# Clustering Point Sets Revisited

**Md. Billal Hossain**
Department of Computer Science, University of Texas at Dallas, TX, USA

**Benjamin Raichel**
Department of Computer Science, University of Texas at Dallas, TX, USA

## Abstract

In the sets clustering problem one is given a collection of point sets $\mathcal{P} = \{P_1, \dots P_m\}$ in $\mathbb{R}^d$, where for any set of $k$ centers in $\mathbb{R}^d$, each $P_i$ is assigned to its nearest center as determine by some local cost functions. The goal is then to select a set of $k$ centers to minimize some global cost function of the corresponding local assignment costs. Specifically, we consider either summing or taking the maximum cost over all $P_i$, where for each $P_i$ the cost of assigning it to a center $c$ is either $\max_{p\in P_i} \|c-p\|$, $\sum_{p\in P_i} \|c-p\|$, or $\sum_{p\in P_i} \|c-p\|^2$.

Different combinations of the global and local cost functions naturally generalize the $k$-center, $k$-median, and $k$-means clustering problems. In this paper, we improve the prior results for the natural generalization of $k$-center, give the first result for the natural generalization of $k$-means, and give results for generalizations of $k$-median and $k$-center which differ from those previously studied.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** Clustering, $k$-center, $k$-median, $k$-means

**Digital Object Identifier** 10.4230/LIPIcs.WADS.2025.38

**Funding** Work on this paper was partially supported by NSF CAREER AF Award CCF-1750780 and NSF AF Award CCF-2311179.

## 1 Introduction

Clustering is one of the most well studied problems in computational geometry and computer science as a whole, with a variety of applications. For center based clustering, given a point set $P \subset \mathbb{R}^d$ and integer parameter $k > 0$, the goal is to select a set $C \subset \mathbb{R}^d$ of $k$ *centers*, so as to minimize some cost function of the distances determined by assigning each point in $P$ to its nearest center in $C$. Specifically, for any $p \in P$, let $\|p - C\|$ denote the distance from $p$ to its nearest center in $C$. Then the *$k$-center*, *$k$-median*, and *$k$-means* objectives are to find the set $C$ minimizing $\max_{p\in P} \|p - C\|$, $\sum_{p\in P} \|p - C\|$, and $\sum_{p\in P} \|p - C\|^2$, respectively.

$k$-center, $k$-median, and $k$-means clustering are all known to be NP-hard. $k$-center is NP-hard to approximate within any factor less than 2 in general metric spaces [13], and even in the plane is still hard to approximate within a factor of roughly 1.82 [7]. Conversely, both the standard greedy algorithm by Gonzalez [10] and the alternative scooping method by Hochbaum and Shmoys [12] achieve a 2-approximation for $k$-center. The $k$-median and $k$-means problems are both known to be hard to approximate within a $1 + \gamma$ factor for some constant $\gamma > 0$, even in $O(\log n)$ dimensional Euclidean space. Different values of $\gamma$ are known depending on whether the points are in low or high dimensional Euclidean space, some other metric norm, or a general metric space. Conversely, there are several constant factor approximation algorithms for both problems. For a more in depth discussion see [5] and references therein.

For points in $\mathbb{R}^d$, PTAS's exist for these center based clustering problems when $k$ and $d$ are bounded. For $k$-center, Agarwal and Procopiuc [1] gave a $(1 + \varepsilon)$-approximation in $O(n \log k) + (k/\varepsilon)^{O(k^{1-1/d})}$ time. For $k$-median and $k$-means, Har-Peled and Mazumdar [11] used coresets to achieve a $(1 + \varepsilon)$-approximation algorithm whose running time is linear in $n$.


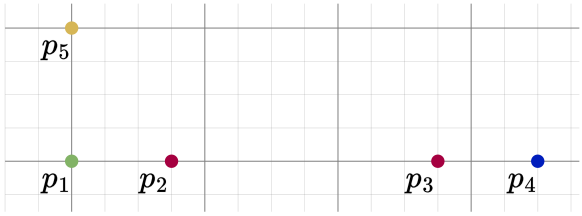


**Figure 1.1** Example showing sets clustering differs from regular clustering, where $\mathcal{P} = \{\{p_1\}, \{p_2, p_3\}, \{p_4\}, \{p_5\}\}$, $P = \{p_1, p_2, p_3, p_4, p_5\}$, and $k = 3$. For $k$-center/means/median clustering on $P$, the optimal centers are $p_5$, the midpoint of $(p_1, p_2)$ and the midpoint of $(p_3, p_4)$. However, the optimal centers for the $k$-center/means/median sets clustering variants on $\mathcal{P}$ that we consider (that is, $cost_{\infty,\infty}/cost_{1,2}/cost_{1,\infty}$) are the midpoint of $(p_1, p_5)$, the midpoint of $(p_2, p_3)$, and $p_4$.

Subsequent coreset based papers improve the time dependency on $k$, $d$, and $\varepsilon$ [4, 8]. Using a sampling based approach, [17] provided a $(1+\varepsilon)$-approximation with probability $\geq 1/2$ in $O(2^{(k/\varepsilon)^{O(1)}} dn)$ time, where the probability can be improved by boosting. In general, there are many approximation schemes for $k$-median and $k$-means in Euclidean settings, though we note a linear dependence on $d$ and $n$ is possible (e.g. [17]), and for $k$-median specifically a polynomial dependence on $k$ is possible for bounded $d$ (e.g. [11]).

### Sets Clustering

While standard clustering focuses on clustering points, naturally one can consider clustering more general geometric objects. In this paper, we consider the *sets clustering problem* [20], where one must cluster a collection of sets, $\mathcal{P} = \{P_1, \ldots P_m\}$, of total size $n = \sum_i |P_i|$. Here each $P_i$ can for example be viewed as a sample from a given object of study, whether say a physical object in 3d, or some class in a high dimensional feature space. With the interpretation that all points in $P_i$ arise from the same object, we thus naturally require they all be covered by the same center.

Given a center $c \in C$, we let $f(c, P_i)$ denote the cost of assigning $P_i$ to $c$, and let $f(C, P_i) = \min_{c \in C} f(c, P_i)$. Here we consider the cost functions $f_\infty(c, P_i) = \max_{p \in P_i} \|c - p\|$, $f_1(c, P_i) = \sum_{p \in P_i} \|c - p\|$, and $f_2(c, P_i) = \sum_{p \in P_i} \|c - p\|^2$. Note these cost functions are the 1-center, 1-median, and 1-means costs, and thus are motivated by the applications of those respective problems. However, unlike the 1-center/median/means problems, here we must cluster multiple sets and thus we need an aggregate cost. We consider two possibilities for the cost of clustering $\mathcal{P}$, namely $cost_{\infty,\beta}(C, \mathcal{P}) = \max_i f_\beta(C, P_i)$ and $cost_{1,\beta}(C, \mathcal{P}) = \sum_i f_\beta(C, P_i)$. Observe that when $\mathcal{P}$ is a collection of singleton points that $cost_{\infty,\infty} = cost_{\infty,1}$, $cost_{1,\infty} = cost_{1,1}$, and $cost_{1,2}$ respectively capture the $k$-center, $k$-median, and $k$-means objectives on the point set consisting of these singletons. Conversely, Figure 1.1 shows how when the sets in $\mathcal{P}$ are not singletons, the optimal solutions of the variants of sets clustering that we consider differ from the $k$-center, $k$-median, and $k$-means solutions on $\cup_i P_i$.

Several prior works considered different variants of the sets clustering problem, most notably [20]. Our aim is both to improve prior results, as well as to consider new variants of the sets clustering problem. We break our discussion into whether a particular sets clustering problem is most naturally viewed as a generalization of $k$-center, $k$-means, or $k$-median.

- **$k$-center:** Minimizing $cost_{\infty,\infty}(C, \mathcal{P})$ naturally generalizes the $k$-center problem. Previously, for points in constant dimensions, [20] provided an $O(n + m \log k)$ time 3-approximation as well as an $O(nk)$ time $(1 + \sqrt{3})$-approximation. These results require minimum enclosing ball (MEB) computations, and thus have running times with hidden

constants depending on $d$, though they remark that for higher dimensions they can use a $(1+\varepsilon)$-approximate MEB at the cost of $+\varepsilon$ in the approximation quality.

In Section 3, for points in any dimension $d$, we show that a natural adaptation of the standard $O(dnk)$ time greedy 2-approximation algorithm for $k$-center remains an $O(dnk)$ time 2-approximation for the sets clustering problem, a fact which surprisingly appears to have been overlooked in prior work. Moreover, our 2-approximation algorithm works in any metric space, and thus is optimal for general metric spaces (as it captures $k$-center, which is hard to approximate within a factor of 2). Additionally, for any constant $d$, using a standard grid based approach we provide an $O(nk^{k+1}/\varepsilon^{dk})$ time $(1+\varepsilon)$-approximation, that is, a linear time algorithm when $k$, $d$, and $\varepsilon$ are constants.

- **$k$-means:** Minimizing $cost_{1,2}(C,\mathcal{P})$ naturally generalizes the $k$-means problem, and we study this objective in Section 4. Surprisingly, it appears this cost function for sets clustering has not been studied in prior works, despite the ubiquity of $k$-means clustering. Consider the point set $P$ obtained by replacing each set $P_i$ with $|P_i|$ copies of its centroid. Using previously observed facts about the geometry of the $k$-means objective, we prove the strong statement that an $\alpha$-approximation to the $k$-means clustering problem on $P$ is also an $\alpha$-approximation for the sets clustering problem.
- **$k$-median:** Minimizing $cost_{1,1}(C,\mathcal{P})$ generalizes the $k$-median problem. Previously, [20] provided a polynomial time $(3+\varepsilon)$ approximation in $\mathbb{R}^d$ for constant $d$, by replacing each set $P_i$ with $|P_i|$ copies of its $(1+\varepsilon)$-approximate 1-median, and then applying a $(1+\varepsilon)$-approximation to $k$-median for constant $d$ (such as [11]). In Section 5, we instead consider the problem of minimizing $cost_{1,\infty}(C,\mathcal{P})$, which is also equivalent to $k$-median when $\mathcal{P}$ consists of singletons.[1] Minimizing $cost_{1,\infty}(C,\mathcal{P})$ models the case where how well a center covers a set is determined by the furthest point in the set from the center (i.e. the minimum radius ball at the center enclosing the whole set). To address this problem we apply a similar approach as [20], replacing each set with the center of its minimum enclosing ball. We argue that this provides a $(1+\alpha)$-approximation where $\alpha$ is the approximation quality of the $k$-median subroutine.
- **$k$-center Alternative:** In Section 6 we considered the problem of minimizing $cost_{\infty,1}(C,\mathcal{P})$. Similar to the $cost_{\infty,\infty}$ considered in Section 3 and prior work, this problem also generalizes standard $k$-center clustering. This problem is also related to the $cost_{1,\infty}$ problem considered in Section 5, but where we inverted the max and sum operators. This problem is more challenging, though we still can provide a polynomial time $(3+\varepsilon)$-approximation for point sets in the plane and for any constant $0<\varepsilon\leq 1$.

We emphasize that the algorithms in sections 3, 4, and 5 for the $cost_{\infty,\infty}, cost_{1,2}$, and $cost_{1,\infty}$ objectives, either modify or directly reduce to the algorithms respectively used for $k$-center, $k$-means, and $k$-median. Thus our running times are virtually equivalent to the algorithms used for these standard clustering problems, which is the best one could hope for as these standard clustering problems are special cases of our sets clustering problems. On the other hand, our polynomial time approximation for the more challenging $cost_{1,\infty}(C,\mathcal{P})$ in Section 6 is slower, and we leave it as an open problem for future work to optimize the time.

We study the combinations of maximums, sums, and sums of squares which we believe to be the most natural, though other combinations exist that we do not study, such as squaring sums of squares. Several prior works also considered clustering based on the closest point in a

[1] There is also a loose connection between $cost_{1,\infty}(C,\mathcal{P})$ and clustering to minimize the sum of radii [3], though unlike $k$-median there is no immediate reduction.

set to its nearest center. Specifically, let $f_0(C,P) = \min_{c\in C, p\in P} \|c-p\|$, then [15] considered minimizing $\sum_{P_i\in\mathcal{P}} f_0(C,P_i)^2$, providing a near linear time $(1+\varepsilon)$-approximation for the case when $k$, $d$, and $z$ are constants, where $z = \max_i |P_i|$. [14] considered the problem of minimizing $\max_{P_i\in\mathcal{P}} f_0(C,P_i)$, where the $P_i$ sets are continuous convex regions, providing a number of results such as a $(5+2\sqrt{3})$-approximation for the case of disks. Many other papers have considered clustering other continuous geometric objects, and in particular unbounded objects such as lines or hyperplanes [9, 18, 6], though such results are less relevant than the above mentioned papers on clustering discrete sets.

## 2 Setup

For two points $p, q \in \mathbb{R}^d$, we use $\|p-q\|$ to denote the Euclidean distance between $p$ and $q$. Similarly, for a point $q \in \mathbb{R}^d$ and a finite point set $P \subset \mathbb{R}^d$ we have $\|q-P\| = \min_{p\in P} \|q-p\|$.

Given a point set $P \subset \mathbb{R}^d$ and a set of $k$ centers $C \subset \mathbb{R}^d$, the standard cost functions for $k$-center, $k$-median, and $k$-means clustering are as follows.

- $kcenter(C,P) = \max_{p\in P} \|p-C\|$
- $kmedian(C,P) = \sum_{p\in P} \|p-C\|$
- $kmeans(C,P) = \sum_{p\in P} \|p-C\|^2$

Given a parameter $k$ and point set $P \subset \mathbb{R}^d$, the goal of $k$-center, $k$-median, or $k$-means clustering is then to find a set $C$ of $k$ centers minimizing the respective cost function.

We now generalize these notions to collections of point sets. So let $\mathcal{P} = \{P_1, \ldots P_m\}$ be a collection of $m$ sets of points, where $P_i \subset \mathbb{R}^d$ for all $i$, and $n = \sum_i |P_i|$ is the total size. For a point $c \in \mathbb{R}^d$, let $f_\beta(c,P_i)$ be some non-negative function, representing the cost of assigning point set $P_i$ to a center $c$. We consider three cost functions denoted by $\beta = \infty$, 1, or 2.[2]

- $f_\infty(c,P_i) = \max_{p\in P_i} \|c-p\|$
- $f_1(c,P_i) = \sum_{p\in P_i} \|c-p\|$
- $f_2(c,P_i) = \sum_{p\in P_i} \|c-p\|^2$

For a single point $p$ we write $f_\beta(c,p) = f_\beta(c,\{p\})$, where in particular we have $f_\infty(c,p) = f_1(c,p) = \|c-p\|$ and $f_2(c,p) = \|c-p\|^2$

Given a set $C = \{c_1, \ldots, c_k\} \subset \mathbb{R}^d$ of $k$ centers, define $f_\beta(C,P_i) = \min_{c\in C} f_\beta(c,P_i)$, that is $P_i$ is assigned to its nearest center under the function $f_\beta$. Now the $f_\beta(C,P_i)$ values over all $i$ define a vector of length $m$, and we consider either the $\ell_\infty$ or $\ell_1$ norm. Namely, we define the cost functions, $cost_{\alpha,\beta}$, for $\alpha = \infty$ or 1 as follows:

- $cost_{\infty,\beta}(C,\mathcal{P}) = \max_i f_\beta(C,P_i)$
- $cost_{1,\beta}(C,\mathcal{P}) = \sum_i f_\beta(C,P_i)$

Finally, let $optcost_{\alpha,\beta}(k,\mathcal{P}) = \min_{C\subseteq\mathbb{R}^d, |C|=k} cost_{\alpha,\beta}(C,\mathcal{P})$, and let $opt_{\alpha,\beta}(k,\mathcal{P})$ denote a set $C$ of $k$ centers achieving $optcost_{\alpha,\beta}(k,\mathcal{P})$. For some $\gamma \geq 1$, a set $C \subset \mathbb{R}^d$ with $|C| = k$ is referred to as a *$\gamma$-approximation* to $opt_{\alpha,\beta}(k,\mathcal{P})$ if $cost_{\alpha,\beta}(C,\mathcal{P}) \leq \gamma \cdot optcost_{\alpha,\beta}(k,\mathcal{P})$. (That is, $k$ is a hard constraint and the approximation is on the cost, as is standard for clustering.)

▶ **Observation 1.** *Given a point set $P$, let $\mathcal{P}(P)$ denote the collection of $|P|$ singleton sets $\{p\}$ for each $p \in P$. Then $kcenter(C,P) = cost_{\infty,\infty}(C,\mathcal{P}(P)) = cost_{\infty,1}(C,\mathcal{P}(P))$, $kmedian(C,P) = cost_{1,\infty}(C,\mathcal{P}(P)) = cost_{1,1}(C,\mathcal{P}(P))$, and $kmeans(C,P) = cost_{1,2}(C,\mathcal{P}(P))$.*

[2] The different $\beta$ values are supposed to be vaguely reminiscent of corresponding vector norms.

## 3 k-center Sets Clustering

Given a collection of point sets $\mathcal{P} = \{P_1, \dots P_m\}$ in $\mathbb{R}^d$ and a parameter $k$, [20] considered the problem of covering $\mathcal{P}$ with $k$ equal radius balls of minimum radius such that each $P_i$ is entirely contained in one of the balls. Using the terminology defined above, this is equivalent to computing $opt_{\infty,\infty}(k, \mathcal{P})$. For this problem, [20] gave a $(1+\sqrt{3})$-approximation. Here we show that the standard greedy Gonzalez algorithm for $k$-center clustering can be adapted to yield a 2-approximation. Not only is this a better approximation ratio, but this algorithm will in fact work for any metric space. Moreover, for general metrics it is not possible to get a $2-\varepsilon$ approximation for $k$-center clustering for any $\varepsilon > 0$, unless P=NP [13]. As $k$-center is a special case of our problem, our approximation algorithm is thus optimal.

Given a point set $P$ and a parameter $k$, we recall that the Gonzalez algorithm [10] greedily outputs a set of $k$ center $c_1, \dots, c_k$, where $c_1$ is an arbitrary point in $P$, and for $i > 1$, $c_i = \arg\max_{p \in P} \|p - \{c_1, \dots, c_{i-1}\}\|$. We argue the following natural adaptation of this algorithm to our sets clustering problem, which greedily picks the furthest set rather than the furthest point, also yields a 2-approximation. (Figure 1.1 shows that naively running Gonzalez on $P = \cup_i P_i$ does not achieve a 2-approximation. Namely, Gonzalez on $P$ might pick $p_1, p_4$, and $p_5$ as the centers, resulting in a cost of 11 for sets clustering on $\mathcal{P}$, however, choosing $p_1$, the midpoint of $(p_2, p_3)$, and $p_4$, gives a cost of 4.)

**Algorithm 1** **greedy**$(k, \mathcal{P} = \{P_1, \dots, P_m\})$.

```
Initialize all point sets P_1, P_2, ..., P_m as unmarked.
Initialize C = {c_1}, where c_1 is an arbitrary point from P_1, and mark P_1
for i = 2 to k do
    P' = arg max_{unmarked P ∈ P} f_∞(C, P)
    Set C = C ∪ {c'} where c' is an arbitrary point from P', and mark P'.
return C
```

▶ **Theorem 2.** *Given a parameter $k$ and a collection of point sets $\mathcal{P} = \{P_1, \dots P_m\}$ in $\mathbb{R}^d$,* **greedy**$(k, \mathcal{P})$ *gives an $O(dnk)$ time 2-approximation for computing $opt_{\infty,\infty}(k, \mathcal{P})$, where $n = \sum_i |P_i|$.*

**Proof.** Let $C = \{c_1, \dots, c_k\}$ be the output of **greedy**$(k, \mathcal{P})$ and let $C^* = \{c_1^*, c_2^*, \dots, c_k^*\} = opt_{\infty,\infty}(k, \mathcal{P})$ with cost $r^* = optcost_{\infty,\infty}(k, \mathcal{P}) = \min_{C \subseteq \mathbb{R}^d, |C|=k} cost_{\infty,\infty}(C, \mathcal{P})$, where recall $cost_{\infty,\infty}(C, \mathcal{P}) = \max_i f_\infty(C, P_i)$.

There are two possible cases. In the first case, for all $c^* \in C^*$, there exists some $c \in C$ such that $\|c^* - c\| \leq r^*$. Consider any set $P_i$ with $c^*$ as its nearest optimal center under $f_\infty$, that is $f_\infty(c^*, P_i) = f_\infty(C^*, P_i)$. By definition, $f_\infty(c^*, P_i) = \max_{p \in P_i} \|c^* - p\| \leq r^*$. Thus for any point $p \in P_i$, by the triangle inequality, $\|p - c\| \leq \|p - c^*\| + \|c^* - c\| \leq 2r^*$. Therefore, $f_\infty(c, P_i) = \max_{p \in P_i} \|c - p\| \leq 2r^*$. This in turn implies $f_\infty(C, P_i) \leq 2r^*$, and as this holds for all $i$ it implies $C$ is a 2-approximation.

In the second case, there exists some $c^* \in C^*$, such that for any $c \in C$ we have $\|c^* - c\| > r^*$. For a given $c \in C$, let $P(c)$ denote the set $P_i \in \mathcal{P}$ from which $c$ was selected in **greedy**$(k, \mathcal{P})$. Observe that if $\|c^* - c\| > r^*$ then, $f_\infty(c^*, P(c)) > r^*$, that is $P(c)$ is not covered by $c^*$ in the optimal solution. However, $f_\infty(C^*, P_i) \leq r^*$ for all $i$, and so by the pigeon hole principle if such a $c^* \in C^*$ occurs then there must exist some other $c_j^* \in C^*$ and centers $c_\alpha, c_\beta \in C$ with $\alpha < \beta$ such that $f_\infty(c_j^*, P(c_\alpha)), f_\infty(c_j^*, P(c_\beta)) \leq r^*$. This implies $\|c_j^* - p\| \leq r^*$ for any $p \in P(c_\alpha) \cup P(c_\beta)$. Thus by the triangle inequality, for any $p \in P(c_\beta)$ we have $\|c_\alpha - p\| \leq \|c_\alpha - c_j^*\| + \|c_j^* - p\| \leq 2r^*$, which implies $f_\infty(c_\alpha, P(c_\beta)) \leq 2r^*$.

For any $1 \leq i \leq k$, let $C_i = \{c_1, \ldots, c_i\}$, i.e. the first $i$ centers chosen by $\mathbf{greedy}(k, \mathcal{P})$, and let $\delta_i = f_\infty(C_{i-1}, P(c_i))$ (where $\delta_1 = \infty$). Denote the cost of the final solution $C$ output by $\mathbf{greedy}(k, \mathcal{P})$ as $\delta_{k+1} = \max_{P \in \mathcal{P}} f_\infty(C, P)$. Observe that $\delta_1 \geq \delta_2 \geq \ldots \geq \delta_k \geq \delta_{k+1}$. Above we argued that $f_\infty(c_\alpha, P(c_\beta)) \leq 2r^*$ where $\alpha < \beta$, thus $\delta_{k+1} \leq \delta_\beta = f_\infty(C_{\beta-1}, P(c_\beta)) \leq f_\infty(C_\alpha, P(c_\beta)) \leq f_\infty(c_\alpha, P(c_\beta)) \leq 2r^*$, and thus $C$ is a 2-approximation.

As for the running time, we can achieve $O(dnk)$ time in a similar fashion to the standard $k$-center Gonzalez algorithm. Specifically, let $C_i$ be the set of centers after some $i$ iterations of the algorithm, and assume for all unmarked $P \in \mathcal{P}$ we maintain $f_\infty(C_i, P)$. Then in the $i+1$ round, $c_{i+1}$ is some arbitrary point from $\arg\max_{\text{unmarked } P \in \mathcal{P}} f_\infty(C_i, P)$, which can thus be found by a linear scan of the $f_\infty(C_i, P)$ values. We also need to update the $f_\infty(C_i, P)$ values to $f_\infty(C_{i+1}, P)$. To do so observe that for any $P \in \mathcal{P}$, $f_\infty(C_{i+1}, P) = \min\{f_\infty(C_i, P), f_\infty(c_{i+1}, P)\}$. As $f_\infty(c_{i+1}, P)$ can be computed in $O(d|P|)$ time, we can thus update all of these values in $O(dn)$ time. As the algorithm performs $k$ iterations, it thus takes $O(dnk)$ time overall. ◀

The algorithm used in [20] for achieving the $O(1+\sqrt{3})$-approximation requires computing the minimum enclosing ball (MEB) for each point set, and thus has a hidden constant in the running time depending on $d$. Thus for high dimensions, [20] use a $(1+\varepsilon)$-approximate MEB (which can be computed in $O(dn/\varepsilon + 1/\varepsilon^5)$ time [2]), resulting in a $O(1+\sqrt{3}+\varepsilon)$-approximation factor. In comparison, our algorithm avoids computing the MEB altogether, and as the above proof only relied on the triangle inequality, the same proof verbatim yields a 2-approximation for any metric space.[3]

$k$-center clustering is known to be hard to approximate within a factor of roughly 1.82 even in the plane [7], however, a $(1+\varepsilon)$-approximation is possible in constant dimensions, though at the cost of having a time exponential in $k$. We now show this standard algorithm also applies to our problem.

We will make use of the following standard observation (adapted for our settings).

▶ **Observation 3.** *Let $C = \{c_1, \ldots, c_k\} \subset \mathbb{R}^d$ be a set of $k$ points, and let $C' = \{c'_1, \ldots, c'_k\}$ be a set of $k$ points such that $\|c_i - c'_i\| \leq x$. Then for any collection of point sets $\mathcal{P} = \{P_1, \ldots P_m\}$, we have that $cost_{\infty,\infty}(C', \mathcal{P}) \leq cost_{\infty,\infty}(C, \mathcal{P}) + x$. This follows since if for a given set $P_j \in \mathcal{P}$ its nearest center under $f_\infty$ was $c_i$, then by the triangle inequality we have that $f_\infty(c'_i, P_j) \leq f_\infty(c_i, P_i) + x$.*

▶ **Theorem 4.** *Given a parameter $k$ and a collection of point sets $\mathcal{P} = \{P_1, \ldots P_m\}$ in $\mathbb{R}^d$, for constant $d$, then for any $\varepsilon > 0$ there is an $O(nk^{k+1}/\varepsilon^{dk})$ time $(1+\varepsilon)$-approximation for computing $opt_{\infty,\infty}(k, \mathcal{P})$, where $n = \sum_i |P_i|$.*

**Proof.** Let $C = \{c_1, c_2, \ldots, c_k\}$ be the centers output by $\mathbf{greedy}(k, \mathcal{P})$, and let $r = cost_{\infty,\infty}(C, \mathcal{P})$. Also let $C^* = \{c^*_1, c^*_2, \ldots, c^*_k\} = opt_{\infty,\infty}(k, \mathcal{P})$ where $r^* = cost_{\infty,\infty}(C^*, \mathcal{P})$. By Theorem 2 we know that $r^* \leq r \leq 2r^*$.

Let $\mathcal{B}(x) = \cup_i B(c_i, x)$ where $B(c_i, x)$ is the ball of radius $x$ centered at $c_i$. By definition of $C$ and $r$, we know $\cup_i P_i \subseteq \mathcal{B}(r)$. We can assume that every center $c^* \in C^*$ is within distance $r^* \leq r$ of some point in $\cup_i P_i$, as otherwise $c^*$ can be deleted without affecting the optimal solution quality. Thus we have that $C^* \subseteq \mathcal{B}(2r)$.

Now consider the axis aligned grid over $\mathbb{R}^d$ with cell side length $\delta = \varepsilon r/2\sqrt{d}$. Any ball of radius $2r$ intersects at most $O((2r/\delta)^d) = O(1/\varepsilon^d)$ cells, for constant $d$. Thus $\mathcal{B}(2r)$ intersects $O(k/\varepsilon^d)$ grid cells, and as $C^* \subseteq \mathcal{B}(2r)$, there are thus $O(k/\varepsilon^d)$ possible cells for each center

[3] This is our only result that holds for arbitrary metric spaces, which is why $opt_{\alpha,\beta}$ was defined for points specifically in $\mathbb{R}^d$.

in $C^*$. Now for any point $q = (q_1, \ldots, q_d) \in \mathbb{R}^d$, let $grid_\delta(q) = (\delta\lfloor q_1/\delta \rfloor, \ldots, \delta\lfloor q_d/\delta \rfloor)$ be the lowest corner of the grid cell containing $q$ (specifically the coordinate-wise minimal corner). Let $grid_\delta(C^*) = \{grid_\delta(c_1^*), \ldots, grid_\delta(c_k^*)\}$. As the diameter of each cell is $\varepsilon r/2$, by Observation 3 we have that $cost_{\infty,\infty}(grid_\delta(C^*), \mathcal{P}) \leq cost_{\infty,\infty}(C^*, \mathcal{P}) + \varepsilon r/2 = r^* + \varepsilon r/2 \leq r^* + \varepsilon r^* = (1+\varepsilon)r^*$.

The above motivates the following algorithm. First compute $C = \{c_1, c_2, \ldots, c_k\}$ using **greedy**$(k, \mathcal{P})$. This determines a set of $O(k/\varepsilon^d)$ grid cells around the centers of $C$ which must contain $C^*$. We try all possible $O((k/\varepsilon^d)^k)$ possibilities for these $k$ cells and for each possibility we use the lowest corners of the respective cells as our candidate set of centers. We then return as our solution the set of such centers $C'$ which had the minimum $cost_{\infty,\infty}(C', \mathcal{P})$ value. As $grid_\delta(C^*)$ will be one of the possible sets of centers considered, where we already argued that $cost_{\infty,\infty}(grid_\delta(C^*), \mathcal{P}) \leq (1+\varepsilon)r^*$, this procedure is guaranteed to output a $(1+\varepsilon)$-approximation.

As for the running time, the initial call to **greedy**$(k, \mathcal{P})$ takes $O(nk)$ time. There are $O((k/\varepsilon^d)^k)$ subsets of $k$ cells we consider, and for each one we first compute the set $C'$ of lower corners in $O(k)$ time. Then we compute $cost_{\infty,\infty}(C', \mathcal{P})$, which can be done in $O(nk)$ time. Thus the overall time is $O(nk(k/\varepsilon^d)^k) = O(nk^{k+1}/\varepsilon^{dk})$ ◀

▶ Remark 5. For $k$-center clustering it is common to consider the discrete version of the problem where the centers must come from the input point set. One can easily argue that both Theorem 2 and Theorem 4 extend to this variant. (In the proof of Theorem 4, we would restrict to non-empty grid cells for potential center locations.)

## 4 k-means Sets Clustering

Given $\mathcal{P} = \{P_1, \ldots P_m\}$ and a parameter $k$, for any set $C \subset \mathbb{R}^d$ of $k$ centers recall that

$$cost_{1,2}(C, \mathcal{P}) = \sum_{i=1}^{m} f_2(C, P_i) = \sum_{i=1}^{m} \min_{c \in C} f_2(c, P_i) = \sum_{i=1}^{m} \min_{c \in C} \sum_{p \in P_i} \|c - p\|^2.$$

Note that this is the natural analogue of the $k$-means problem to clustering point sets. Specifically, when each set is a single point, that is $P_i = \{p_i\}$ for all $i$, then this is equivalent to the standard $k$-means clustering cost on the point set $P = \{p_1, \ldots, p_m\}$, as then $cost_{1,2}(C, \mathcal{P}) = \sum_{i=1}^{m} \|p_i - C\|^2$

▶ **Definition 6.** *For a given point set $P \subset \mathbb{R}^d$, the* centroid *of $P$ is defined as $\bar{p} = \bar{p}(P) = \frac{\sum_{p \in P} p}{|P|}$.*

*Given a collection of sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^d$, let $\bar{P}_i = \{\bar{p}_{i1}, \ldots, \bar{p}_{i|P_i|}\}$ be the set of $|P_i|$ distinct copies of $\bar{p}_i = \bar{p}(P_i)$, and let $\bar{P}(\mathcal{P}) = \cup_i \bar{P}_i$. Note, while $\bar{p}_{ij}$ and $\bar{p}_{ik}$ are collocated they are viewed as distinct points, and thus $|\bar{P}_i| = |P_i|$ and $|\bar{P}(\mathcal{P})| = \sum_i |P_i|$.*

It is well known that the optimal solution to the 1-mean problem with respect to a point set $P$ is the centroid $C = \{\bar{p}(P)\}$. We require the following standard lemma, whose proof can be found in [16].

▶ **Lemma 7.** *For a point set $P \subset \mathbb{R}^d$ with centroid $\bar{p} = \bar{p}(P)$ and any point $x \in \mathbb{R}^d$, we have, $f_2(x, P) = f_2(\bar{p}, P) + |P| \cdot f_2(x, \bar{p})$.*

Given a collection of point sets, we now argue that one can solve the sets clustering problem by solving the $k$-means problem on the centroids of the sets.

▶ **Theorem 8.** *For a parameter $k$ and a point set $P \subset \mathbb{R}^d$, let* **kmeansAlg**$(k, P)$ *denote any algorithm which achieves an $\alpha$-approximation for the $k$-means problem.*

*Given a parameter $k$ and collection of sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^d$,* **kmeansAlg**$(k, \bar{P}(\mathcal{P}))$ *is an $\alpha$-approximation for computing $opt_{1,2}(k, \mathcal{P})$.*

**Proof.** For any two point sets $C, P \subset \mathbb{R}^d$, by lemma 7 we have:

$$f_2(C, P) = \min_{c \in C} f_2(c, P) = \min_{c \in C}(f_2(\bar{p}, P) + |P| \cdot f_2(c, \bar{p})) = f_2(\bar{p}, P) + |P| \cdot \min_{c \in C} f_2(c, \bar{p}).$$

Recall from Section 2 that $kmeans(C, P) = \sum_{p \in P} \|p - C\|^2 = \sum_{p \in P} \min_{c \in C} f_2(c, p)$. Thus summing the above equation for $P_i$ over all $i$ gives:

$$\begin{aligned} cost_{1,2}(C, \mathcal{P}) &= \sum_{i=1}^{m} f_2(C, P_i) = \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + \sum_{i=1}^{m} |P_i| \cdot \min_{c \in C} f_2(c, \bar{p}_i) \\ &= \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + \sum_{p \in \bar{P}(\mathcal{P})} \min_{c \in C} f_2(c, p) = \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + kmeans(C, \bar{P}(\mathcal{P})). \end{aligned}$$

Thus $cost_{1,2}(C, \mathcal{P}) = \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + kmeans(C, \bar{P}(\mathcal{P}))$. So let $C^*$ denote the optimal $k$-means solution on $\bar{P}(\mathcal{P})$, that is $C^* = \arg\min_{C \subset \mathbb{R}^d, |C|=k} kmeans(C, \bar{P}(\mathcal{P}))$. As $\sum_{i=1}^{m} f_2(\bar{p}_i, P_i)$ does not depend on $C$, the above equation implies that minimizing $kmeans(C, \bar{P}(\mathcal{P}))$ also minimizes $cost_{1,2}(C, \mathcal{P})$, i.e. $C^* = \arg\min_{C \subset \mathbb{R}^d, |C|=k} cost_{1,2}(C, \mathcal{P})$.

So let $C'$ be the $\alpha \geq 1$ approximation returned by **kmeansAlg**$(k, \bar{P}(\mathcal{P}))$, meaning $kmeans(C', \bar{P}(\mathcal{P})) \leq \alpha \cdot kmeans(C^*, \bar{P}(\mathcal{P}))$. Then we have

$$\begin{aligned} cost_{1,2}(C', \mathcal{P}) &= \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + kmeans(C', \bar{P}(\mathcal{P})) \\ &\leq \sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + \alpha \cdot kmeans(C^*, \bar{P}(\mathcal{P})) \\ &\leq \alpha(\sum_{i=1}^{m} f_2(\bar{p}_i, P_i) + kmeans(C^*, \bar{P}(\mathcal{P}))) = \alpha \cdot cost_{1,2}(C^*, \mathcal{P}) \end{aligned}$$ ◀

As discussed in the introduction, there are many known $(1+\varepsilon)$-approximation algorithms for $k$-means, with various trade-offs in the running times depending on the parameters $n, d, k$, and $\varepsilon$. Moreover, as $k$-means is a special case of our problem, we should not expect our times to be faster than what is possible for $k$-means.

Before we can apply such algorithms the above theorem requires we compute $\bar{P}(\mathcal{P})$. However, this takes only $O(dn)$ time, where $n = \sum_i |P_i|$, as the centroid of a set is simply the average of the points. Thus the above theorem immediately implies that we have the following.

▶ **Corollary 9.** *For a parameter $k$ and set $P \subset \mathbb{R}^d$ of $n$ points, let $T(n, k, d, \varepsilon)$ denote the running time of any algorithm which achieves a $(1+\varepsilon)$-approximation for $k$-means.*

*Given a parameter $k$ and collection of sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^d$, where $n = \sum_i |P_i|$, there is a $(1+\varepsilon)$-approximation for computing $opt_{1,2}(k, \mathcal{P})$ with $O(dn + T(n, k, d, \varepsilon))$ running time.*

## 5 k-median Sets Clustering

Given $\mathcal{P} = \{P_1, \dots P_m\}$ and a parameter $k$, for any set $C \subset \mathbb{R}^d$ of $k$ centers recall that

$$cost_{1,\infty}(C, \mathcal{P}) = \sum_{i=1}^{m} f_\infty(C, P_i) = \sum_{i=1}^{m} \min_{c \in C} f_\infty(c, P_i) = \sum_{i=1}^{m} \min_{c \in C} \max_{p \in P_i} \|c - p\|.$$

As discussed in the introduction, the above cost (as well as $cost_{1,1}(C, \mathcal{P})$ studied in [20]) generalizes the $k$-median problem to clustering point sets. Specifically, when each set consists of a single point, i.e., $P_i = \{p_i\}$ for all $i$, the problem reduces to the standard $k$-median clustering cost on the point set $P = \{p_1, \dots, p_m\}$, as then $cost_{1,\infty}(C, \mathcal{P}) = \sum_{i=1}^{m} \|p_i - C\|$.

▶ **Lemma 10.** *Let $\mathcal{P} = \{P_1, \dots, P_m\}$ be a collection of sets and $B = \{b_1, \dots, b_m\}$ where $b_i$ is the center of the minimum enclosing ball of $P_i$, then $\sum_{i=1}^{m} \max_{p \in P_i} \|p - b_i\| \leq optcost_{1,\infty}(k, \mathcal{P})$.*

**Proof.** Since $b_i$ is the center of the minimum enclosing ball of $P_i$, by definition we have that $\max_{p \in P_i} \|p - b_i\| = \min_{b \in \mathbb{R}^d} \max_{p \in P_i} \|p - b\| = \min_{b \in \mathbb{R}^d} f_\infty(b, P_i)$. So if $C^* = \{c_1^*, c_2^*, \dots, c_k^*\} = opt_{1,\infty}(k, \mathcal{P})$, then,

$$\sum_{i=1}^{m} \max_{p \in P_i} \|p - b_i\| = \sum_{i=1}^{m} \min_{b \in \mathbb{R}^d} f_\infty(b, P_i) \leq \sum_{i=1}^{m} \min_{c \in C^*} f_\infty(c, P_i) = \sum_{i=1}^{m} f_\infty(C^*, P_i) = optcost_{1,\infty}(k, \mathcal{P})$$

◀

▶ **Lemma 11.** *Let $\mathcal{P} = \{P_1, \dots, P_m\}$ be a collection of point sets from $\mathbb{R}^d$, let $B = \{b_1, \dots, b_m\}$ where $b_i$ is the center of the minimum enclosing ball of $P_i$, and let $C \subset \mathbb{R}^d$ be any set of $k$ centers, then $kmedian(C, B) \leq cost_{1,\infty}(C, \mathcal{P})$.*

**Proof.** First, we make the standard observation that for any point $q \in \mathbb{R}^d$, $\|q - b_i\| \leq \max_{p \in P_i} \|p - q\|$. (Assume $q \neq b_i$ as otherwise the inequality trivially holds.) This holds by considering the hyperplane passing through $b_i$ whose normal is in the direction from $b_i$ to $q$. As $b_i$ is the center of the minimum enclosing ball of $P_i$, the closed halfspace defined by this hyperplane and not containing $q$ must contain a point from $P_i$, and thus this point is at least as far as $b_i$ from $q$. Thus we have,

$$\begin{aligned} kmedian(C, B) &= \sum_{i=1}^{m} \|b_i - C\| = \sum_{i=1}^{m} \min_{c \in C} \|b_i - c\| \leq \sum_{i=1}^{m} \min_{c \in C} \max_{p \in P_i} \|p - c\| \\ &= \sum_{i=1}^{m} f_\infty(C, P_i) = cost_{1,\infty}(C, \mathcal{P}) \end{aligned}$$

◀

▶ **Theorem 12.** *For a parameter $k$ and a point set $P \subset \mathbb{R}^d$, let* **kmedianAlg**$(k, P)$ *denote any algorithm which achieves an $\alpha$-approximation for the $k$-median problem.*

*Given a parameter $k$ and collection of sets $\mathcal{P} = \{P_1, \dots, P_m\}$ in $\mathbb{R}^d$, let $B = \{b_1, \dots, b_m\}$ where $b_i$ is the center of the minimum enclosing ball of $P_i$. Then* **kmedianAlg**$(k, B)$ *is a $(1 + \alpha)$-approximation for computing $opt_{1,\infty}(k, \mathcal{P})$.*

**Proof.** Let $C = \{c_1, \dots, c_k\} \subset \mathbb{R}^d$ be the set of centers returned by **kmedianAlg**$(k, B)$, for which we have $kmedian(C, B) \leq \alpha \cdot \min_{C' \subseteq \mathbb{R}^d, |C'| = k} kmedian(C', B)$. Also let $C^* = opt_{1,\infty}(k, \mathcal{P})$. Then using triangle inequality,

$$
\begin{aligned}
cost_{1,\infty}(C,\mathcal{P}) &= \sum_{i=1}^{m} \min_{c\in C}\max_{p\in P_i}\|c-p\| \leq \sum_{i=1}^{m} \min_{c\in C}\max_{p\in P_i}(\|p-b_i\|+\|c-b_i\|) \\
&= \sum_{i=1}^{m}((\max_{p\in P_i}\|p-b_i\|)+(\min_{c\in C}\|c-b_i\|)) \\
&= \sum_{i=1}^{m}\max_{p\in P_i}\|p-b_i\| + \sum_{i=1}^{m}\min_{c\in C}\|c-b_i\| \\
&= \sum_{i=1}^{m}\max_{p\in P_i}\|p-b_i\| + kmedian(C,B) \\
&\leq optcost_{1,\infty}(k,\mathcal{P}) + kmedian(C,B) && \text{Lemma 10} \\
&\leq optcost_{1,\infty}(k,\mathcal{P}) + \alpha\cdot\min_{C'\subseteq\mathbb{R}^d,|C'|=k} kmedian(C',B) \\
&\leq optcost_{1,\infty}(k,\mathcal{P}) + \alpha\cdot kmedian(C^*,B) \\
&\leq optcost_{1,\infty}(k,\mathcal{P}) + \alpha\cdot cost_{1,\infty}(C^*,\mathcal{P}) && \text{Lemma 11} \\
&= optcost_{1,\infty}(k,\mathcal{P}) + \alpha\cdot optcost_{1,\infty}(k,\mathcal{P}) \\
&= (1+\alpha)optcost_{1,\infty}(k,\mathcal{P}) && \blacktriangleleft
\end{aligned}
$$

As discussed in the introduction, there are many known $(1+\varepsilon)$-approximation algorithms for $k$-median, with various trade-offs in the running times depending on the parameters $n, d, k$, and $\varepsilon$. Moreover, as $k$-median is a special case of our problem, we should not expect our times to be faster than what is possible for $k$-median.

Before we can apply such algorithms, the above theorem requires computing the minimum enclosing ball of each set $P_i$, for which we can use $(1+\varepsilon)$-approximate minimum enclosing ball center, which can be computed in $O(dn/\varepsilon+1/\varepsilon^5)$ time [2]. Doing so would introduce a $(1+\varepsilon)$ factor in Lemma 10, and thus using this together with a $(1+\varepsilon)$-approximation for $k$-median, the same analysis from the proof of Theorem 12, would yield a $(1+\varepsilon)^2$ approximation to computing $opt_{1,\infty}(k,\mathcal{P})$. By observing that $(1+\varepsilon/4)^2 \leq (1+\varepsilon)$ for any $0<\varepsilon\leq 1$ we immediately have the following.

▶ **Corollary 13.** *For a parameter $k$ and set $P\subset\mathbb{R}^d$ of $n$ points, let $T(n,k,d,\varepsilon)$ denote the running time of any algorithm which achieves a $(1+\varepsilon)$-approximation for $k$-median. Moreover, let $MEB(n,d,\varepsilon)$ denote the time to compute a $(1+\varepsilon)$-approximate minimum enclosing ball of $P$.*

*Given a parameter $k$ and collection of sets $\mathcal{P}=\{P_1,\ldots,P_m\}$ in $\mathbb{R}^d$, where $n=\sum_i|P_i|$, then for any $0<\varepsilon\leq 1$, there is a $(2+\varepsilon)$-approximation for computing $opt_{1,\infty}(k,\mathcal{P})$ with $O(MEB(n,d,\varepsilon/4)+T(n,k,d,\varepsilon/4))$ running time.*

## 6 Another Variant of k-center Sets Clustering

Above we gave improved approximation algorithms for the $cost_{\infty,\infty}(C,\mathcal{P})$ objective, previously considered by [20], and which naturally generalizes the $k$-center problem to sets. Here we consider an alternative generalization of $k$-center to sets clustering. Namely, given $\mathcal{P}=\{P_1,...,P_m\}$ and a parameter $k$, find the set $C\subset\mathbb{R}^d$ of $k$ centers that minimizes:

$$cost_{\infty,1}(\mathcal{P},C) = \max_{P\in\mathcal{P}} f_1(C,P) = \max_{P\in\mathcal{P}}\min_{c\in C} f_1(c,P) = \max_{P\in\mathcal{P}}\min_{c\in C}\sum_{p\in P}\|c-p\|.$$

This variant poses additional challenges, so for simplicity we will assume that the sets in $\mathcal{P}$ lie in $\mathbb{R}^2$, though a similar approach should extend to any constant dimension. Moreover, in this section we state our running times simply as being polynomial in $n$, as the precise constant in the exponent is large, as opposed to earlier sections which may be linear in $n$ (depending on the subroutines used).

▶ **Definition 14.** *In the* weighted $k$-center problem*, we are given a parameter $k$, a set of points $P = \{p_1, \ldots, p_n\} \subset \mathbb{R}^2$ and a set of non-negative weights $W = \{w_1, \ldots, w_n\}$, where $w_i$ is the weight associated with point $p_i$, and the goal is to find the set $C \subset \mathbb{R}^2$ of $k$ centers which minimizes the cost function: $wkcenter(C, P) = \max_{p_i \in P} \min_{c \in C} w_i \cdot \|p_i - c\|$.*

Observe that the weighted $k$-center objective can be viewed as a special instance of our $cost_{\infty,1}$ objective, as a point with weight $|P_i|$ models the case when all points in $P_i$ are collocated, for each $P_i \in \mathcal{P}$. Thus intuitively, to approximate $cost_{\infty,1}$ will require approximating $wkcenter$.

[19] considered the weighted $k$-center problem where the input is instead an edge-weighted graph $G = (V, E)$, where each vertex $v$ has an associated non-negative weight $w_v$. Specifically, let $d(u, v)$ denote the shortest path distance between $u$ and $v$ (with respect to edge weights, not vertex weights), then they seek the set $C \subseteq V$ of $k$ centers which minimizes: $wkcenter(C, V) = \max_{v \in V} \min_{c \in C} w_v \cdot d(v, c)$. For this problem [19] give a 2-approximation, and we now describe how this implies a 2-approximation for the case in plane.

▶ **Lemma 15.** *There exists a polynomial time 2-approximation algorithm for the weighted $k$-center problem in the plane. That is, given a parameter $k$, a set of points $P = \{p_1, \ldots, p_n\} \subset \mathbb{R}^2$, and a set of non-negative weights $W = \{w_1, \ldots, w_n\}$ where $w_i$ is the weight associated with point $p_i$, there is a polynomial time 2-approximation to $\min_{C \subset \mathbb{R}^2, |C|=k} wkcenter(C, P)$.*

**Proof.** As we allow centers to be located anywhere in the plane, we first describe how to find a polynomial sized set $\Gamma$ which contains the optimal set of centers as a subset.

For any two points $p_i, p_j \in P$, define their weighted bisector, $\beta(p_i, p_j)$, as the subset of points $q \in \mathbb{R}^2$ such that $w_i\|p_i - q\| = w_j\|p_j - q\|$. It is well known that $\beta(p_i, p_j)$ is a line when $w_i = w_j$, and otherwise is a circle (called the Apollonius circle) containing the heavier weight point. Consider the arrangement of all such weighted bisectors over all pairs of points in $P$. The vertices of the arrangement occur at intersections of bisectors, and the edges are circular arcs between vertices (or entire bisectors, when the bisector does not intersect any other bisector). Let $\Gamma \subset \mathbb{R}^2$ be the set containing all points in $P$, all vertices in the arrangement, and for every edge on a bisector $\beta(p_i, p_j)$, a point on that edge achieving the minimum value of $w_i\|p_i - q\| = w_j\|p_j - q\|$ (which may occur at a vertex of the arrangement). Observe that $|\Gamma| = O(n^4)$ as this is the complexity of the arrangement.

So let $C^*$ be an optimal set of centers, and consider some $c^* \in C^*$. We now argue that if $c^* \notin \Gamma$, then it can be moved to a point in $\Gamma$ in such a way that the weighted $k$-center objective does not increase. First, let $P'$ be the subset of points from $P$ which call $c^*$ their nearest center in $C^*$ and let $q = \arg\max_{p_i \in P'} w_i\|p_i - c^*\|$. Consider moving $c^*$ continuously from its initial location towards $q$ (i.e. along the segment $c^*q$). As we move along this segment the weighted distance to $q$ strictly decreases, and moreover until we cross a bisector $\beta(q, p_i)$ for some $p_i$, $q$ remains the point in $P'$ furthest from $c^*$ (i.e. the point determining $c^*$'s contribution to the overall objective). First suppose that $c^*$ is moved all the way to $q$ without crossing a bisector involving $q$. In this case $q$ remained the furthest point from $c^*$, and so the cost of assigning $P'$ to $c^*$ did not increase (implying in fact that $c^* = q$ initially). In the second case, we run into some bisector $\beta(q, p_i)$ for some $p_i$, at which point we stop

moving $c^*$ towards $q$ (to ensure $q$ remains the furthest). Now if $c^*$ is at a vertex of the arrangement of bisectors, then we are done as we included all vertices in $\Gamma$. Otherwise, $c^*$ is on the interior of an edge in the arrangement. Now the point achieving the minimum value of the weighted distance from $q$ (or equivalently $p_i$) to $c^*$ along this edge is in included in $\Gamma$ (and may in fact be a vertex), and so we can move $c^*$ to this point without increasing the cost of assigning $P'$ to $c^*$, as we do not cross any other bisector while moving along this edge.

Now we construct a weighted graph instance $G = (V, E)$ of weighted $k$-center as follows. First, set $V = \Gamma$ (recall $P \subseteq \Gamma$). Now for every pair of points in $V$ set its edge weight equal to the Euclidean distance between the corresponding set of points. Finally, for the vertex weights, set the weight of all vertices in $\Gamma \setminus P$ equal to 0 (i.e. they do not have to be clustered), and for each $p_i \in P$ its weight remains $w_i$ (i.e. the weight from the weighted $k$-center instance in the plane). Note that by construction, the cost of any solution $C \subseteq V$ to this graph based weighted $k$-center problem is equivalent to the cost of $C$ for the corresponding weighted $k$-center problem in the plane. Thus, as $V = \Gamma$ is guaranteed to contain an optimal solution to our instance in the plane, if we simply call the polynomial time 2-approximation algorithm from [19] on this graph instance, then the returned solution is also a 2-approximation for our instance in the plane. ◀

Our goal now is to use the above lemma to get a constant factor approximation to $opt_{\infty,1}(k, \mathcal{P})$. First we need the following lemma, which is the analogue of Lemma 10 from the prior section.

▶ **Lemma 16.** *Let $\mathcal{P} = \{P_1, ..., P_m\}$ be a collection of sets. Then, for any point-set $P_i \in \mathcal{P}$,*

$$\min_{\mu \in \mathbb{R}^d} \sum_{p \in P_i} \|\mu - p\| \leq optcost_{\infty,1}(k, \mathcal{P}).$$

**Proof.** Let $C^* = opt_{\infty,1}(k, \mathcal{P})$. Then for any point set $P_i \in \mathcal{P}$,

$$\min_{\mu \in \mathbb{R}^d} \sum_{p \in P_i} \|\mu - p\| \leq \min_{c^* \in C^*} \sum_{p \in P_i} \|c^* - p\| \leq \max_{P_i \in \mathcal{P}} \min_{c^* \in C^*} \sum_{p \in P_i} \|c^* - p\| = optcost_{\infty,1}(k, \mathcal{P}).$$ ◀

▶ **Theorem 17.** *For a parameter $k$, a set of points $P = \{p_1, \ldots, p_l\} \subset \mathbb{R}^2$, and weights $W = \{w_1, \ldots, w_l\}$ where $w_i \geq 0$ is the weight associated with point $p_i$, let* **weightedKCenter**$(k, P, W)$ *denote any algorithm which achieves an $\alpha_1$-approximation for the weighted $k$-center problem. Also, for a point set $P \subset \mathbb{R}^2$, let* **1medianAlg**$(P)$ *denote any algorithm that achieves an $\alpha_2$-approximation for the 1-median problem.*

*Given a parameter $k$ and collection of sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^2$, there is a polynomial time $(\alpha_1 + \alpha_2 + \alpha_1 \cdot \alpha_2)$-approximation for computing $opt_{\infty,1}(k, \mathcal{P})$.*

**Proof.** For any set $C \subset \mathbb{R}^2$ of $k$ centers, and any set of points $X = \{x_1, \ldots, x_m\}$, by the triangle inequality we have

$$\begin{aligned} cost_{\infty,1}(C, \mathcal{P}) &= \max_{P_i \in \mathcal{P}} \min_{c \in C} \sum_{p \in P_i} \|c - p\| \leq \max_{P_i \in \mathcal{P}} \min_{c \in C} \Big( \sum_{p \in P_i} \|x_i - p\| + \sum_{p \in P_i} \|c - x_i\| \Big) \\ &\leq \max_{P_i \in \mathcal{P}} \sum_{p \in P_i} \|x_i - p\| + \max_{P_i \in \mathcal{P}} \min_{c \in C} |P_i| \cdot \|c - x_i\| \end{aligned}$$

Now, let the optimal set of centers for our problem be $C^* = opt_{\infty,1}(k, \mathcal{P})$ and let $C'$ be the $\alpha_1$-approximation returned by **weightedKCenter**$(k, X, W)$ where $X = \{x_1, \ldots, x_m\}$ is the set of $\alpha_2$-approximate 1-medians from the $P_i$ sets, i.e. $x_i =$ **1medianAlg**$(P_i)$, and $W = \{|P_1|, \ldots, |P_m|\}$. Then,

$$
\begin{aligned}
&cost_{\infty,1}(C', \mathcal{P}) \\
&\quad \leq \max_{P_i \in \mathcal{P}} \sum_{p \in P_i} \|x_i - p\| + \max_{P_i \in \mathcal{P}} \min_{c \in C'} |P_i| \cdot \|c - x_i\| \\
&\quad \leq \max_{P_i \in \mathcal{P}} (\alpha_2 \cdot \min_{\mu \in \mathbb{R}^2} \sum_{p \in P_i} \|\mu - p\|) + \alpha_1 \cdot \min_{\bar{C} \subset \mathbb{R}^2, |\bar{C}| = k} (\max_{P_i \in \mathcal{P}} \min_{c \in \bar{C}} |P_i| \cdot \|c - x_i\|) \\
&\quad \leq \alpha_2 \cdot \max_{P_i \in \mathcal{P}} \min_{\mu \in \mathbb{R}^2} \sum_{p \in P_i} \|\mu - p\| + \alpha_1 \cdot \max_{P_i \in \mathcal{P}} \min_{c \in C^*} |P_i| \cdot \|c - x_i\| \\
&\quad \leq \alpha_2 \cdot costopt_{\infty,1}(k, \mathcal{P}) + \alpha_1 \cdot \max_{P_i \in \mathcal{P}} \min_{c \in C^*} \sum_{p \in P_i} (\|x_i - p\| + \|c - p\|) \qquad \text{Lemmma 16} \\
&\quad \leq \alpha_2 \cdot costopt_{\infty,1}(k, \mathcal{P}) + \alpha_1 \cdot (\max_{P_i \in \mathcal{P}} \sum_{p \in P_i} \|x_i - p\| + \max_{P_i \in \mathcal{P}} \min_{c \in C^*} \sum_{p \in P_i} \|c - p\|) \\
&\quad \leq \alpha_2 \cdot costopt_{\infty,1}(k, \mathcal{P}) + \alpha_1 \cdot (\max_{P_i \in \mathcal{P}} (\alpha_2 \cdot \min_{\mu \in \mathbb{R}^2} \sum_{p \in P_i} \|\mu - p\|) + costopt_{\infty,1}(k, \mathcal{P})) \\
&\quad \leq \alpha_2 \cdot costopt_{\infty,1}(k, \mathcal{P}) + \alpha_1 \cdot (\alpha_2 \cdot costopt_{\infty,1}(k, \mathcal{P}) + costopt_{\infty,1}(k, \mathcal{P})) \\
&\quad = (\alpha_1 + \alpha_2 + \alpha_1 \cdot \alpha_2) \cdot costopt_{\infty,1}(k, \mathcal{P}) \qquad \blacktriangleleft
\end{aligned}
$$

▶ **Corollary 18.** *Given a parameter $k$ and collection of sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^2$, there is a polynomial time $(5+\varepsilon)$-approximation for computing $opt_{\infty,1}(k, \mathcal{P})$, for any constant $\varepsilon > 0$.*

**Proof.** Using [11], we can get a polynomial time $(1+\varepsilon/3)$-approximation for the 1-median, for any constant $\varepsilon > 0$. Lemma 15 gives us a polynomial time 2-approximation for weighted $k$-center. So we have $\alpha_1 = 2$, $\alpha_2 = (1+\varepsilon/3)$, and $(\alpha_1+\alpha_2+\alpha_1\alpha_2) = (2+(1+\varepsilon/3)+2(1+\varepsilon/3)) = (5+\varepsilon)$. Thus, applying Theorem 17 with these algorithms as subroutines gives a polynomial time $(5+\varepsilon)$-approximation for $opt_{\infty,1}(k, \mathcal{P})$. ◀

## 6.1 Approximation Improvement

We now show how to improve the above $(5+\varepsilon)$-approximation into a $(3+\varepsilon)$-approximation, by constructing and searching with a $(3+\varepsilon)$-approximate decision procedure.

**Algorithm 2** **decider**$(\mathcal{P} = \{P_1, \ldots P_m\}, k, r)$.

```
Initialize all point sets P_1, P_2, ..., P_m as unmarked.
Compute M = {μ_1, ..., μ_m} where μ_i = 1medianAlg(P_i) is an α-approximate median
Initialize C = {}.
repeat
    Let P_i = arg max_{unmarked P∈P} |P|.
    Mark point-set P_j if Σ_{p∈P_j} ||p − μ_i|| ≤ r.
    Set C = C ∪ {μ_i}.
until all sets are marked;
if |C| > k then
    return "optcost_{∞,1}(k, P) > r/(2 + α)"
return C
```

▶ **Lemma 19.** *For a point set $P \subset \mathbb{R}^2$, let* **1medianAlg**$(P)$ *denote any algorithm that achieves an $\alpha$-approximation for the 1-median problem.*

*Given a collection of sets $\mathcal{P} = \{P_1, ..., P_m\}$ in $\mathbb{R}^2$, a parameter $k$, and a radius $r \in \mathbb{R}$, then* **decider**$(\mathcal{P}, k, r)$ *either returns a set $C \subset \mathbb{R}^2$ of $k$ centers such that $cost_{\infty,1}(C, \mathcal{P}) \leq r$, or correctly returns that $optcost_{\infty,1}(k, \mathcal{P}) > r/(2+\alpha)$.*

**Proof.** **decider**$(\mathcal{P}, k, r)$ starts by initializing all point sets in $\mathcal{P}$ as unmarked, and computing the approximate medians of each point set using **1medianAlg**. Next, the algorithm picks the approximate median of the unmarked point set which has the most points, adds it to the set $C$ of centers, and mark all the sets that are $\leq r$ from this approximate median. This process is repeated until all sets are marked. At the end of the loop, if $|C| \leq k$, then the algorithm returns $C$, as clearly we have obtained a valid clustering with cost $\leq r$.

Otherwise, $|C| \geq k+1$, in which case the algorithm returns "$optcost_{\infty,1}(k, \mathcal{P}) > r/(2+\alpha)$", which we now argue is correct. Let $\{c_1, ..., c_{k+1}\}$ be the first $k+1$ centers selected by algorithm, where $c_i \in M$ was selected in the $i$th iteration, and let $\mathcal{S} = \{S_1, S_2, ..., S_{k+1}\}$, where $S_i \in \mathcal{P}$ is the set such that $c_i =$ **1medianAlg**$(S_i)$. Let $C^* = opt_{\infty,1}(k, \mathcal{P})$. Then by the pigeonhole principle, there must be some $c^* \in C^*$ and sets $S_i, S_j \in \mathcal{S}$ with $i < j$ such that $c^* = \arg\min_{c \in C^*} \sum_{s \in S_i} \|c - s\| = \arg\min_{c \in C^*} \sum_{s \in S_j} \|c - s\|$. Note that as $i < j$, by line 5 of the algorithm we know $|S_i| \geq |S_j|$. Then,

$$\begin{aligned}
\sum_{s \in S_j} \|c_i - s\| &\leq \sum_{s \in S_j} (\|c_i - c^*\| + \|c^* - s\|) = \sum_{s \in S_j} \|c_i - c^*\| + \sum_{s \in S_j} \|c^* - s\| \\
&\leq |S_j| \cdot \|c_i - c^*\| + optcost_{\infty,1}(k, \mathcal{P}) \leq |S_i| \cdot \|c_i - c^*\| + optcost_{\infty,1}(k, \mathcal{P}) \\
&\leq \sum_{s \in S_i} (\|c_i - s\| + \|c^* - s\|) + optcost_{\infty,1}(k, \mathcal{P}) \\
&= \sum_{s \in S_i} \|c_i - s\| + \sum_{s \in S_i} \|c^* - s\| + optcost_{\infty,1}(k, \mathcal{P}) \\
&\leq \alpha \cdot \min_{c' \in \mathbb{R}^2} \sum_{s \in S_i} \|c' - s\| + optcost_{\infty,1}(k, \mathcal{P}) + optcost_{\infty,1}(k, \mathcal{P}) \\
&\leq \alpha \cdot optcost_{\infty,1}(k, \mathcal{P}) + 2 \cdot optcost_{\infty,1}(k, \mathcal{P}) \\
&= (2+\alpha) \cdot optcost_{\infty,1}(k, \mathcal{P})
\end{aligned}$$

Since $i < j$, we know that $c_i$ did not cover $S_j$ within radius $r$, that is $r < \sum_{s \in S_j} \|c_i - s\|$. Thus by the above, $r < \sum_{s \in S_j} \|c_i - s\| \leq (2+\alpha) \cdot optcost_{\infty,1}(k, \mathcal{P})$. Therefore, the algorithm correctly returns that $optcost_{\infty,1}(k, \mathcal{P}) > \frac{r}{2+\alpha}$. ◀

▶ **Theorem 20.** *Given a parameter $k$ and a collection of point sets $\mathcal{P} = \{P_1, \ldots, P_m\}$ in $\mathbb{R}^2$, for any constant $0 < \varepsilon \leq 1$, there exists a polynomial time $(3+\varepsilon)$-approximation algorithm for computing $opt_{\infty,1}(k, \mathcal{P})$.*

**Proof.** Let $C'$ be the $(5+\varepsilon)$-approximate set of centers returned by Corollary 18. Let $r = cost_{\infty,1}(C', \mathcal{P})$ and let $r^* = opt_{\infty,1}(k, \mathcal{P})$, where by Corollary 18 we know that $r^* \in [\frac{r}{5+\varepsilon}, r]$. Consider the candidate radii set $R = \{r_0, r_1, \ldots, r_z\}$, where $r_i = \frac{r}{5+\varepsilon}(1+\varepsilon/4)^i$ and $z = \lceil \log_{1+\varepsilon/4}(5+\varepsilon) \rceil = O(1)$ for any constant $\varepsilon > 0$. Note that for any $i$ we have $r_{i+1}/r_i = (1+\varepsilon/4)$, and $z$ was chosen such that $r_z \geq r$.

Now we iterate through the candidate radii $r_i \in R$, in order from $i = 0$ to $z$. In the $i$th iteration we call Algorithm 2 with radius $(2+\alpha) \cdot r_i$ where $\alpha$ is the approximation ratio of the subroutine used to compute the approximate 1-medians. We can use [11] to get an $\alpha = (1+\varepsilon/8)$-approximate 1-median, which runs in polynomial time for any

constant $\varepsilon > 0$. Let $j$ be the first iteration where Algorithm 2 returns a set $C$ rather than "$optcost_{\infty,1}(k,\mathcal{P}) > (2+\alpha)\cdot r_j/(2+\alpha)$". We claim that this set $C$ is the desired approximation.

First, observe that this procedure is guaranteed to return some set $C$, that is $j$ is well defined. Specifically, when $j = z$, the call to Algorithm 2 with radius $(2+\alpha)\cdot r_z$, must return a set $C$ since otherwise by Lemma 19 the optimal radius $r^* > \frac{(2+\alpha)\cdot r_z}{2+\alpha} = r_z \geq r$ which is a contradiction to the fact that $r^* \in [\frac{r}{5+\varepsilon}, r]$.

Lemma 19 guarantees that $cost_{\infty,1}(C,\mathcal{P}) \leq (2+\alpha)\cdot r_j$. If $j = 0$, then we are done as we know $(2+\alpha)\cdot r_j = (3+\varepsilon/8)\cdot r_0 = (3+\varepsilon/8)\cdot \frac{r}{5+\varepsilon} \leq (3+\varepsilon/8)\cdot r^* \leq (3+\varepsilon)\cdot r^*$. Otherwise, $j > 0$ in which case we know $r_{j-1}$ returned "$optcost_{\infty,1}(k,\mathcal{P}) > (2+\alpha)\cdot r_{j-1}/(2+\alpha)$", and so again by Lemma 19 we then know that $r^* > \frac{(2+\alpha)\cdot r_{j-1}}{2+\alpha} = r_{j-1}$. Summarizing, $r_{j-1} < r^* \leq cost_{\infty,1}(C,\mathcal{P}) \leq (2+\alpha)\cdot r_j$, and so the approximation quality of the solution $C$ is bounded by the ratio:

$$\frac{(2+\alpha)\cdot r_j}{r_{j-1}} = (2+\alpha)(1+\varepsilon/4) = (3+\varepsilon/8)(1+\varepsilon/4) = 3+(7/8)\varepsilon+\varepsilon^2/32 \leq 3+\varepsilon$$

Overall this is a polynomial time algorithm, as we made $O(1)$ calls to Algorithm 2, which itself is polynomial time when using a polynomial time subroutine to compute the approximate 1-medians. ◀

## References


1. Pankaj K. Agarwal and Cecilia Magdalena Procopiuc. Exact and approximation algorithms for clustering. *Algorithmica*, 33(2):201–226, 2002. doi:10.1007/s00453-001-0110-y.
2. Mihai Badoiu and Kenneth L. Clarkson. Smaller core-sets for balls. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 801–802. ACM/SIAM, 2003. URL: http://dl.acm.org/citation.cfm?id=644108.644240.
3. Moses Charikar and Rina Panigrahy. Clustering to minimize the sum of cluster diameters. *J. Comput. Syst. Sci.*, 68(2):417–441, 2004. doi:10.1016/J.JCSS.2003.07.014.
4. Ke Chen. On coresets for k-median and k-means clustering in metric and euclidean spaces and their applications. *SIAM J. Comput.*, 39(3):923–947, 2009. doi:10.1137/070699007.
5. Vincent Cohen-Addad, Karthik C. S., and Euiwoong Lee. Johnson coverage hypothesis: Inapproximability of k-means and k-median in *lp*-metrics. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1493–1530. SIAM, 2022. doi:10.1137/1.9781611977073.63.
6. Eduard Eiben, Fedor V. Fomin, Petr A. Golovach, William Lochet, Fahad Panolan, and Kirill Simonov. EPTAS for *k*-means clustering of affine subspaces. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2649–2659. SIAM, 2021. doi:10.1137/1.9781611976465.157.
7. Tomás Feder and Daniel H. Greene. Optimal algorithms for approximate clustering. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC)*, pages 434–444. ACM, 1988. doi:10.1145/62212.62255.
8. Dan Feldman, Morteza Monemizadeh, and Christian Sohler. A PTAS for k-means clustering based on weak coresets. In *Proceedings of the 23rd ACM Symposium on Computational Geometry (SOCG)*, pages 11–18. ACM, 2007. doi:10.1145/1247069.1247072.
9. Jie Gao, Michael Langberg, and Leonard J. Schulman. Clustering lines in high-dimensional space: Classification of incomplete data. *ACM Trans. Algorithms*, 7(1), December 2010. doi:10.1145/1868237.1868246.
10. T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985. doi:10.1016/0304-3975(85)90224-5.

11 Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In László Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC)*, pages 291–300. ACM, 2004. doi:10.1145/1007352.1007400.

12 D. S. Hochbaum and D. B. Shmoys. A best possible heuristic for the $k$-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985. doi:10.1287/moor.10.2.180.

13 W. Hsu and G. L. Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1(3):209–215, 1979. doi:10.1016/0166-218X(79)90044-1.

14 Hongyao Huang, Georgiy Klimenko, and Benjamin Raichel. Clustering with neighborhoods. In *32nd International Symposium on Algorithms and Computation (ISAAC)*, volume 212 of *LIPIcs*, pages 6:1–6:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ISAAC.2021.6.

15 Ibrahim Jubran, Murad Tukan, Alaa Maalouf, and Dan Feldman. Sets clustering. In *Proceedings of the 37th International Conference on Machine Learning, (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pages 4994–5005. PMLR, 2020. URL: http://proceedings.mlr.press/v119/jubran20a.html.

16 Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. A local search approximation algorithm for k-means clustering. *Comput. Geom.*, 28(2-3):89–112, 2004. doi:10.1016/J.COMGEO.2004.03.003.

17 Amit Kumar, Yogish Sabharwal, and Sandeep Sen. Linear-time approximation schemes for clustering problems in any dimensions. *J. ACM*, 57(2), February 2010. doi:10.1145/1667053.1667054.

18 Sagi Lotan, Ernesto Evgeniy Sanches Shayda, and Dan Feldman. Coreset for line-sets clustering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL: http://papers.nips.cc/paper_files/paper/2022/hash/f2ce95887c34393af4eb240d60017860-Abstract-Conference.html.

19 Qingzhou Wang and Kam-Hoi Cheng. A heuristic algorithm for the k-center problem with vertex weight. In *International Symposium on Algorithms (SIGAL)*, volume 450 of *Lecture Notes in Computer Science*, pages 388–396. Springer, 1990. doi:10.1007/3-540-52921-7_88.

20 Guang Xu and Jinhui Xu. Efficient approximation algorithms for clustering point-sets. *Computational Geometry*, 43(1):59–66, 2010. Special Issue on the 14th Annual Fall Workshop. doi:10.1016/j.comgeo.2007.12.002.