Polynomial time algorithms for α-stable Instances for $\alpha < 2$

We study the clustering instance where the every pairwise distance is either 1 or 2. Let us assume the instance is 1.9-stable and is embedded in Euclidean Space. Therefore, we observe that the distance of a point p to it's optimal center must be 1 and to all other optimal centers must be 2. We also observe that the distance between any two optimal centers must be 2 since a in Euclidean Space, the center is the mean of all the points in the cluster.

We further consider the specific case where the size of each cluster is the same. Since we have k optimal clusters, each cluster has $\frac{n}{k}$ points. Note that a center c_i must have exactly $\frac{n}{k}$ points at a distance of 1 and the rest $n - \frac{n}{k}$ points at a distance 2. The distances between non centers can be either 1 or 2.

Let $B_1(p)$ be the set of points that are a distance 1 from p and let $B_2(p)$ be the set of points that are a distance 2. Note that $B_1(p) \cup B_2(p) = S$, where S is the set of all points. In the worst case, $|B_1(p)| = \frac{n}{k}$. We know that $c_i \in B_1(p)$ since each point must be a distance 1 to its own center. However, the rest of the points in $B_1(p)$ can belong to different clusters. For a sufficiently large k, this is true for all non center points.

1 Polynomial Time algorithm for points embedded in Euclidean Space

The first claim we make is that for any set of k non center points, the clustering induced by these points has a high cost.

Claim 1. *Given a set of k non center points $p_1, p_2 \dots p_k$, the cost of the clustering induced by these points is at least* $1.9n - 2.8k$.

Proof. Consider the following perturbation :

$$d'(p,q) = \begin{cases} 1.9d(p,q), & \text{if } p = c_i^*, q \in C_i^* \\ d(p,q), & \text{otherwise} \end{cases} \tag{1}$$

The cost of the optimal clustering under perturbation d' is $1.9(n-k)$, since each optimal center has $\frac{n}{k} - 1$ edges of weight 1 and there are k optimal centers. The cost of any other clustering that is not optimal, induced by k non-center points, is greater than $1.9(n-k)$. Therefore, without the perturbation, any set of k non center points induce a clustering that has cost at least $1.9(n-k) - (0.9k)$. The last term ensures that the perturbed distances are set back to 1. Therefore, the total cost is at least $1.9n - 2.8k$. □

Let P be the clustering induced by $p_1, p_2 \dots p_k$. We now bound the number of edges with weight 1 in P.

Claim 2. *The number of edges with weight 1 in P is at most* $0.1(n-k)$.

Proof. Let n_1 be the number of edges with weight 1 and n_2 be the number of edges with weight 2 in P. Since the total number of edges from $p_1, p_2 \dots p_k$ are $n - k$,

$$n_1 + n_2 = n - k \tag{2}$$

By Claim 1,

$$n_1 + 2n_2 \geq 1.9n - 2.8k \tag{3}$$

Solving for n_1, we get $n_1 \leq 0.1(n) - 0.8k < 0.1(n-k)$. □

Let us look at points that have exactly $\frac{n}{k} - 1$ edges of weight 1.

Corollary 3. *At most $\frac{k}{10}$ points in S have exactly $\frac{n}{k} - 1$ edges of weight 1, i.e $|B_1(p)| = \frac{n}{k} - 1$*

Proof. We begin by picking k random non center points. By Claim 2 at most $0.1(n-k)$ edges are of weight 1. Maximizing the number of points with $|B_1(p)| = \frac{n}{k} - 1$, we know that there can be at most $\frac{0.1(n-k)}{\frac{n}{k}-1} \leq 0.1(k)$ such points.

Next, we keep the points that have $|B_1(p)| = \frac{n}{k} - 1$ and replace the others with new points. Claim 2 still holds. Therefore, at most $\frac{k}{10}$ points in S can be such that $|B_1(p)| = \frac{n}{k} - 1$. □

Therefore, we can construct a list L such that $\forall p \in L$, $|B_1(p)| = \frac{n}{k} - 1$. We also know that $|L| \leq k + \frac{k}{10} = 11\frac{k}{10}$.

We can further reduce the list L to a list L' such that the number of optimal centers in L' is at most $\frac{k}{10}$ and the number of non centers in L' is also at most $\frac{k}{10}$. The number of centers removed are at least $\frac{9k}{10}$ and we can cluster these optimally.

1. For all $p, q \in L$, if $d(p,q) = 1$ join (p,q).

2. Extract all points of degree 0.

This procedure shows that we extract at least $\frac{9k}{10}$ optimal centers.

In the list L', we have a smaller clustering instance that is 1.9-stable. Let x be the number of optimal centers left in L'. We know x since we know exactly how many optimal centers we extracted and $x \leq \frac{k}{10}$. The number of points n' in L' are $x * \frac{n}{k}$. Therefore, we recurse with parameters $n = n'$ and $k = x$ on the set L'.

Therefore, we can find the optimal clustering for the { 1, 2 } instance embedded in Euclidean Space after recursing for $O(\log k)$ steps.

2 Generalization of Algorithm to $\alpha \geq 1 + \epsilon$

We assume our instance is α-stable, such that $\alpha \geq 1 + \epsilon$, for $\epsilon > 0$.

Claim 4. *Given a set of k non-center points $p_1, p_2, \ldots p_k$, the cost of the clustering induced by these points is at least $\alpha(n) - (2\alpha - 1)k$.*

Proof. We construct the same perturbation as Claim 1. The statement follows as before. □

Let P be the clustering induced by $p_1, p_2, \ldots p_k$. We now bound the number of edges with weight 1 in P.

Claim 5. *The number of edges with weight 1 in P is strictly less than $(2 - \alpha)(n - k)$*

Proof. Same as Claim 2. □

Corollary 6. *The number of points in S that have exactly $\frac{n}{k} - 1$ edges of weight 1 is strictly less than $(2 - \alpha)k$.*

Proof. By Claim 5, the total number of edges with weight 1 in P are at most $(2-\alpha)(n-k)$. Each cluster induces $\frac{n}{k}-1$ edges. Therefore, the total number of points in P with $\frac{n}{k}-1$ edges of weight 1 is at most $\frac{(2-\alpha)(n-k)}{\frac{n}{k}-1} = (2-\alpha)k$. □

Therefore, we can construct a List L such that $\forall p \in L$, $|B_1(P)| = \frac{n}{k}-1$. By Corollary 6, we know that there will be at most $(2-\alpha)k$ non-centers in L. Note, for $\alpha > 1+\epsilon$, the number of non-centers are strictly less than $(1-\epsilon)k < k$. Therefore, using the algorithm from section 1, we can guarantee that we extract at least 1 optimal clustering in each iteration and then recurse on the remaining set. Therefore, we recurse at most k times and obtain the optimal clustering.

Therefore, we can extract the optimal clustering for such an instance for any $\alpha > 1$. A similar algorithm can be constructed for the k-means objective. Thus, we can show that such instances of perturbation resilience are easy under the k-means and k-median objective. Since existing lower bound techniques like Balcan, Haghtalab and White (2016) and Ben-David and Reyzin (2014) reduce NP-Complete problems to instances where pair-wise distances are either 1 or 2, we can show that such techniques cannot be used to prove hardness results for k-means and k-median objectives.

3 Extension to Non-Euclidean Space

We now consider the general case where the distances between two optimal centers are not restricted to being weight 2. Note that since a point p is α times closer to its own center than any other center, the distance of p to other optimal centers must be 2.

We begin by showing that we can construct the list L by picking points that have a 1 degree (edges of weight 1) between $\frac{n}{k}-1$ and $\frac{n}{k}+k$. Since the cost of any k non center points must be more than $1.9n$, we know we can get at most $\frac{k}{10}$ non center points.

We use the following recursive algorithm to get an optimal clustering:

1. Construct list L by picking all points p such that $\frac{n}{k}-1 \leq |B_1(p)| \leq \frac{n}{k}+k$
2. For all $p \in S$, if $|B_1(p) \cap L| = 1$, connect (p,q), where q is a point in L s.t. $q \in B_1(p)$.
3. For all $p \in L$, if $deg(p) > 0$, add p to list L'.
4. For all $p \in L'$, if $|B_1(p) \setminus L| = \frac{n}{k}-1$, extract $p \cup B_1(p)$. Recurse.

Intuitively, we can recurse using the aforementioned algorithm if at each iteration we can extract at least 1 optimal cluster. The first key observation is that if our list L' has at least $\frac{k}{10}+1$ optimal centers, we can extract at least 1 optimal cluster. Note, we also assume that L' does not have any non center points. We first assume this is true and show how it implies a polynomial time algorithm. Then we prove that L' must have at least $\frac{k}{10}+1$ optimal centers and does not have any non center points.

Claim 7. *If List L' has at least $\frac{k}{10}+1$ optimal centers, we can extract at least one optimal clustering and recurse on the rest.*

Proof. For each point $c_i \in L'$, we look at the set $B_1(c_i)$. We know that all points in the optimal cluster C_i^* must lie in the set $B_1(c_i)$. In addition, c_i may be a distance 1 to other optimal centers, which can also lie in $B_1(c_i)$. Observe, no non center from any other optimal cluster can lie in $B_1(c_i)$. Therefore, the set $B_1(c_i) \setminus L$ does not contain any other optimal center, since all optimal centers lie in the list L. However, as a consequence, we may remove a point p from $B_1(c_i)$ such that $p \in C_i^*$ and $p \in L$. If this is the case, $|B_1(c_i) \setminus L| < \frac{n}{k} - 1$ and we don't extract such an optimal center and its neighborhood. Next, we observe that there can be at most $\frac{k}{10}$ such c_i's in L' since there are at most $\frac{k}{10}$ optimal centers. Therefore, there exists at least 1 center in L' such that $|B_1(c_j) \setminus L| = \frac{n}{k} - 1$ and for such a center, we extract the set $B_1(c_j) \setminus L$. Thus we know we have extracted an optimal cluster and can recurse on the remaining set. □

By Claim we know that in each step of the recursion, we extract at least 1 optimal center. Thus, we recurse at most k times. This yields a polynomail time algorithm for clustering such an instance.

It remains to be argued that list L' can be constructed such that $|L'| \geq \frac{k}{10} + 1$ and L' has no non-center points.

Claim 8. *Given a list L such that L contains all k optimal centers and at most $\frac{k}{10}$ non-centers, we can construct a list L' such that*

1. *L' does not contain any non-centers*
2. $|L'| \geq \frac{k}{10} + 1$

Proof. Let p be a point in set S. We observe that p is always distance 1 to its own center and distance 2 to other optimal centers. If p is a distance 1 to more than 1 point in L, then it must lie in B_1 for a non center. In this case, note that our algorithm will not connect p to any point in L. Therefore, the degree of all non-centers in L must be 0. Since L' only has points from L that have non-zero degree, L' does not contain any non-centers. This completes the proof of Statement 1.

Next, we observe that only points p s.t p lies in $B_1(q)$ where q is a non center in L can be a distance 1 to more than 1 point in L'. In other words, p is distance 1 to q and its optimal center. In the worst case, $|B_1(q)| \leq \frac{n}{k} + k$. Therefore, the total number of points that are distance 1 to more than 1 point in L' is at most $(\frac{n}{k} + k) * \frac{k}{10} \leq \frac{n+k^2}{10}$. Therefore, at least $n - \frac{n+k^2}{10} = \frac{9n-k^2}{10}$ points are distance 1 to exactly 1 point in L'. Since an optimal center can have at most $\frac{n}{k}$ points that are distance 1, we know that there are at least $\frac{\frac{9n-k^2}{10}}{\frac{n}{k}} = \frac{9k}{10} - \frac{k^2}{10n}$ points that have a non zero degree. Thus, if $\frac{9k}{10} - \frac{k^2}{10n} > \frac{k}{10}$, we can show that L' has at least $\frac{k}{10} + 1$ centers. Solving for k, we get that if $k < \sqrt{8n}$, then $|L'| > \frac{k}{10}$. This completes the proof of Statement 2.

□

3.1 $(1.5 + \epsilon)$-Stable Algorithm for { 1 , 2 } Instances

In general, we have the following pair of equations :

Let n_1 and n_2 be the number of edges in the induced clustering for a set of k non centers such that n_1 has wt 1 and n_2 has wt 2. The first is the cost equation, that follows from Claim 1

$$n_1 + 2n_2 > \alpha(n-k) \tag{4}$$

The second equation comes from total number of edges in an induced clustering.

$$n_1 + n_2 = n - k \tag{5}$$

Solving the two equations for a bound on n_1, we get

$$n_1 < (2-\alpha)n \tag{6}$$

Thus, the number of non centers that can have $\frac{n}{k} - 1$ edges of wt 1 is at most $\frac{(2-\alpha)n}{\frac{n}{k}-1} = (2-\alpha)k$. Therefore, the list L we construct can have at most $k + (2-\alpha)k$ points.

Note that we can do the following check to make sure the non centers don't have a k additional weight 1 edges to points outside L. For all $p \in L$, if $|B_1 \setminus L| \leq \frac{n}{k} - 1$, keep p, else discard p.

Thus, any non center in L can now have at most $\frac{n}{k} - 2$ points from outside L. Therefore, the total number of points that exist in the B_1 neighborhood of non centers in L is at most $\frac{n}{k} * (2-\alpha)k = (2-\alpha)n$. Therefore at least $n - (2-\alpha)n = (\alpha - 1)n$ points are distance 1 to exactly 1 point in L. Therefore, at least $\frac{(\alpha-1)n}{\frac{n}{k}} = (\alpha-1)k$ centers have non zero degree. If $(\alpha-1)k > (2-\alpha)k$, we can extract at least 1 center and recurse. This condition is satisfied for any $\alpha > 1.5$.