2022.02.09 Lecture 5

k-mean 为 partitional clustering

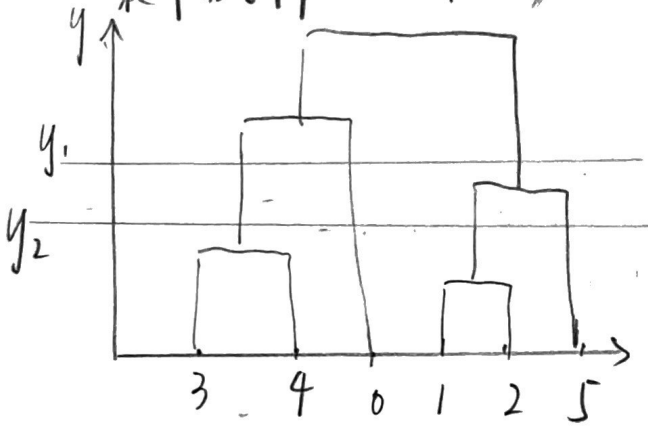
Hierarchical Clustering

2种 types: Agglomerative, Divisive

Agglomerative

最开始每个 point 都为1个 cluster, 逐渐将 distance 近的 cluster 合并

称: dendrogram



y轴表示每个 cluster 都在什么距离下合并的.

例 y1 处切断, 可知 3,4 为1个 cluster, 6为1个 cluster, 1,2,5 为1个 cluster

y2 处则 (3,4) (6) (1,2) (5)

每次将2个最近的 cluster 合并, 直到只有1个 cluster

如何算 clusters 间距离?

注: 小写 $d(p_1, p_2)$ 表点间距, 大写 $D(C_1, C_2)$ 表 cluster 间距.

- Single-Link Distance

$D(C_1, C_2)$ = C_1 中和 C_2 中最近的点的距离

$$= \min\{ d(p_1, p_2) \mid p_1 \in C_1 \wedge p_2 \in C_2 \}$$

- Complete-Link Distance

$$D(C_1, C_2) = \max\{ d(p_1, p_2) \mid p_1 \in C_1 \wedge p_2 \in C_2 \}$$

properties: ① 可能会 split up large-cluster

② 一般分出的 cluster 的直径差不多

Average - Link

$$D(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{p_1 \in C_1, p_2 \in C_2} d(p_1, p_2)$$

所有pairwise distance的average.

Centroid

Ward's Distance

$$D(C_1, C_2) = \sum_{p \in C_{12}} d(p, m_{12}) - \sum_{p_1 \in C_1} d(p_1, m_1) - \sum_{p_2 \in C_2} d(p_2, m_2)$$

算每个点离m的距离之和. 合并之后的distance比合并前distance多多少?

Density-Based Clustering

densely packed在一起的分为1个cluster

定义density:

Given radius (ε) around a point, 若ε之内至少有(min_pts)个data points, 则cluster

需自行定义ε和min_pts

Def: core points: ε radius内有至少min-pts个neighbor的点.

border points: 不是core points, 但在某core points ε之内.

noise points: 不在core points的ε之中.

DBScan Algorithm.

定义 ε 和 min_pts.

1. 找出以每个点为中心，ε为radius的圆
2. 看每个点为圆心的圆内有多少 data points，找出所有 core points 并 label 之.
3. 对于每个 core point，找出在其圆内但不是 core 的点，label 为 border
4. 其余为 noise
5. 相互在对方圆内的2个 core points 分为一个 cluster
6. borders 分配到临近 cluster 中.

Core idea: core points 之间互为 neighbors，则将这些 core points 及它们的 neighbors 分为一组.

noise points 不分入任何组.

Limitations:

1. 生成的 cluster 的 density 一般差不多
2. cluster 的 density 不同时可能 fail
3. high-dimension space 中 "density" 难定义