clustering which identifies nested clusters. If your data necessitates outlier identification, abnormal cluster identification, and exclusive clusters, density-based clustering should be used over hierarchical or partition methods. If you are also unsure of the number of clusters in your data, density-based clustering may be a good choice. Unlike partition and hierarchical methods, density-based methods do not require the number of clusters as a parameter. Finally, density-based methods identify clusters based on density rather than similarity/distance. Density-based methods should be used when your goal is to identify where many points are clustered (high density regions) not just identify groupings of points that are similar to each other.

Density-based clustering methods are better suited than partitioning or hierarchical clustering methods when working with spatial data, data that describes the location of objects in physical space. Considering the current pandemic, density-based clustering may be useful for identifying the highly infected areas. There’s no reason to believe the virus would spread in a spherical shape, so we want to be able to identify abnormal shapes. Also, identifying outliers would be useful for knowing where potential new dense regions of infection may start. Clustering may also be useful for identifying areas of a city struggling economically, by identifying areas with a high density of foreclosures or empty commercial buildings, for example. This could help direct city resources to economically stressed areas.

**10.16 Describe each of the following clustering algorithms in terms of the following criteria:**

**(1) shapes of clusters that can be determined; (2) input parameters that must be**

**specified; and (3) limitations.**

(a) *k*-means: (1) can determine spherical clusters (2) Input parameters include k (number of clusters) (3) Limitations: k-means can only identify spherical clusters (cannot identify abnormal shaped clusters) and includes all points in a cluster (cannot identify outliers/includes outliers in clusters to which they may not truly belong). K-means also cannot identify clusters of different sizes. The clusters identified are sensitive to the initial points chosen as centroids. It is advised that the k-means algorithm be run multiple times with different starting points to check the robustness of findings.

(b) *k*-medoids: (1) can determine spherical clusters (2) Input parameter: *k* (number of clusters) (3) Limitations: initial medoid choice is arbitrary and resulting clusters are sensitive to initial medoid choice. Does not scale well to larger data sets. Cannot identify arbitrary cluster shapes.

(c) CLARA: (1) can determine spherical clusters (2) Input parameter: *k* (number of clusters) (3) Limitations: if the best *k*-medoids are not part of the selected sample, it is impossible that they will be selected using CLARA.

(d) BIRCH: (1) spherical (2) Input parameters: *k* (number of clusters), branching factor (max number of children per nonleaf node, threshold (max diameter of subclusters stored at the leaf nodes of the tree), and length of a node (3) Limitations: Because each node in a cluster feature tree is limited in size, the CF-tree node may not always correspond to what a research may think of as a natural cluster. BIRCH uses radius and diameter to control the boundary of a cluster, so it will have difficulty with non-spherical clusters. BIRCH only works with numerical data. CF-tree is affected by the order in which data objects are inserted.

(e) CHAMELEON: (1) can identify arbitrary cluster shapes (2) Input parameters: *k*, ,minsize, alpha (3) Limitations: high computational demand for high dimensional space. Initial sparsification may not produce subclusters, phase 2 agglomerative clustering with produce more errors.

(f) DBSCAN: (1) abnormal clusters may be determined (2) input parameters include epsilon (radius from core point) and minPts (minimum points within epsilon neighborhood to be considered a core point) (3) Limitations: the input parameters are arbitrary (determined by the user) and differences in parameter choice will greatly effect the clusters identified. DBSCAN does not work well with sparse data or data of varying densities. This could become problematic with high dimensional data (may become too sparse).