Clustering with Same Cluster Queries

Problem → Let X be a subset of some Euclidean space R^d.
Let $C_X = \{C_1 \ldots C_k\}$ be a clustering of X. & x_1 & x_2 belongs to the same cluster is denoted by $x_1 \overset{C_X}{\sim} x_2$
Define $n = |X|$ & $k =$ no. of clusters.
Clustering C_X is center-based if $\exists$ set of centers $\mu = \{\mu_1 \ldots \mu_k\} \subset R^n$ such that the clustering corresponds to the Voronoi diagram over those center points. OR
$\forall x \in X$ & $i \leq k$, $x \in C_i \Leftrightarrow i = \arg\min_j d(x, \mu_j)$.
We also assume that centers μ^* corresponding to C^* are centers of mass of corresponding clusters.
$\mu_i^* = \frac{1}{|C_i|} \sum_{x \in C_i^*} x$. So, oracle's clustering corresponds to the optimal clustering.

γ-Margin Property → Let X be set of points in metric-space M.
Let $C_X = \{C_1 \ldots C_k\}$ be a center-based clustering of X induced by centers $\mu_1, \mu_2 \ldots \mu_k \in M$.
C_X satisfies the γ-margin property iff
$\forall i \in [k]$ & $\forall x \in C_i$ & $\forall y \in X / C_i$,

$$\gamma d(x, \mu_i) < d(y, \mu_i)$$

Query function → For a clustering $C^* = \{C_1^*, C_2^* \ldots C_k^*\}$, a C^*-oracle is a f^n Q_{C^*} that answers queries according to the clustering.

$$Q_{C^*}(x_1, x_2) = \begin{cases} \text{true} & \text{if } x_1 \overset{C^*}{\sim} x_2 \\ \text{false} & \text{else} \end{cases}$$

This is called a semi-supervised active clustering framework (SSAC)

↳ An SSAC instance is determined by the ~~tg~~ tuple (X, d, C^*).

1. An SSAC algorithm A is called a q-solver if for a family G of instances, for every instance (X, d, C^*), it can recover C^* by access to (X, d) & q ~~ques~~ queries to a C^*-oracle
2. It is a polynomial q-solver if its time complexity is polynomial in n & k.
3. G admits an $O(q)$ query complexity if $\exists$ an algo A such that it is a polynomial q-solver $\forall$ instances $\in G$.

Cluster assignment Query → Asks for the cluster index, $Q_{C^*}(x) = i$ iff $x \in C_i^*$. This can be replaced with k same-cluster queries.

Lemma 1 → Let (X, d, C) be a clustering instance that satisfies the γ-margin property. Let μ be the set of centers of C. Let μ_i' such that

$$d(\mu_i, \mu_i') \leq r(C_i)\epsilon, \quad \text{where } r(C_i) = \max_{x \in C_i} d(x, \mu_i)$$

Then if $\gamma \geq 1 + 2\epsilon \Rightarrow$

$$\forall x \in C_i, \forall y \in X / C_i \Rightarrow d(x, \mu_i') < d(y, \mu_i')$$

Proof → Fix any $x \in C_i$ & $y \in C_j$. $d(x, \mu_i') \leq d(x, \mu_i) + d(\mu_i, \mu_i')$
$\leq r(C_i)(1+\epsilon)$

Similarly, $d(y, \mu_i') \geq d(y, \mu_i) - d(\mu_i, \mu_i') \geq (\gamma - \epsilon) r(C_i)$

From both of them, $\Rightarrow d(x, \mu_i') < \frac{1+\epsilon}{\gamma-\epsilon} d(y, \mu_i') < d(y, \mu_i')$

Hence proved.

Lemma 2 → Let Z_p, C_p, μ_p, μ_p' be defined as in the following algo, & $\epsilon = \frac{\gamma - 1}{2}$. If $|Z_p| > \eta$ then the probability that $d(\mu_p, \mu_p') > r(C_p)\epsilon \leq \frac{\delta}{k}$

Theorem → Let (X, d, C) be a clustering instance, where C is center-based & satisfies the γ-margin property. Let $\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$. Assume $\delta \in (0,1)$ & $\gamma > 1$. Then with probability $> 1-\delta$, the following algorithm outputs C.

Algorithm → Inp: Clustering instance X, oracle Q, number of clusters k, parameter $\delta \in (0,1)$

Output → A clustering C of set X,

Initialize → $C = \{\}$, $S_1 = X$; $\eta = \beta \frac{\log k + \log(\frac{1}{\delta})}{(\gamma - 1)^4}$

~~Steps~~ Pseudocode → for $i = 1$ to k do

Phase 1

$l = k\eta + 1$

$Z \sim U^l[S_i]$ // Draw l independent elements from S_i

For $1 \leq t \leq i$

$Z_t = \{x \in Z : Q(x) = t\}$ // Ask cluster assignment queries

$p = \arg\max_t |Z_t|$

$\mu_p' = \frac{1}{|Z_p|} \sum_{x \in Z_p} x$

<u>Phase 2</u>

// We know $\exists\ r_i$ such that $\forall x \in S_i$, $x \in C_p \Leftrightarrow d(x, \mu_p') < r_i$

// So r_p can be found by binary search.

$\hat{S}_i = \text{Sorted}(\{S_i\})$ // sort on basis of $d(x, \mu_p')$

$r_i = \text{BinarySearch}(\hat{S}_i)$ // Using up to $O(\log|S_i|)$ same cluster queries

$C_p' = \{x \in S_i : d(x, \mu_p') \le r_i\}$

$S_{i+1} = S_i \setminus C_p'$

$C = C \cup \{C_p'\}$

end

<u>Proof</u> $\rightarrow$ In first-phase, $l > k\eta$ cluster-assignment queries are made. By pigeonhole-principle, $|Z_p| > \eta$. So, by lemma 2, $d(\mu_p, \mu_p') \le r(C_p)\epsilon$ with probability $> 1 - \frac{\delta}{k}$. By lemma 1,

$\Rightarrow d(x, \mu_p') < d(y, \mu_p')\ \forall x \in C_p\ \&\ y \notin C_p$. Hence radius r_i found in phase 2 of algo is $r_i = \max_{x \in C_p} d(x, \mu_p')$. Hence $C_p' = C_p$

So, with probabability $\frac{\delta}{k}$, ~~con~~ one iteration of algo correctly finds all points in cluster C_p.

Let A_i denote the event that cluster $C_p \ne C_p'$, then

$P(A_i) < \frac{\delta}{k}$

By union-bound

$\Rightarrow P(A_1 \cup A_2 \cup \ldots A_k) < P(A_1) + P(A_2) \ldots P(A_k) < k\left(\frac{\delta}{k}\right) = \delta$

$\Rightarrow$ Probability of correct ~~no~~ clustering $> \boxed{1-\delta}$ Hence proved.

Theorem → The algo makes $O\left(k\log n + \frac{k^2(\log k + \log(\frac{1}{\delta}))}{(\gamma-1)^4}\right)$ same cluster queries to the oracle Q.

It runs in $O\left(kn\log n + k^2\frac{\log k + \log(\frac{1}{\delta})}{(\gamma-1)^4}\right)$ time

Proof → In each iteration → phase 2 has $O(\log n)$ queries
& phase 1 has $O(k\eta)$ assignment queries during the whole algo (by reusing results). ~~Same~~ assignment queries are same as k same-cluster queries
So we get $O(k\log n + k^2\eta)$ queries in total.
We sort S_i ~~Ek~~ in phase 2, → $O(n\log n)$ per iter.
Phase 1 runs in $O(k\eta)$ time ~~Ek~~ per iter.
So $O(kn\log n + k^2\eta)$
Hence proved.

Proof of lemma 2 → Define a uniform distribution U over C_p. Then, μ_p & μ_p' are real & empirical means.
statistics → Some inequality shows they are close enough.