聚类

秦睿哲

一、k-Means

1、Voronoi Diagrams

考虑一个集合 $S=\{s_1,s_2,\ldots,s_k\}\subset R^d$，我们想知道这些点是如何分割空间 R^d 的。Voronoi 图将 R^d 分解为 k 个区域，每个区域对应一个 Voronoi 单元。这样就定义了点 s_i 的区域：

$$R_i=\{x\in R^d \mid \phi_S(x)=s_i\}$$

其中 $\phi_S(x)$ 就是指对于点 $x\in R^d$， S 中到它距离最近的点: $\phi_S(x)=\arg\min_{s_i\in S}\|x-s_i\|$

2、Lloyd's Algorithm

具体来说，k-Means 聚类问题是找到有 k 个聚类的集合 S，最小化

$$\mathrm{cost}(X,S)=\sum_{x\in X}\|\phi_S(x)-x\|^2$$

考虑给定的点集 $P\subseteq R^d$， $|P|=n$，需要划分的聚类个数 $k\in Z^+$。

Lloyd's Algorithm(n,d,k) 算法过程：

① 随机取 k 个聚类中心 $\{c_1,c_2,\ldots,c_k\}\subseteq P$

② 迭代过程：

（1）计算每一个点到聚类中心的距离，选取最小值划分到 k 类（Voronoi Diagrams）

（2）更新聚类中心 $c_i=\frac{1}{|C_i|}\sum_{p\in C_i}p$

重复（1）（2）过程，直到损失函数达到最小保持不变，或者其他的终止条件。

（[1] MATHEMATICAL FOUNDATIONS FOR DATA ANALYSIS）

遗憾的是，无论如何选择收敛条件，该算法都不能保证找到最优的集合，很容易陷入局部最小值，这种情况在 k 很大的情况下比较常见。也就是说，Lloyd's Algorithm 的结果可能是任意差的。

3、k-Means 的 hardness 结论

① 即使在低维空间（k 是一个常数）中，仍是一个 NP-hard 问题；

② 在高维空间（k 不是一个常数），即使 k=2，也是 NP-hard 问题。

总的来说，参数（n，d，k）中，d 和 k 只要有一个不是常数，k-Means 都是 NP-hard 问题。

4、k-Means++（[2] K-Means++: The Advantages of Careful Seeding）

由于 k-means 算法的分类结果会受到初始点的选取而有所区别，因此提出这种算法的改进 k-means++ 。这个算法是对初始点的选择进行改进，其他步骤都一样。初始类中心选取的基本思路就是，初始的聚类中心之间的相互距离要尽可能的远。

算法主要分成两部分：（a）和（b）

（a）如何选取初始类中心 $\{c_1, c_2, \ldots, c_k\}$

（b）Lloyd’s Algorithm

关于（a）的算法：

① 令 $C = \Phi$，随机取 $c_1 \in P, C = \{c_1\}$；

② 对于 $j = 2 \sim k$

1）对每一个 $u \in P$，定义：$D(u, C) = \min\{\|u - c_i\| \mid 1 \le i \le j-1\}$ 和概率

$$f(u) = \frac{D(u, C)^2}{\sum_{u' \in P} D(u', C)^2}$$

2）基于概率 f，从 P 中选取下一个类中心点 c_j

整个过程（a）的时间复杂度为 $\Theta(k \cdot n \cdot d)$，这与 Lloyd’s Algorithm 一次循环的复杂度相等。

k-Means 的近似比的期望为 $\Theta(\log k)$。

考虑一个简单的情况：

假设 $\{A_1, A_2, \ldots, A_k\}$ 为最优的 k 个类，在每一个 A_j，随机选取一个点为 c_j。

定义

$$m(A_j) = \frac{1}{|A_j|} \sum_{u \in A_j} u,$$

$$\mathrm{cost}(A_j, u_0) = \sum_{u \in A_j} \|u - u_0\|^2$$

$$
\begin{aligned}
E\left[\operatorname{cost}\left(A_j,u_0\right)\right] &= \sum_{u_0\in A_j}\frac{1}{\left|A_j\right|}\left(\sum_{u\in A_j}\left\|u-u_0\right\|^2\right) \\
&= \frac{1}{\left|A_j\right|}\sum_{u_0\in A}\sum_{u\in A_j}\left\|u-u_0\right\|^2 \\
&= \frac{1}{\left|A_j\right|}\sum_{u_0\in A}\sum_{u\in A_j}\left\|u-m\left(A_j\right)+m\left(A_j\right)-u_0\right\|^2 \\
&= 2\sum_{u\in A_j}\left\|u-m\left(A_j\right)\right\|^2
\end{aligned}
$$

$\Rightarrow$近似比的期望为 2。

二、k-Median

Center-Based Clustering：核心是找到 k 个类中心点$\{c_1,c_2,\ldots,c_k\}$

（1）k-Means：计算$\sum_{u\in P}\min_{1\le j\le k}\left\|u-c_j\right\|^2$

（2）k-Median：计算$\sum_{u\in P}\min_{1\le j\le k}\left\|u-c_j\right\|$

考虑 k=1 的情况下，两种方法在几何上的差异。

k-Means：$c_1=\frac{1}{|P|}\sum_{u\in P}u$ 几何上为找点的重心。

类似的，k-Median 找的c_1是几何上点的 Fermat 点。

结论：Lloyd Algorithm 和 k-Means++都可以扩展到 k-Median。

假设我们找到的类中心点为$\{c_1,c_2,\ldots,c_k\}$,则在 k-Means++中，$f\left(u_0\right)=\frac{D\left(u_0,C\right)^2}{\sum_{u'\in P}D\left(u,C\right)^2}$

同样的，我们就可以在 k-Median++中计算概率$f\left(u_0\right)=\frac{D\left(u_0,C\right)}{\sum_{u'\in P}D\left(u,C\right)}$

三、k-Center

1、k-Center 算法的目标是最小化 $\max_{u\in P}\left\{\min_{1\le j\le k}\|u-c_i\|\right\}$。我们使用的方法是 Gonzalez 算法（[1] MATHEMATICAL FOUNDATIONS FOR DATA ANALYSIS）。

Gonzalez 算法（$P\subseteq R^d,k$）

（1）初始化 $C=\Phi$，任取 P 中的一个点，令其为 c_1，$C=\{c_1\}$；

（2）$j=2$；循环下列步骤直到 $j=k$：

① 取 $c_j=\arg\max_{u\in P}\left\{\min_{1\le i\le j-1}\|u-c_i\|\right\}$；

② $C=C\cup\{c_j\}$，$j=j+1$.

---

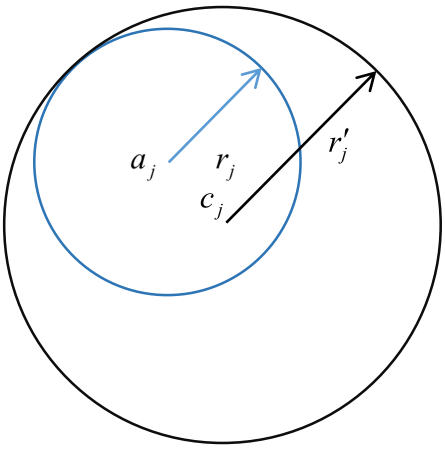
*注：

- Gonzalez(n,d,k)算法的复杂度为 $\Theta(k\cdot n\cdot d)$
- 算法的质量保证：$\dfrac{\text{我们得到的半径}}{\text{最优解的半径}}\le 2$

2、【定理】Gonzalez 算法输出 2 倍近似比的 k-Center 聚类结果。

证明：假设最优的聚类结果为 $A_1,A_2,\ldots,A_k$，对应的类中心为 $a_1,a_2,\ldots,a_k$

情况 1：$\{c_1,c_2,\ldots,c_k\}$恰好落在 $A_1,A_2,\ldots,A_k$ 里，记作 $c_1\in A_1,c_2\in A_2,\ldots,c_k\in A_k$。考虑以 a_j 为球心包住整个 A_j，所需的半径为 r_j；而以 c_j 为球心包住整个 A_j，所需的半径为 r_j'



从图上可以看出，显然有 $r_j'\le 2r_j$，从而近似比 ≤ 2

情况 2：$\{c_1,c_2,\ldots,c_k\}$中存在两个类中心落在同一类中。不失一般性，我们假设 $c_1\in A_1,c_2\in A_2,\ldots,c_i\in A_i$，$c_{i+1}\in A_t$，$1\le t\le i$。

则 $\min_{1\le j\le i}\|c_{i+1}-c_j\|\ge\max_{u\in\bigcup_{j=i+1}^{k}A_j}\left\{\min_{1\le j\le i}\|u-c_i\|\right\}$（由 Gonzalez 算法中心点的选择可知，$c_{i+1}$

是当前最远的点）

不等式左侧$\le \|c_{i+1}-c_t\| \le \|c_{i+1}-a_t\| + \|a_t - c_t\| \le 2r_{\text{opt}}$　其中r_{opt}为最优半径

不等式右侧 $\max\limits_{u \in \bigcup_{j=i+1}^{k} A_j} \left\{ \min\limits_{1 \le j \le i} \|u - c_j\| \right\} \le 2r_{\text{opt}}$，这意味着 $A_{i+1}, A_{i+2}, \ldots, A_k$ 中的点到 $\{c_1, c_2, \ldots, c_i\}$ 的距离 $\le 2r_{\text{opt}}$。同时与情况 1 相同的是，$A_1, A_2, \ldots, A_i$ 中的点到 $\{c_1, c_2, \ldots, c_i\}$ 的距离 $\le 2r_{\text{opt}}$。因此，可以推出 $\{c_1, c_2, \ldots, c_k\}$ 导致的近似比 ≤ 2。

3、关于 k-Center 聚类 Hardness 结论：任何近似比 < 2 的结果都是 NP-hard 的问题，即使 $d = \Theta(1)$。

参考文献：

[1] Jeff M. Phillips. *MATHEMATICAL FOUNDATIONS FOR DATA ANALYSIS.* Section 8 Clustering.

[2] Arthur D, Vassilvitskii S. *K-Means++: The Advantages of Careful Seeding*[C]// Eighteenth Acm-siam Symposium on Discrete Algorithms. ACM, 2007.