Clustering II

Piotr Indyk

Today

- Goal: given points $P = p^1 \ldots p^n$, find *medians* $c^1 \ldots c^k$ and partitioning $C^1 \ldots C^k$ which minimize

$$\sum_{j=1}^{k} \sum_{p \in C^j} D(p, c^j)$$

- Since optimal C^i consists of all points for which c^i is a nearest median, we minimize

$$\sum_{p} \min_{j=1}^{k} D(p, c^j)$$

- Similar to k-means

- Could use A-LINK

- We focus on *provable* algorithms

k-median

- Unlike k-center, no simple algorithm+analysis known
- Will do:
  - Sampling algorithm under *large clusters* assumption
  - Greedy algorithm
- Both are *bicriterion* algorithms: if there are k medians with cost OPT, we produce ak medians with cost $bOPT$, $a, b > 1$

k-median via sampling

First things first: how to quickly solve $k = 1$?

- Sample a point c uniformly at random from P
- The expected cost $\sum_p D(p,c)$ is $\leq 2 \sum_p D(p,c^*)$, where c^* is the optimal median:

$$\begin{aligned} E[\sum_p D(p,c)] &= \sum_p E[D(p,c)] \\ &\leq \sum_p E[D(p,c^*) + D(c^*,c)] \\ &= \sum_p D(p,c^*) + |P| E[D(c^*,c)] \\ &= OPT + OPT \end{aligned}$$

- The cost is $\leq 2t \cdot OPT$ with prob. $\geq 1 - 1/t$ (Markov inequality)

k-median via sampling, ctd.

The general k:

- Assume that in the optimal solution, all C^i's have size $\geq \alpha/k$
- Then, if we choose c^i, $i = 1 \ldots k \ln(2k)/\alpha$, independently and uniformly at random from P, we have a constant probability that $\sum_p \min_{c^i} D(p, c^i)$ is $\leq 2t \cdot OPT$, for $t > 1$

Analysis

- For each C^i, the probability of not choosing any point from it is

$$(1-\alpha/k)^{k\ln(2k)/\alpha} \le e^{-\ln(2k)} = \frac{1}{2k}$$

- The probability that there exists C^i not "touched" by the sample is at most $k/2k = 1/2$

- With probability $1/2$, all C^i's contain at least one sample point (say c^i), randomly chosen from C^i. Conditioned on this event

$$E[\sum_j \sum_{p\in C^i} D(p,c^i)] = \sum_j E[\sum_{p\in C^i} D(p,c^i)] \le 2OPT$$

- Thus, with pr. $1/2\cdot(1-1/t)$ we have cost $\le 2t\cdot OPT$

We get an $[\ln(2k)/\alpha, 2t]$-approximation algorithm (under an assumption)

A $[O(\ln n), O(\ln n)]$-approximation algorithm

- $R = P$
- Repeat until $R = \emptyset$:
  - Choose a median $c \in P$ for which
  $$\sum_{p \in NN(c,R)} D(c,p)$$
  is minimized, where $NN(c,R)$ is the set of $\frac{|R|}{2k}$ nearest neighbors of c in R
  - Remove $NN(c,R)$ from R
- Output all medians found by the process

Intuition

- We carve out "cheap pieces" of clusters, of size $\geq t = \frac{|R|}{2k}$, and output them as clusters (in practice, we would find better medians for them etc.)

- If the t too small, we would have too many clusters

- If the t too large, we could not guarantee the existence of "cheap" pieces. E.g., if all k clusters have equal size and are far from each other, there is no "cheap" piece of size $> |R|/k$

- Setting $t = \frac{|R|}{2k}$ satisfies both constraints

Analysis

Lemma: Let $c^1..c^k$, $C^1 \dots C^k$ be the optimum solution. At any iteration of the algorithm, there exists a set $B \subset C^i \cap R$ of size $\geq \frac{|R|}{2k}$ such that

$$\sum_{p \in B} D(p, c^i) \leq OPT/k$$

Given the fact:

- In each iteration, the size of R decreases by a factor of $(1 - \frac{1}{2k})$
- After $k' = 2k \ln n + 1$ steps, its size is $\leq n(1 - \frac{1}{2k})^{2k \ln n + 1} < 1$, and we are done
- The total cost is at most $k' \cdot OPT/k = O(\ln n) \cdot OPT$

Proof of the lemma

- Define $S^i = C^i \cap R$
- Assume $S^1 \dots S^k$ sorted in the decreasing order of size
- Take the largest i such that $|S^i| \geq \frac{|R|}{2k}$
- Observe that $\sum_{j=1}^{i} |S^j| \geq |R|/2$
- Cheating: assume that each $|S^j|$ is a multiple of $\frac{|R|}{2k}$
- Split each S^j into blocks of size $\frac{|R|}{2k}$
- There are k blocks
- All points in the blocks can be clustered with cost $\leq OPT$
- One of the blocks B has to have cost $\leq OPT/k$

Running time

- Natural implementation: $O(n^2 k \ln n)$
- Sampling-based implementation: roughly $O(nk^2 \ln k)$:
  - In each iteration, sample $O(k)$ candidates for the new median c
  - Choose the best one
  - Similar (more complex) analysis

References

- Sampling-based algorithm for k-median: folklore. A more complex approach (clustering large clusters, identifying points from small clusters, and re-clustering those points) appeared in
  P. Indyk, "Sublinear-time algorithms for metric space problems", STOC'99.

- Greedy algorithm: an adaptation of an algorithm by M. Charikar, O'Callaghan and R. Panigrahy.