2022.2.7 Lecture 4

Jaccard Similarity

	w_1	w_2	...	w_d
x	1	0		
y	1	1		

w_i 为 word，x,y 为 documents. 若 w_i 出现在 x 中，w_i 为 1

算 distance？

$L_1(x,y) = \sum_{i=1}^{d} |x_i - y_i|$ 看少词不同

↓

称为 Manhattan distance.

↓

not useful! 例：x_1, y_1 极长，相似，有1个词不同

x_2, y_2 极短，也只有1个词不同，但相似程度不同

Jaccard Similarity

$$Jsim(x,y) = \frac{|x \cap y|}{|x \cup y|} \qquad JDist(x,y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$

Distance 表示 dissimilarity. dis 与 sim 相反. 因此 1 - similarity = distan

clustering

将 data 分组：similar data points 在同组中

不相似的 data points 在不同组中

Clustering 的对

· partitional

每个 object 只属于1个 cluster (i.得 n个objects 分成 k个 clusters)

计算每组中每对points之间的 distance 之和. 使整体最小.

$\sum_k \sum_{x_i, x_j \in C_k} d(x_i, x_j)$ 每个cluster中有对possible pair间distance之和.

↓ 表示 cluster k

↓

太难算 instead, 找k个 points, 使 $\sum_i^k \sum_{x \in C_i} d(x, u_i)$ 最小

Lloyd's Algorithm:

1. randomly 选 k个 centers 做 u_i
2. 将每个data points 归入距 center 最小的那一组.
3. 计算各cluster的 mean, 成为每组cluster新的center.
4. 重复2 & 3 直到每个 data point 不再更改组别

每一次iteration中 $\sum_i^k \sum_{x \in C_i} d(x, u_i)$ 会变小.

properties:

① Always converge!

② 不一定为最优 cluster

↓

改进版: K-means++

选1个 random [data point] 做 1st center, 每次选下1个 center 时, 算当前 center 和其它 data points (potential centers) 距离. 对于每个下一个 potential centers, 其被选中概率 is proportional to 其与当前 center 的距离.