K-Means, Kernal K-Means K-Medians and K-Medoids – Approach

Cluster Analysis

- A cluster is a collection of data objects which are
  - Similar (or related) to one another within the same group (i.e., cluster)
  - Dissimilar (or unrelated) to the objects in other groups (i.e., clusters)
- Cluster analysis (or clustering, data segmentation, …)
  - Given a set of data points, partition them into a set of groups (i.e., clusters) which are as similar as possible
- Cluster analysis is unsupervised learning (i.e., no predefined classes)
  - This contrasts with classification (i.e., supervised learning)
- Typical ways to use/apply cluster analysis
  - As a stand-alone tool to get insight into data distribution, or
  - As a preprocessing (or intermediate) step for other algorithms

We have various partitioning methods to help identify clusters.

Partitioning method: Discovering the groupings in the data by optimizing a specific objective function and iteratively improving the quality of partitions

K-partitioning method: Partitioning a dataset D of n objects into a set of K clusters so that an objective function is optimized (e.g., the sum of squared distances is minimized, where ck is the centroid or medoid of cluster Ck)

A typical objective function: Sum of Squared Errors (SSE)

Problem definition: Given K, find a partition of K clusters that optimizes the chosen partitioning criterion. Global optimal: Needs to exhaustively enumerate all partitions

There are various heuristic methods (i.e., greedy algorithms): K-Means, K-Medians, K-Medoids, etc

Lets first look at K-Means Clustering Method

K-Means Algorithm

- Each cluster is represented by the center of the cluster
- Given K, the number of clusters, the K-Means clustering algorithm is outlined as follows
- Select K points as initial centroids
- Repeat
- Form K clusters by assigning each point to its closest centroid
- Re-compute the centroids (i.e., mean point) of each cluster
- Until convergence criterion is satisfied
- Different kinds of measures can be used
- Manhattan distance (L1 norm), Euclidean distance (L2 norm), Cosine similarity

- Efficiency: O(tKn) where n: # of objects, K: # of clusters, and t: # of iterations
    - Normally, K, t << n; thus, an efficient method
- K-means clustering often terminates at a local optimal
    - Initialization can be important to find high-quality clusters
- Need to specify K, the number of clusters, in advance
    - There are ways to automatically determine the “best” K
    - In practice, one often runs a range of values and selected the “best” K value
- Sensitive to noisy data and outliers
    - Variations: Using K-medians, K-medoids, etc to avoid noisy data and outliers
- K-means is applicable only to objects in a continuous n-dimensional space
    - Using the K-modes for categorical data
- Not suitable to discover clusters with non-convex shapes
    - Using density-based clustering, kernel K-means, etc.

Variations of K-Means

- There are many variants of the K-Means method, varying in different aspects
    - Choosing better initial centroid estimates
        - K-means++, Intelligent K-Means, Genetic K-Means
- Choosing different representative prototypes for the clusters
    - K-Medoids, K-Medians, K-Modes
- Applying feature transformation techniques
    - Weighted K-Means, Kernel K-Means

Initialization of K-Means

- Different initializations may generate rather different clustering results (some could be far from optimal)
- Select K seeds randomly
    - Need to run the algorithm multiple times using different seeds
- There are many methods proposed for better initialization of k seeds
    - K-Means++ (Arthur & Vassilvitskii’07):
        - The first centroid is selected at random
        - The next centroid selected is the one that is farthest from the currently selected (selection is based on a weighted probability score)
        - The selection continues until K centroids are obtained

K-Medoids – Handling Outliers from K-Means

- The K-Means algorithm is sensitive to outliers! —since an object with an extremely large value may substantially distort the distribution of the data
- K-Medoids: Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster
- The K-Medoids clustering algorithm:
    - Select K points as the initial representative objects (i.e., as initial K medoids)
    - Repeat
        - Assigning each point to the cluster with the closest medoid

        - Randomly select a non-representative object oi
        - Compute the total cost S of swapping the medoid m with oi
        - If S < 0, then swap m with oi to form the new set of medoids
    - Until convergence criterion is satisfied

K-Medians – Another approach to handling outliers from K-Means

- Medians are less sensitive to outliers than means
    - Think of the median salary vs. mean salary of a large firm when adding a few top executives!
- K-Medians: Instead of taking the mean value of the object in a cluster as a reference point, medians are used (L1-norm as the distance measure)
- The criterion function for the K-Medians algorithm:
- The K-Medians clustering algorithm:
    - Select K points as the initial representative objects (i.e., as initial K medians)
    - Repeat
        - Assign every point to its nearest median
        - Re-compute the median using the median of each individual feature
    - Until convergence criterion is satisfied

K-Modes – Clustering Categorical Data

- K-Means cannot handle non-numerical (categorical) data
    - Mapping categorical value to 1/0 cannot generate quality clusters for high-dimensional data
- K-Modes: An extension to K-Means by replacing means of clusters with modes
- Dissimilarity measure between object X and the center of a cluster Z
    - $\Phi(xj, zj) = 1 - njr/nl$ when $xj = zj$; 1 when $xj \neq zj$
        - where zj is the categorical value of attribute j in Zl, nl is the number of objects in cluster l, and njr is the number of objects whose attribute value is r
- This dissimilarity measure (distance function) is frequency-based
- Algorithm is still based on iterative object cluster assignment and centroid update
- A fuzzy K-Modes method is proposed to calculate a fuzzy cluster membership value for each object to each cluster
- A mixture of categorical and numerical data: Using a K-Prototype method

Kernal K-Means Clustering

- Kernel K-Means can be used to detect non-convex clusters
    - K-Means can only detect clusters that are linearly separable
- Idea: Project data onto the high-dimensional kernel space, and then perform K-Means clustering
    - Map data points in the input space onto a high-dimensional feature space using the kernel function
    - Perform K-Means on the mapped feature space
- Computational complexity is higher than K-Means
    - Need to compute and store n x n kernel matrix generated from the kernel function on the original data

- The widely studied spectral clustering can be considered as a variant of Kernel K-Means clustering

References: Course CS412 Introduction to Data Mining