Clustering: similarity is hard to judge in high-dimension space. < hierarchical / point assignment

一. Hierarchical Clustering. { represent cluster = euclidean case / non-euclidean case; determine "nearness"; when to stop combining }

1. euclidean case: represent cluster by centroid, measure by distance of centroid.

2. non-euclidean case: clustroid (data point closest to other points), measure by distance of clustroid. → smallest max-dist / avg-dist / sum-square-dist

cohesion (merge clusters whose union is most cohesive) { diameter of mergered cluster → max-dist in cluster; avg-dist between points in cluster; density-based approach: diameter / avg, then / # of point }

3. complexity: n data point, at most (n-1) step of merge, naive: $O(n^3)$ to form dist-matrix $\Rightarrow O(n^3)$

→ using priority queue: $O(n^2 \log n)$. → still to expensive.

4. measure: (1) sum square error (SSE): $\sum (dist^2(point, it's\ centroid))$ (when to stop)

二. K-means Algs. 1. Step: (1) pick k points randomly as k clusters, then place each point to cluster whose current centroid is nearest → update location of centroid after all points assigned → reassign → repeat

2. how to select k: avg-dist_to_centroid ↑ → k

三. BFR Alg. 1. assumption: ①. clusters are normally distributed around a centroid in Euclidean. → memory required: O(k).

2. step: read disk → clustering to k cluster → points in cluster summarized as N and σ. (DS: discard set). → second read → update N and σ and drop points.

3. types of points: (1) DS: (discard set): close enough to a centroid, to be summarized and dropped

(2). CS (compression set): points close together but not close to any existing centroid. to be summarized but not assigned to cluster

(3) RS (retained set): isolated points waiting to be assigned to CS → outliers.

4. For each cluster, DS is summarized by (1). N: # of point, (2) vector SUM: SUM[i] is sum of coordinates of points in ith dimension, (3). vector SUMSQ, SUMSQ[i] is sum of square of coordinate in ith dimension. $Var(X) = E(X^2) - E^2(X)$

5. (2d+1) values represent any size cluster in d dimension., centroid in i dimension = $\frac{SUM_i}{N}$.

var of a cluster's DS in ith dimension = $\frac{SUMSQ_i}{N} - \left(\frac{SUM_i}{N}\right)^2$

6. Mahalanobis distance: normalized Euclidean dist from centroid

$y_i = \frac{x_i - c_i}{\sigma_i}$ $d(x, c) = \sqrt{\sum (y_i)^2} = \sqrt{\sum_{i=1}^{d} \left(\frac{x_i - c_i}{\sigma_i}\right)^2}$ ⇒ if clusters are normally distributed in d

dimension, then after transformation, one standard deviation = $\sqrt{d}$

7. when do two compressed set combined: calculate var of combined clusters if less than some threshold

III. CURE Alg. - for arbitrary shape - use collection of representative points to represent cluster

1. problem of BFR / k-means: assumption & axes are fixed.

2. step: (1) Pass 1: pick random points → cluster these points hierarchically → for each cluster, pick a sample of points as representative points (as dispersed as possible) → then remove xx% toward the centroid. (2) pass 2: rescan dataset and place data point to "closest cluster" closeset to one of representative points

3. BFR vs CURE.