* Q10.1 – Briefly describe and give examples of each of the following approaches to clustering: *partitioning* methods, *hierarchical* methods, *density-based* methods, and *grid-based* methods.

Partitioning methods of clustering is the most popular method and involves the process of grouping data points into exclusive groups based on the partitioning algorithm. This algorithm has an objective function of finding the maximum similarity between points in a cluster and minimum similarity between other clusters. It does with an iterative approach that redistributes data points in successive rounds of analysis based on cluster similarity measures. Examples of this method include k-means, k-modes, and k-medoids.

Hierarchical methods of clustering look to create a tree structure of clusters to represent the data set. This is useful for visualizations, because clusters can be viewed at different levels of granularity within the data and re-arrange the clusters based on the current level being viewed. They work from an agglomerative (bottom-up) approach that starts with a single data point in their own cluster and iteratively grouping them together. The other approach, divisive method (top-down) approach, starts with all points into a single cluster and iteratively splitting the cluster into more until a termination criterion is met. Examples of these methods include algorithmic (BIRCH and chameleon), probabilistic, and Bayesian hierarchical methods.

Density-based methods of clustering are better suited to find abnormally shaped clusters, such as an “S” or circle. The algorithm looks for dense regions of objects in a neighborhood that are also separated by regions of sparsity. Examples of this method include DBSCAN, OPTICS, and DENCLUE

Grid-based methods of clustering split the data space into a number of cells containing a number of data objects and then uses different methods to summarize or optimize the model/clusters. One example of this is STING, which will look at each cell and compute the statistical parameters of each cell and group them accordingly. CLIQUE is another method that combines a density-based approach with a grid-based approach, along with the inclusion of the Apriori method. The cells are analyzed to see if they meet a designated threshold, if not they would be trimmed from the model.

* Q10.2 – Suppose that the data mining task is to cluster points (with (x,y) representing location) into three clusters, where the points are:

A1 (2,10), A2 (2,5), A3 (8,4), B1 (5,8), B2 (7,5), B3 (6,4), C1 (1,2), C2 (4,9).

The distance function is Euclidean distance. Suppose initially we assign A1, B1, and C1 as the center of each cluster, respectively. Use the *k-means* algorithm to show *only*

1. The three cluster centers after the first round of execution.

After using the k-means algorithm, the resulting three clusters and centers are:

Cluster 1: {A3, B1, B2, B3, C2}, Center of Cluster: (6,6)

Cluster 2: {A2, C1}, Center of Cluster (1.5, 3.5)

Cluster 3: {A1}, Center of Cluster: (2, 10)

1. The final three clusters.

After using the k-means algorithm, the final three cluster are:

Cluster 1: {A3, B2, B3}

Cluster 2: {A1, B1, C2}

Cluster 3: {A2, C1}

* Q10.3 – Use an example to show why the k-means algorithm may not find the global optimum, that is, optimizing the with-cluster variation.

The k-means algorithm may not find the global optimum, because it depends on the random selection of center point at the beginning of the process. Depending on the initial random selection, as well as the desired number of clusters, the user may end up with a different outcome than the next time the clustering algorithm is done. The global optimum could be determined if all possible outcomes are calculated; however, that would take an exponential amount of time based on the number of points. It is more practical to run the k-means clustering process several times with different starting cluster centers.

Scenarios similar to Example 10.2 in the textbook show this drawback of the k-means and within-cluster variation. There is an outlier in the data and two clusters are defined as the desired output from the user. As a result, the resulting clusters are grouped based on the optimizing the minimum within-cluster variation. This leads to re-assigning data points into clusters with a mean value relatively far from all points within it. If the outlier is removed, or the desired number of clusters is increased from two, then there would be drastically different assigned clusters. Therefore, a different local optimum would be optimized, rather than the global optimum, unless every possible outcome is considered.

* Q10.6 – Both *k-means* and *k-medoids* algorithms can perform effective clustering.

1. Illustrate the strength and weakness of *k-means* in comparison with *k-medoids*.

The main strength of k-means over k-medoids is that is less computationally intense to find the mean than the median of a cluster. The main weakness of k-means is that it can be significantly impacted by the presence of noise and outliers. K-medoids is not impacted by noise and outliers to the extent that k-means is.

1. Illustrate the strength and weakness of these schemes in comparison with a hierarchical clustering scheme (e.g., AGNES).

The main strengths of k-means and k-medoids over hierarchical methods, such as AGNES, are their effectiveness in findings spherical-shaped clusters and the ability to undo previous merges/splits. Hierarchical methods of clustering are unable to go backwards once a splitting or merge point is decided, which also makes it difficult to scale hierarchical methods up as the data set increases. The main weakness of k-means and k-medoids is that a defined number of clusters must be chosen, this is not a problem with hierarchical methods. Additionally, hierarchical methods can make up for their weaknesses by merging with other clustering methods to improve quality and performance.

