Low-Distortion Clustering with Ordinal and Limited Cardinal Information

Jakob Burkhardt[1], Ioannis Caragiannis[1], Karl Fehrs[1], Matteo Russo[2]
Chris Schwiegelshohn[1], Sudarshan Shyam[1]

[1]Department of Computer Science, Aarhus University, Åbogade 34, 8200 Aarhus N, Denmark,
[2]Department of Computer, Control, Management Engineering, Sapienza University of Rome, Via Ariosto 25, 00185, Rome, Italy
{jakob, iannis, karl, schwiegelshohn, shyam}@cs.au.dk, russo.matteo0801@gmail.com

Abstract

Motivated by recent work in computational social choice, we extend the metric distortion framework to clustering problems. Given a set of n agents located in an underlying metric space, our goal is to partition them into k clusters, optimizing some social cost objective. The metric space is defined by a distance function d between the agent locations. Information about d is available only implicitly via n rankings, through which each agent ranks all other agents in terms of their distance from her. Still, even though no cardinal information (i.e., the exact distance values) is available, we would like to evaluate clustering algorithms in terms of social cost objectives that are defined using d. This is done using the notion of distortion, which measures how far from optimality a clustering can be, taking into account all underlying metrics that are consistent with the ordinal information available.

Unfortunately, the most important clustering objectives (e.g., those used in the well-known k-median and k-center problems) do not admit algorithms with finite distortion. To sidestep this disappointing fact, we follow two alternative approaches: We first explore whether resource augmentation can be beneficial. We consider algorithms that use more than k clusters but compare their social cost to that of the optimal k-clusterings. We show that using exponentially (in terms of k) many clusters, we can get low (constant or logarithmic) distortion for the k-center and k-median objectives. Interestingly, such an exponential blowup is shown to be necessary. More importantly, we explore whether limited cardinal information can be used to obtain better results. Somewhat surprisingly, for k-median and k-center, we show that a number of queries that is polynomial in k and only logarithmic in n (i.e., only sublinear in the number of agents for the most relevant scenarios in practice) is enough to get constant distortion.

1 Introduction

The typical computational social choice problem consists of optimizing a function over alternatives, each with a different associated cost or value. A classic example is given by representative election. Each voter has a different representation score for every candidate, which we assume to correspond to the distance in some underlying metric. Ideally, the representation minimizes the sum of distances of each voter to their closest representative. In the full information setting, this corresponds to solving the classic k-median problem. But this example already illustrates the difficulty of implementing any voting mechanism: Even if the representation scores are assumed to be distances, they might be unknown even to the participating voters. However, we may readily know if a voter prefers alternative a over alternative b.

Such examples have given rise to *ordinal algorithms*. An ordinal algorithm mainly allows for comparisons between distances in the underlying metric. That is, given three points a, b, c, we are freely given information whether $d(a, b) \leq d(a, c)$, but we are not given the exact numerical values of $d(a, b)$ and $d(a, c)$. The objective is to solve a given problem relying primarily on the ordinal information, while using as few (ideally zero) distance queries as possible. The goodness of such an algorithm is measured in terms of the quality of the computed solution C compared to the quality of the optimal solution OPT that is given full information, commonly known as the *metric distortion*.

Finding the median is arguably the most important problem in this field. Given a set of points X and a distance function d, the median m is defined to be the point minimizing the sum of distances. Following a long line of work (Anshelevich et al. 2021; Anshelevich, Filos-Ratsikas, and Voudouris 2022; Feldman, Fiat, and Golomb 2016; Goel, Krishnaswamy, and Munagala 2017; Kempe 2020; Munagala and Wang 2019), there now exists a deterministic algorithm with optimal metric distortion 3 (Gkatzelis, Halpern, and Shah 2020), which is also optimal (Anshelevich, Bhardwaj, and Postl 2015; Anshelevich et al. 2018). Using randomization, Charikar et al. (2023) recently achieved an important breakthrough, achieving a metric distortion of 2.753. The best known lower bound is at least 2.1126 (Charikar and Ramakrishnan 2022).

Extensions to more general clustering objectives such as (k, z)-clustering and facility location are comparatively much harder, see Anshelevich and Zhu (2017); Caragiannis, Shah, and Voudouris (2022). In facility location, we ask for a set of centers C such that

$$\sum_{x \in X} \min_{c \in C} d(x, c) + f \cdot |C|$$

is minimized, where f is the cost of opening a center. For (k, z)-clustering, we instead consider the objective

$$\sqrt[z]{\sum_{x \in X} \min_{c \in C} d(x, c)^z},$$

i.e., the algorithm does not incur a cost for opening the centers, but instead has a budget of at most k centers that can be placed. Special cases include k-median where $z = 1$ and k-center which corresponds to $z \to \infty$.[1]

Unfortunately, there are strong impossibility results for purely ordinal algorithms. Even for 2-median, it is not possible to obtain an algorithm with bounded metric distortion (Anshelevich and Zhu 2017). Therefore, research has begun to design algorithms that are given more power than purely ordinal information. Indeed, there has been some recent success in providing guarantees using only a constant number of queries per point, see Amanatidis et al. (2022a,b). For clustering, recent work by Pulyassary (2022) has show that using at most $\text{polylog}(n)$ distance queries per point, or $n \cdot \text{polylog}(n)$ queries overall, it is possible to achieve a constant factor approximation. The same work also showed that k queries per point, or $O(nk)$ queries overall are sufficient to achieve a constant factor approximation for k-median. Thus, we ask:

Question 1.1. *What is the minimum number of queries necessary for an algorithm to achieve constant metric distortion for k-median, k-center, and facility location?*

While distance queries are a natural way of lending more power to the algorithm designer, obtaining the distances may be expensive as mentioned above. This leads to the question whether other models exist that allow the algorithm designer to bound the metric distortion. A very natural way of doing so for clustering algorithms is by allowing the algorithm to return a (α, β)-bicriteria approximation. Such algorithms bound the clustering cost by at most α times the cost of an optimal k clustering, while using β many centers. We ask:

Question 1.2. *What is the minimum value of β such that a bicriteria clustering algorithm using only ordinal information has constant metric distortion?*

Our Results

In this paper we make substantial progress towards answering both questions. In the low-query setting, we give two deterministic polynomial time algorithms for k-center that, using at most $O(k^2)$ overall distance evaluations, obtain a 2-distortion and, using at most $O(k)$ overall distance evaluations, obtain a 4-distortion. We also show that the latter result is optimal in terms of the number of necessary queries, while the former is optimal for any polynomial time algorithm. For (k, z)-clustering, we obtain a randomized polynomial time algorithm that uses at most $\text{poly}(k, \log n)$ overall distance queries and achieves constant metric distortion. Note that all of these bounds are sublinear in the input size, that is assuming $k \ll n$, we make $o(1)$ queries per point.

Finally, for facility location, there exists a simple adaptation of the seminal Meyerson algorithm (Meyerson 2001) that achieves a constant distortion using exactly one query per point or n queries overall, see also Section 4.1 of Pulyassary (2022). We show that no algorithm can achieve a constant factor approximation using less than $\Omega(n)$ queries, effectively closing the problem.

In the zero-query setting, we first show that there exists a $(2, 2^{k-1})$-bicriteria algorithm for k-center. Moreover, this algorithm is optimal in the sense that any algorithm achieving finite distortion must use $\Omega(2^k)$ centers. For (k, z)-clustering, we obtain two algorithms that solve all (k, z)-clustering objectives. The first succeeds with constant probability and achieves constant distortion with $(O(\log n)^{k-1+o(1)})$ many centers. The second requires $(O(\log n)^{k+o(1)})$ and achieves $O(1)$ distortion both in expectation and with high probability. We complement this result by showing that, for any constant factor distortion to k-median, $\Omega((2^{\log^* n})^{k-1} + 2^k \log n)$ centers are necessary even with a constant probability of success. For the special case of 2-median, our bounds are optimal.

Related Work

Ordinal Preferences and Distortion The first paper to consider optimization problems using ordinal information was probably Procaccia and Rosenschein (2006). Subsequently, two main directions have been established. Continuing to work with the model introduced by Procaccia and Rosenschein, one line focuses mainly on maximizing welfare subject to normalization assumptions, but without assuming any metric properties, see Amanatidis et al. (2021, 2022a,b); Caragiannis and Procaccia (2011); Filos-Ratsikas, Micha, and Voudouris (2020). The other line of work studies problem without the normalization assumptions, but assuming that the preferences are metric, i.e., they satisfy the triangle inequality. Beyond clustering papers covered in the introduction, several other distortion problems have been studied (Borodin et al. 2019; Cheng, Dughmi, and Kempe 2017, 2018; Pierczynski and Skowron 2019). While rare, it is also possible to achieve some results without making either a normalization or metric assumptions, see Abramowitz and Anshelevich (2018).

Clustering and Facility Location (k, z)-clustering is APX-hard in general metrics (Cohen-Addad, C. S., and Lee 2021), though it is possible to obtain very accurate algorithms when making assumptions on either the metric (Friggstad, Rezapour, and Salavatipour 2019; Cohen-Addad, Feldmann, and Saulpic 2021) or the input (Angelidakis, Makarychev, and Makarychev 2017; Awasthi, Blum, and Sheffet 2010; Cohen-Addad and Schwiegelshohn 2017). For k-center, Gonzalez (1985) gave an optimal 2-approximation algorithm. For k-median, k-means and facility location, following a long line of research (Jain and Vazirani 2001; Jain, Mahdian, and Saberi 2002; Arya et al. 2004; Li and Svensson 2016; Cohen-Addad et al. 2022, 2023), the current state of the art is a 2.613 approximation for k-median (Gowda et al. 2023), a 9 approximation for k-means (Ahmadian et al. 2020), and a 1.488 approximation for facility location (Li 2013). For general (k, z)-clustering, there are few claimed bounds, though most of the proofs for k-median and k-means go through while losing a $\exp(z)$ approximation factor. Explicit results can be found in Cohen-Addad, Klein, and Mathieu (2019); Cohen-Addad, Saulpic, and Schwiegelshohn (2021).

[1]Sometimes the $\sqrt[z]{}$ operation is omitted, as is the case for k-means corresponds to $(k, 2)$-clustering. An α-approximation to $\sqrt[z]{\sum_{x \in X} \min_{c \in C} d(x, c)^z}$ implies an $O(\alpha^z)$-approximation to $\sum_{x \in X} \min_{c \in C} d(x, c)^z$.

2 Preliminaries

Let (X, d) be a metric space where X is a set of n points and $d : X \times X \to \mathbb{R}_{\geq 0}$ is a metric. The distance between any two points $x, y \in X$ can be accessed by a *query* of the form $d(x, y)$. We assume that such a query is associated with a cost. An algorithm is given a budget and each query that the algorithm makes consumes one unit of its budget. While querying the exact distance between two points is costly, our model assumes that, for every point, *ordinal* information about its relative distance to the other points is freely available. More specifically, each point $x \in X$ provides a *ranking* $\pi_x : [n] \to X$ that is *consistent* with d in the sense that $d(x, \pi_x(i)) \leq d(x, \pi_x(j))$ for every $i, j \in [n], i < j$. That is, points that are closer to x appear *higher* in x's ranking. An ordinal *preference profile* P is then just the collection of the points' rankings, i.e., $P = \{\pi_x\}_{x \in X}$. We write $\mathcal{P}(d)$ for the set of profiles where each point's ranking is consistent with the distances d.

It is often convenient to restrict the ranking of a point to a certain subset of X. Let $S \subseteq X$ and $m = |S|$. The *restriction of* π_x *to* S is a function $\pi_{x,S} : [m] \to S$ such that, for any two $y, y' \in S$, y is ranked higher in $\pi_{x,S}$ than y' if and only if y is ranked higher in π_x than y'.

The ordinal preference profile provides a very rough sketch of the underlying distance metric d. However, the relative distances expressed by the profile can enable an algorithm to allocate its budget in a very economic way. Consider the following operation: For a set of points $S \subseteq X$ and a point $x \in X$, we define the *distance of* x *to* S to be $d(x, S) = \min_{y \in S} d(x, y)$.

Given the ordinal information, the point $z = \arg\min_{y \in S} d(x, y)$ can readily be identified as x's highest ranked point among S. Hence, an algorithm can determine the distance of x to S with a single query $d(x, z)$. Clearly, the same observation can be made about finding $z = \arg\max_{y \in S} d(x, y)$ and the distance $d(x, z)$.

We intend to study the loss in outcome optimality if we restrict an algorithm $\mathcal{A}$ to the ordinal information and a fixed query budget. We consider a variety of clustering problems where the goal is to find a solution that minimizes a given cost function ϕ. We denote by $\mathcal{M}$ the set of all metric spaces. For a metric space $(X, d) \in \mathcal{M}$ and a profile $P \in \mathcal{P}(d)$, let $\mathcal{A}(P, d)$ be the solution (set of centers) computed by algorithm $\mathcal{A}$, and let $C^*(d)$ be a solution (set of centers) of minimal cost. We say that an algorithm $\mathcal{A}$ achieves *distortion* D with constant (respectively high) probability, if

$$\sup_{\substack{(X,d) \in \mathcal{M} \\ P \in \mathcal{P}(d)}} \frac{\phi(\mathcal{A}(P, d))}{\phi(C^*(d))} \leq D$$

with probability at least $2/3$ (respectively probability at least $1 - 1/n$). The *expected distortion* of $\mathcal{A}$ is given by the ratio

$$\sup_{\substack{(X,d) \in \mathcal{M} \\ P \in \mathcal{P}(d)}} \frac{\mathbb{E}[\phi(\mathcal{A}(P, d))]}{\phi(C^*(d))}.$$

We now state the definition of the (k, z)-clustering problem in the ordinal setting and introduce a few standard terms that are commonly used in the context of clustering problems.

Definition 2.1. *In the* ordinal (k, z)-clustering problem*, we are given positive integers* k, z *and a set* X *of* n *points that form a metric space* (X, d) *under distances* d*. Each point* $x \in X$ *reports a ranking* π_x *that is consistent with the distances* d*. Let* $P = \{\pi_x\}_{x \in X}$*. For a subset* $S \subseteq X$ *of the points, we denote the cost of a given solution* $C \subseteq X$ *by*

$$\phi_C(S, d) = \sqrt[z]{\sum_{x \in S} d(x, C)^z}.$$

The goal is to find a set C *of* k *points such that the cost function* $\phi_C(X, d)$ *is minimized. For compactness, we drop the dependence on* d *and denote by* $\phi_{OPT}(S)$ *the cost of the optimal solution on an arbitrary set of points* $S \subseteq X$.

Given a solution C to an ordinal (k, z)-clustering instance, we typically call the elements of C *centers*. C naturally induces a partition of X into k *clusters* $\{A_c\}_{c \in C}$ where, for each $c \in C$, $A_c = \{x \in X : \pi_{x,C}(1) = c\}$. We refer to the collection of these clusters as a *clustering* of X.

Finally, we define sampling probabilities for all (k, z)-clustering objectives.

Definition 2.2. *Let* z *be a positive integer, and let* $C \subseteq X$ *be a set of centers. The* sampling *probability of point* $c \in X$ *conditioned on having already selected a set of centers* C *is*

$$p_z(c) := \mathbb{P}[c \text{ is added to } C \mid C] = \frac{d(c, C)^z}{\sum_{x \in X} d(x, C)^z},$$

and denote the induced distribution by D_z^{++}.

3 Algorithms for k-Center

We present three algorithms for solving the ordinal k-center ((k, ∞)-clustering) problem. Our algorithms are based on a greedy procedure by Gonzalez (1985), which is known to yield a 2-approximation of the k-center problem. This procedure simply chooses an arbitrary center to begin with and then, in $k-1$ iterations, chooses the center that is farthest away from the already chosen centers (farthest-first traversal).

2-Distortion Algorithm The farthest-first traversal method lends itself well to be adapted to the ordinal setting. Clearly, given a set of clusters, the farthest point from these clusters can be determined with one distance query per cluster. For completeness, we give a pseudocode implementation of the procedure in the full version of our paper. This immediately gives rise to the following result.

Theorem 3.1. *There exists a deterministic* 2*-distortion algorithm for* k*-center that makes* $\frac{k^2-k}{2}$ *distance queries.*

For the zero-query regime, we extend the farthest-first traversal method such that, in every iteration, the farthest point in *every* cluster is chosen. Since the algorithm and its analysis are straightforward adaptions of Gonzalez (1985), we merely state the result and give the details in the full version.

Theorem 3.2. *There exists a deterministic algorithm that, using only ordinal preferences, returns a set of centers* C *of size* $|C| = 2^{k-1}$*, such that* $\max_{x \in X} d(x, C) \leq 2\phi_{OPT}$*, where* ϕ_{OPT} *is the cost of an optimal* k*-center clustering.*

4-Distortion Algorithm with $O(k)$ Queries

To achieve a constant distortion via a linear (in k) number of queries, the idea is to perform a $\frac{1}{2}$-approximate farthest-first traversal. Such a farthest-first traversal is robust with respect to the distortion, losing only a factor of 2. Surprisingly, using ordinal information, we can execute a $\frac{1}{2}$-approximate farthest-first-traversal with an optimal query bound. At a very high level, we keep track of (center,farthest point) pairs for all clusters throughout the algorithm. However, we do not query all the pairs. Instead we keep a track of an independent set of pairs to query which helps us bound the number of new pairs created, while ensuring that the distance of the unqueried pairs are at most twice the queried distances. We give the complete analysis here and the pseudocode in the full version.

Theorem 3.3. *There exists a deterministic 4-distortion algorithm to the optimal k-center clustering that makes $2k$ queries.*

Throughout the algorithm's run, let C be the solution set and let $Q \subseteq C$ be the so-called query set. Both C and Q will change over time, so we denote C_i as the solution and Q_i as the query set after the i-th iteration, for clarity of exposition. Moreover, for $y \in C_i$, let $S_{y,i}$ be the set of points such that, for each of these points, y is the closest center among C_i, and let $z_i = \arg\max_{x \in S_{y,i}} d(y, x)$. Note that we query the distance $d(y, z_i)$, if y belongs to the query set Q_i.

In iteration $i \in \{0\} \cup [k-1]$ of the algorithm, we perform the following steps:

1. Select the cluster $S_{y,i}$, for $y \in Q_i$ such that $d(y, z_i)$ is maximized and add z_i to C_i, forming C_{i+1}.
2. Remove y from Q_i and let $R_{i+1} := C_{i+1} \setminus Q_i$ (i.e. R_{i+1} always consists at least of y and z_i).
3. Add centers from R_{i+1} to Q_i to obtain Q_{i+1} as follows: Let $u \in R_{i+1}$.
   - If there exists a center $p \in Q_i$ such that $d(p, q) \geq d(w, q)$, where $w = \arg\max_{x \in S_{u,i+1}} d(u, x)$ and $q = \arg\max_{x \in S_{p,i+1}} d(p, x)$, do not add u to Q_i.
   - If no such p exists, add u to Q_i.

   Once all u's have been discarded, we have obtained our new set Q_{i+1}. All distances between centers in Q_{i+1} and the respective furthest points are queried. Note that we only have to query novel pairs, i.e. already queried pairs do not require a new query.

We now prove several claims about the algorithm. The first two bound the number of queries. The final two claims yield the desired bound on the distortion: In particular, we show that we select, at each iteration, a point that is no closer than half the distance of the furthest point and that such an approximate farthest-first traversal also yields a constant distortion to the optimal k-center solution.

Invariant 3.4. *If $y \in Q_i$ and $z_i = \arg\max_{x \in S_{y,i}} d(y, x) \notin C_{i+1}$, then $\arg\max_{x \in S_{y,i}} d(y, x) = \arg\max_{x \in S_{y,i+1}} d(y, x)$.*

Proof. We prove this by induction, the base case of which is trivial as initially we only have an arbitrary center and its most distant point in S_0 and Q_0.

Let $\{w\} = C_{i+1} \setminus C_i$ and let u be the center of the cluster containing w in C_i. Consider any $y \in Q_i$. If y was added to Q_i before u, then we know $d(z_i, w) > d(y, z_i)$, hence $z_i = z_{i+1}$. If y was added to Q_i after u, then $d(z_i, w) > d(u, w)$. But since $d(y, z_i) \leq d(u, w)$, we have $d(y, z_i) < d(z_i, w)$ which also implies $z_i = z_{i+1}$. □

Lemma 3.5. *The total number of queries is at most $2k$.*

Proof. By Invariant 3.4, the only way a point can be removed from Q_i is if it was added to C_{i+1}. Therefore, the number of queries made that lead to a deletion are exactly k. The remaining number of queries are upper bounded by at most k, and the claim follows. □

This shows that the total number of queries made by the algorithm are bounded by $O(k)$. We now turn to the distortion factor. The following lemma shows that the algorithm executes a $\frac{1}{2}$-farthest first traversal.

Lemma 3.6. *Let $\{z\} = C_{i+1} \setminus C_i$ and let $z \in S_{y,i}$. Then for any $u \in C_i$ and $w \in S_{u,i}$, we have $d(y, z) \geq \frac{1}{2} \cdot d(u, w)$.*

Proof. We selected $\arg\max_{y \in Q_i} d(y, z_i)$. Hence it suffices to compare $d(y, z_i)$ with $d(u, w)$ for $u \notin Q_i$. Since $u \notin Q_i$, we know that there exists some $y' \in Q_i$ s.t. $d(z'_i, w) \leq d(y', z'_i) \leq d(y, z_i)$. By the triangle inequality. $d(y', z'_i) \geq d(y', w) - d(z'_i, w) \geq d(u, w) - d(z'_i, w)$. Rearranging, we have

$$d(u, w) \leq d(y', z'_i) + d(z'_i, w) \leq 2d(y', z'_i) \leq 2d(y, z_i),$$

which concludes the proof. □

Finally, we show that an approximate farthest-first traversal yields a constant distortion to the optimal k-center solution.

Lemma 3.7. *Suppose we iteratively select points such that, in every iteration, $d(z, C_i) \geq \alpha \cdot \arg\max_{x \in X} d(x, C_i)$, for $\alpha \in (0, 1]$. Then, C_{k-1} yields a $\frac{2}{\alpha}$- distortion to the optimal k-center clustering:*

$$\max_{x \in X} \min_{u \in C_{k-1}} d(x, u) \leq \frac{2}{\alpha} \cdot \phi_{OPT},$$

where ϕ_{OPT} is the cost of an optimal k-center clustering.

Proof. Let $\mathcal{C}^* = \{A_1, \ldots A_k\}$ be the optimal clustering. If $C_{k-1} \cap A_j$ is non-empty, for all $A_j \in \mathcal{C}^*$, the distortion is 2 due to the triangle inequality. Otherwise, we let i be the first iteration where we added a second point x_2 from some cluster A_j to A_i and let x_1 be the first point from A_j added to C. Then for any u

$$d(u, C_i) \leq \frac{1}{\alpha} \cdot d(x_2, C_i) \leq \frac{1}{\alpha} \cdot d(x_2, x_1) \leq \frac{2}{\alpha} \cdot \phi_{\text{OPT}},$$

which concludes the proof. □

Combining Lemmas 3.6, 3.7, and 3.5 then yields the theorem. Achieving a *strictly* smaller than 4-distortion with a *strictly* subquadratic number of queries (or proving that it is impossible) is an interesting open problem.

4 Algorithms for (k, z)-Clustering

In this section, we present our algorithms for solving the (k, z)-clustering problems. The first makes use of no queries and obtains a bi-criteria distortion guarantee, seeking to trade off distortion with the number of selected centers.

Zero-Query Bi-Criteria Algorithm

The algorithm is based on distance sampling. The seminal k-means++ by Arthur and Vassilvitskii (2007) iteratively selects points proportionate to the squared Euclidean distance of the current set of centers. In this paper, we consider a generalization to (k, z)-clustering, where we sample points proportionate to their cost. In both cases, the expected cost of the computed solution is with a factor of $O(\log k)$ of that of an optimal k-means clustering[2] and this bound is tight even in the Euclidean plane. Improvements to this basic algorithm are abundant in literature. Indeed, $2k$ rounds are already enough to achieve a $O(1)$ bicriteria approximation, see Makarychev, Reddy, and Shan (2020) and Wei (2016). Alternatively, one may sample multiple points in each round. This tends to yield a worse tradeoff between samples and cost, but combined with other algorithms, may yield a constant approximation (Bahmani et al. 2012; Choo et al. 2020; Lattanzi and Sohler 2019; Rozhon 2020; Grunau et al. 2023).

When adapting this procedure to the ordinal setting, the first challenge to overcome is that we do not know pairwise distances. The key idea behind our algorithm is to use ordinal information to approximate the sampling probabilities. We do this by over-sampling, i.e., we pick $O(\log n)$ points for each point that the (k, z)++ algorithm picks. The main results of this section are the following two:

Theorem 4.1. *For the $(2, z)$ clustering instance, Algorithm 1 returns $O(\log n)$ centers achieving a $O(1)$ distortion with constant probability.*

In the full version of the paper, we show that this is optimal.

Theorem 4.2. *There exists a randomized algorithm achieving a $O(1)$-distortion for all (k, z) clustering objectives* simultaneously *using $O(\log n)^{k+o(1)}$ centers, both on expectation and with high probability.*

Now, we present the algorithm for Theorem 4.1 and provide some intuition as to why it works. A similar reasoning can be extended to obtain the algorithm for Theorem 4.2. Theorem 4.1 and Theorem 4.2 are formally proven in the full version of the paper. The algorithm for Theorem 4.2 involves repeating Algorithm 1 to amplify success probability, and augmenting Theorem 3.3's k-center algorithm to bound the worst case.

For the algorithm, we define some new notation. For any set of points $S \in X$ and any point $c \notin S$, we define a partition of S_c into disjoint sets $\{S_{c,0}, S_{c,1}, \ldots, S_{c,\ell}\}$ where $\ell = \lfloor \log |S| \rfloor$. We construct the partition recursively starting from $S_{c,\ell}$. Define $S_{c,\ell}$ to be the singleton set containing just the farthest point in S from c. Next, for each $1 < j < \ell$, define $S_{c,j}$ to be the farthest $2^{\ell-j}$ points from the set $S \backslash \{S_{c,j+1} \cup S_{c,j+2} \cdots \cup S_{c,\ell}\}$. Lastly, let $S_{c,1} = S \backslash \{S_{c,2} \cup S_{c,3} \cdots \cup S_{c,\ell}\}$.

[2]The distribution has been analyzed repeatedly for the k-means problem. Similar statements for (k, z) clustering are folklore, and we provide complete proofs for these problems in the full version.

Algorithm 1: (k, z)-clustering without queries

Input: Point set X, ordinal information $P = \{\pi_p\}_{p \in A}$ and $k \in \mathbb{N}$

1 Initialize the set of centers $C = \emptyset$
2 Sample a point c uniformly at random from X
3 Let $C = \{c\}$
4 **for** $i = 2$ *to* $k - 1$ **do**
5   Initialize $C_i \leftarrow \emptyset$ **for** *each point c in C* **do**
6   Define $S = \{x \in X : \pi_x(c) \leq \pi_x(c') \; \forall c' \in C\}$, i.e., S is the set of points that belong to the cluster with center c, and let $\ell = \lfloor \log |S| \rfloor$
7   Sample $7 \log k$ points uniformly randomly from each of the sets $\{S_{c,1}, S_{c,2}, \ldots, S_{c,\ell}\}$ (defined above) and add them to C_i
      $C \leftarrow C \cup C_i$
8 **return** C

Analysis: We now highlight a key property of Algorithm 1 that shows us why it gives us a $O(1)$ distortion for the 2-median instance with constant probability. The following lemma show that, Algorithm 1, in a sense, performs better than the (k, z)++ algorithm in each iteration. Formally, for each point $c \in X$, we show that the probability that Algorithm 1 picks the point in an iteration is at least the probability that the (k, z)++ algorithm picks the point.

Lemma 4.3. *Let C be the set of centers before at the beginning of line 3 of Algorithm 1 in the i^{th} iteration. For any point $c \in X$ after line 8 of Algorithm 1, we have*

$$\mathbb{P}_{\text{Alg 1}}[c \in C_i | C] \geq p_z(c).$$

The proof of the lemma is deferred to the full version. Though Lemma 4.3 gives us an idea as to why Algorithm 1 indeed does well, it is important to note that statement, by itself, does not imply the bounds in Theorem 4.1 and Theorem 4.2. Specifically, Lemma 4.3 does not imply that we perform better than the (k, z)++ algorithm. The reason is that Algorithm 1 samples points from different rings independently as opposed to the (k, z)++ algorithm. The analysis in Makarychev, Reddy, and Shan (2020) and Bhattacharya et al. (2020) points at the fickle nature of k-means++ algorithm and how slightly perturbing it leads to a worse performance. To get around this, we use over-sampling without making the asymptotic bicriteria approximation worse.

$O(1)$-Distortion Algorithm with $O(k^4 \log^5 n)$ Queries

We design an algorithm that achieves a constant distortion to the cardinal objective with just a few cardinal queries. Formally, we show the following result:

Theorem 4.4. *There exists a randomized algorithm achieving an expected $O(1)$-distortion to the optimal (k, z)-clustering using $O(k^4 \log^5 n)$ queries.*

Our exposition mainly focuses on the k-median objective, for which $z = 1$, however, the proofs almost seamlessly go through for other (k, z) clustering objectives. Due to space constraints, we give a full proof and pseudocode for the algorithm in the full version of the paper and only highlight the key ideas here. To this end, given a current set of centers C, we define an *estimated* cost for each of the rings in question, i.e.,

$$\widehat{\phi_C}(S_{i,j}) = |S_{i,j}| \cdot \min_{x \in S_{i,j-1}} d(x, c_i).$$

Note that to compute the above-estimated cost, we just need one query per ring (in each round). Indeed, we simply need to query the distance between point c_i and the topmost point in c_i's preference list that belongs to $S_{i,j-1}$. Since there are T rounds, the resulting number of queries is $\sum_{t\in[T]} t \cdot \log(|X|) \leq T^2 \log n$. Now, we *emulate* the k-median++ algorithm by sampling a center c belonging to ring S_{rj} with probability equal to

$$\widehat{p}(c) := \frac{1}{|S_{rj}|} \cdot \frac{\widehat{\phi_C}(S_{rj})}{\sum_{i,j} \widehat{\phi_C}(S_{i,j})}.$$

It is not hard to see that the above is non-negative and summing across all i, j we obtain 1, thereby making the above a valid distribution, which, from now on, we will call D. Before discussing the main algorithm in its full details, let us recall that, in the plain k-median++ algorithm, given a current set of centers C, each new center c is sampled (adaptively) with probability

$$p(c) := \frac{d(c, C)}{\sum_{x\in X} d(x, C)}.$$

This probability is proportional to how much they contribute to the current overall cost. Recall that we name the k-median++ induced distribution as D^{++} (since $z = 1$ in this case). The following lemma relates the standard k-median++ distribution D^{++} and the emulating distribution D.

Lemma 4.5. *Let C be the set of centers already chosen. For any point $c \in X$ sampled according to distribution D*

$$\mathbb{P}_D[c \in C_i|C] \geq \frac{1}{2} \cdot p_z(c).$$

Algorithm. From this point onwards, our goal will be to show that Algorithm 6 (whose formal description is deferred to the full version of the paper) achieves an $O(1)$ distortion, as long as the number of rounds T is large enough. The high level idea is not to use a potential function that allows us to bound the cost of hit and not hit clusters, as is done in most k-means++ analyses. Instead, we show that the cost decreases by a constant factor for a sufficient number of samples, similar to Rozhon (2020).

Unfortunately, unlike these works, we cannot guarantee an upper bound on the cost when running a sampling algorithm with the guarantee provided by Claim 4.5. Indeed, there is a non-zero probability that we hit the same clusters over and over again, which can lead to an arbitrarily high distortion.

We sidestep these issues with a careful initialization. For this we use the k-center solution resulting from Section 3. A sufficiently good k-center solution is within a factor $O(n)$ of the cost of a (k, z) clustering. Moreover, our k-center algorithms are deterministic, which modifies our previous low probability event of having unbounded distortion to a low probability event of having $O(n)$ distortion.

Analysis. The proof proceeds as follows: First, let us consider the current set of centers C (initialized to C_0, the k-center clustering output by the algorithm used to prove Theorem 3.3). Then, we consider the optimal clustering collection $\mathcal{C}^* = \{A_1, \ldots, A_k\}$, and the union of uncovered clusters U, i.e., clusters not hit by C. We show that the probability that a given optimal cluster remains uncovered after a fresh center is sampled is inversely exponentially related to its cost (normalized by the total cost). This is crucial because it helps us in showing that the cost of uncovered points has to drop by at least a constant factor at each new iteration of the algorithm, which is the second step of our proof strategy. Lastly, we recall that the initial clustering was a constant distortion to the optimal k-center one, which means an $O(n)$-distortion to the optimal k-median clustering. This, combined with the earlier considerations, leads to a constant distortion provided $T \in O(k \log n)$.

5 Lower Bounds

In this section, we finally present our lower bounds. The lower bounds for k-center are simple and optimal, and thus given here. The lower bounds for k-median are significantly more complicated, but use a similar construction as the k-center lower bound. We conclude this section by presenting a lower bound for the facility location problem. The details for the latter two results are deferred to the full version of the paper.

Theorem 5.1. *For any fixed α, every bicriteria algorithm $\mathcal{A}$ for k-center that has distortion at most α with at least constant probability must return a solution of size at least $\Omega(2^k)$. Moreover, any algorithm that has distortion at most α with at least constant probability must make at least $\Omega(k)$ queries.*

We remark that the distortion bound α has no influence on the number of queries or the number of centers. That is, our lower bounds hold for arbitrary values of α. This property together with the observation that the cost of all (k, z)-clustering objectives are within a poly(n) factor implies that the same bounds indeed hold for any (k, z)-clustering.

Proof. The hard instance is the same for the low query and zero query setting. We start with an analysis for the latter.

The hard instance: Our hard instance consists of 2^{k-1} points. We begin by describing the ordinal information and the underlying metric. Consider a complete binary tree T of depth $k - 1$. For any two nodes p, q, we say that a is the common ancestor of p and q if a is the minimum depth node in the shortest path between p and q in T.

The interpretation of this tree is that the leaves are the points and for any interior node a, the value $d(a)$ stored in a denotes the distances between all points p, q that have a as the common ancestor. Thus, we now require the following invariant to ensure that the tree encodes a metric.

Invariant 5.2. *If the subtree rooted at a contains the interior node b, then $d(a) \geq d(b)$.*

We now specify the ordinal preferences, which we fix *before* determining the values $d(a)$ of the interior nodes. Let p, q, o be three leaves and let $a(p,q)$, $a(p,o)$, and $a(q,o)$ be common ancestors of these pairs of nodes, respectively.

- If the depth of $a(p,q)$ is larger than the depth of $a(p,o)$ and $a(q,o)$ then the preference list of p determines q to be closer to p than to o.
- If the depth of $a(p,q)$ and $a(p,o)$ is equal then the relative ordering of q and o in the preference list of p is arbitrary (w.l.o.g., it may be chosen lexicographically).

We now describe a hard input distribution that satisfies the invariant and is consistent with the ordinal preferences. Select a random path Q between the root of T and an arbitrary node r at depth $k-1$. All nodes a along that path receive the value $d(a) = D$. All remaining nodes receive the value $d(a) = 1$.

Analysis: Note that, for any two trees sampled from the distribution, the values assigned to the interior nodes satisfy Invariant 5.2 and thereby induce a metric on the set of leaf nodes. Since the ordinal preferences are independent from these values, the two trees cannot be distinguished using the ordinal information.

We now determine an optimal k-center solution C. For every interior node a in Q, the children of a form subtrees $T(a, small)$ and $T(a, large)$. The root b of $T(a, small)$ satisfies $d(b) = 1$ and the root c of $T(a, large)$ satisfies $d(c) = D$. For the largest depth interior node a in Q, we introduce the convention that $T(a, large)$ contains the leaf r (i.e., the end point of Q). C now places exactly one center on an arbitrary leaf of $T(a, small)$ and one center on r. The cost of C is therefore 1. Now consider any other solution C'. If C' does not place a center on r, then the cost of C' is D. Otherwise, there must exist some $a \in Q$ for which $T(a, small)$ does not receive a center. Hence, the points in $T(a, small)$ must be served by some center contained in $T(a, large)$ or by a point not contained in the subtree rooted at a. In both cases, the cost of these points is D.

To conclude, it now suffices to analyze the performance of the best deterministic algorithm placing K centers against this hard input distribution. Since the algorithm does not make any queries and cannot determine Q based on the ordinal information, its choice of centers is fixed. There are 2^{k-1} many different nodes at depth $k-1$. Hence, the probability that K includes the leaf node r is $K/2^{k-1}$. Conversely, if $K \notin \Omega(2^k)$ then the probability that K does not include r is at least constant, which leads to a distortion of D.

Finally, we remark on some generalizations of this lower bound. For the low query regime, an algorithm needs to find the entire path Q or, equivalently, identify the leaf r. If it decides to not do so then with probability at least $\frac{1}{2}$ it will have unbounded distortion of D. Again, consider the performance of the best deterministic algorithm against the input distribution. Given that T is a binary tree, at least one query queries is required to reduce the search space for Q (equivalently, for r) by a factor of $\frac{1}{2}$ in expectation. Hence, if the algorithm does not make $\Omega(k)$ queries, its distortion is unbounded. □

Next, we give a different lower bound for any bicriteria algorithm for k-median. Specifically, we show that any bicriteria algorithm for k-median requires $\Omega(2^k \log n)$ centers. For 2-median, this becomes $\Omega(\log n)$, which stands in contrast with 2-center, where we can obtain a true 2-distortion using only ordinal information (see Theorem 3.2). We also show that there exists a slow growing function $g(n)$ increasing in n for every fixed k such that any bicriteria algorithm requires $g(n)^k$ many queries. Moreover, $g(n)$ may be lower bounded by $2^{\log^* n}$, though somewhat higher bounds are likely possible using our construction. We conjecture that the true lower bound is $(\log n)^k$.

Theorem 5.3. *For any fixed α, every bicriteria algorithm $\mathcal{A}$ for k-median that has distortion less than α with at least constant probability must return a solution of size at least $\Omega\left(\frac{\log n}{\log \alpha} \cdot 2^k\right)$. Moreover, any algorithm achieving a constant factor approximation for k-median must make at least $\Omega(k + \log\log n)$ queries.*

Theorem 5.4. *For any fixed α and every fixed k, every bicriteria algorithm $\mathcal{A}$ for k-median that has distortion less than α with at least constant probability must return a solution of size at least $\Omega\left(\left(2^{\log^* n}\right)^{k-1}\right)$. The number of queries to achieve a constant distortion is at least $\Omega(k \cdot 2^{\log^* n})$.*

Finally, we return to the facility location problem. We are interested in lower-bounding the number of queries necessary to achieve any given distortion. Using no queries, it is not possible to obtain bounds on the distortion (Anshelevich and Zhu 2021) beyond the trivial $O(n)$ bound. Our lower bound essentially shows that $\Omega(n)$ queries are necessary to achieve constant distortion, making the adaptation of Meyerson's algorithm optimal.

Theorem 5.5. *For any fixed α, every algorithm $\mathcal{A}$ for facility location that has distortion less than α with at least constant probability must make $\Omega\left(\frac{n}{\alpha}\right)$ distance queries.*

6 Conclusion and Open Problems

We gave optimal algorithm for computing bicriteria approximations for k center both in terms of the number of distance queries as well as number of additional centers in the purely ordinal setting. Additionally, we gave optimal lower bounds for facility location and substantially improved low query and purely ordinal bicriteria algorithms for k-median.

Aside from closing the small remaining gaps left in our analysis, several interesting open problems present themselves. First, our bicriteria algorithm simultaenously achieves small distortion for all (k, z) clustering. Another popular way to interpolate between k-median and k-center is ordered clustering Byrka, Sornat, and Spoerhase (2018); Chakrabarty and Swamy (2018). Is it possible to achieve low distortion algorithms for this problem as well?

Furthermore, there exist many other clustering objectives, such as graph clustering. Which distortion/query tradeoffs are possible for sparsest cut and metric max cut?

7 Acknowledgements

Ioannis Caragiannis and Sudarshan Shyam were partially supported by the Independent Research Fund Denmark (DFF) under grant 2032-00185B. Matteo Russo is supported by the ERC Advanced Grant 788893 AMDROMA "Algorithmic and Mechanism Design Research in Online Markets". Jakob Burkhardt and Chris Schwiegelshohn are partially supported by the Independent Research Fund Denmark (DFF) under a Sapere Aude Research Leader grant No 1051-00106B.

References

Abramowitz, B.; and Anshelevich, E. 2018. Utilitarians Without Utilities: Maximizing Social Welfare for Graph Problems Using Only Ordinal Preferences. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, 894–901.

Ahmadian, S.; Norouzi-Fard, A.; Svensson, O.; and Ward, J. 2020. Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms. *SIAM Journal on Computing*, 49(4).

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2021. Peeking Behind the Ordinal Curtain: Improving Distortion via Cardinal Queries. *Artificial Intelligence*, 296: 103488.

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2022a. Don't Roll the Dice, Ask Twice: The Two-Query Distortion of Matching Problems and Beyond. In *Proceedings of the 36th Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2022b. A Few Queries Go a Long Way: Information-Distortion Tradeoffs in Matching. *Journal of Artificial Intelligence Research*, 74: 227–261.

Angelidakis, H.; Makarychev, K.; and Makarychev, Y. 2017. Algorithms for Stable and Perturbation-Resilient Problems. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing (STOC)*, 438–451.

Anshelevich, E.; Bhardwaj, O.; Elkind, E.; Postl, J.; and Skowron, P. 2018. Approximating Optimal Social Choice under Metric Preferences. *Artificial Intelligence*, 264: 27–51.

Anshelevich, E.; Bhardwaj, O.; and Postl, J. 2015. Approximating Optimal Social Choice under Metric Preferences. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, 777–783.

Anshelevich, E.; Filos-Ratsikas, A.; Shah, N.; and Voudouris, A. A. 2021. Distortion in Social Choice Problems: The First 15 Years and Beyond. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 4294–4301.

Anshelevich, E.; Filos-Ratsikas, A.; and Voudouris, A. A. 2022. The Distortion of Distributed Metric Social Choice. *Artificial Intelligence*, 308: 103713.

Anshelevich, E.; and Zhu, W. 2017. Tradeoffs Between Information and Ordinal Approximation for Bipartite Matching. In *Proceedings of the 10th International Symposium on Algorithmic Game Theory (SAGT)*, 267–279.

Anshelevich, E.; and Zhu, W. 2021. Ordinal Approximation for Social Choice, Matching, and Facility Location Problems Given Candidate Positions. *ACM Transactions on Economics and Computation*, 9(2): art. 9.

Arthur, D.; and Vassilvitskii, S. 2007. k-means++: The Advantages of Careful Seeding. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1027–1035.

Arya, V.; Garg, N.; Khandekar, R.; Meyerson, A.; Munagala, K.; and Pandit, V. 2004. Local Search Heuristics for k-Median and Facility Location Problems. *SIAM Journal on Computing*, 33(3): 544–562.

Awasthi, P.; Blum, A.; and Sheffet, O. 2010. Stability Yields a PTAS for k-Median and k-Means Clustering. In *Proceedings of the 51th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 309–318.

Bahmani, B.; Moseley, B.; Vattani, A.; Kumar, R.; and Vassilvitskii, S. 2012. Scalable K-Means++. *Proceeding of VLDB Endowment*, 5(7): 622–633.

Bhattacharya, A.; Eube, J.; Röglin, H.; and Schmidt, M. 2020. Noisy, Greedy and Not so Greedy k-Means++. In *Proceedings of the 28th Annual European Symposium on Algorithms (ESA)*, 18:1–18:21.

Borodin, A.; Lev, O.; Shah, N.; and Strangway, T. 2019. Primarily about Primaries. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, 1804–1811.

Byrka, J.; Sornat, K.; and Spoerhase, J. 2018. Constant-Factor Approximation for Ordered k-Median. In *Proceedings of the 50th Annual ACM Symposium on Theory of Computing (STOC)*, 620–631.

Caragiannis, I.; and Procaccia, A. D. 2011. Voting Almost Maximizes Social Welfare Despite Limited Communication. *Artificial Intelligence*, 175(9-10): 1655–1671.

Caragiannis, I.; Shah, N.; and Voudouris, A. A. 2022. The Metric Distortion of Multiwinner Voting. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, 4900–4907.

Chakrabarty, D.; and Swamy, C. 2018. Interpolating between k-Median and k-Center: Approximation Algorithms for Ordered k-Median. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*, 29:1–29:14.

Charikar, M.; and Ramakrishnan, P. 2022. Metric Distortion Bounds for Randomized Social Choice. In *Proceedings of the 33rd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2986–3004.

Charikar, M.; Ramakrishnan, P.; Wang, K.; and Wu, H. 2023. Breaking the Metric Voting Distortion Barrier. *CoRR*, abs/2306.17838.

Cheng, Y.; Dughmi, S.; and Kempe, D. 2017. Of the People: Voting Is More Effective with Representative Candidates. In *Proceedings of the 18th ACM Conference on Economics and Computation (EC)*, 305–322.

Cheng, Y.; Dughmi, S.; and Kempe, D. 2018. On the Distortion of Voting With Multiple Representative Candidates. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, 973–980.

Choo, D.; Grunau, C.; Portmann, J.; and Rozhon, V. 2020. k-Means++: Few More Steps Yield Constant Approximation. In *Proceedings of the 37th International Conference on Machine Learning, (ICML)*, 1909–1917.

Cohen-Addad, V.; C. S., K.; and Lee, E. 2021. On Approximability of Clustering Problems Without Candidate Centers. In *Proceedings of the 32nd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2635–2648.

Cohen-Addad, V.; Feldmann, A. E.; and Saulpic, D. 2021. Near-linear Time Approximation Schemes for Clustering in Doubling Metrics. *Journal of the ACM*, 68(6): 44:1–44:34.

Cohen-Addad, V.; Grandoni, F.; Lee, E.; and Schwiegelshohn, C. 2023. Breaching the 2 LMP Approximation Barrier for Facility Location with Applications to k-Median. In *Proceedings of the 34th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 940–986.

Cohen-Addad, V.; Gupta, A.; Hu, L.; Oh, H.; and Saulpic, D. 2022. An Improved Local Search Algorithm for k-Median. In *Proceedings of the 33rd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1556–1612.

Cohen-Addad, V.; Klein, P. N.; and Mathieu, C. 2019. Local Search Yields Approximation Schemes for k-Means and k-Median in Euclidean and Minor-Free Metrics. *SIAM Journal on Computing*, 48(2): 644–667.

Cohen-Addad, V.; Saulpic, D.; and Schwiegelshohn, C. 2021. Improved Coresets and Sublinear Algorithms for Power Means in Euclidean Spaces. In *Proceedings of the 34th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 21085–21098.

Cohen-Addad, V.; and Schwiegelshohn, C. 2017. On the Local Structure of Stable Clustering Instances. In *Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, 49–60.

Feldman, M.; Fiat, A.; and Golomb, I. 2016. On Voting and Facility Location. In *Proceedings of the 17th ACM Conference on Economics and Computation (EC)*, 269–286.

Filos-Ratsikas, A.; Micha, E.; and Voudouris, A. A. 2020. The Distortion of Distributed Voting. *Artificial Intelligence*, 286: 103343.

Friggstad, Z.; Rezapour, M.; and Salavatipour, M. R. 2019. Local Search Yields a PTAS for k-Means in Doubling Metrics. *SIAM Journal on Computing*, 48(2): 452–480.

Gkatzelis, V.; Halpern, D.; and Shah, N. 2020. Resolving the Optimal Metric Distortion Conjecture. In *Proceedings of the 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 1427–1438.

Goel, A.; Krishnaswamy, A. K.; and Munagala, K. 2017. Metric Distortion of Social Choice Rules: Lower Bounds and Fairness Properties. In *Proceedings of the 18th ACM Conference on Economics and Computation (EC)*, 287–304.

Gonzalez, T. F. 1985. Clustering to Minimize the Maximum Intercluster Distance. *Theoretical Computer Science*, 38: 293–306.

Gowda, K. N.; Pensyl, T. W.; Srinivasan, A.; and Trinh, K. 2023. Improved Bi-point Rounding Algorithms and a Golden Barrier for k-Median. In *Proceedings of the 34th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 987–1011.

Grunau, C.; Özüdogru, A. A.; Rozhon, V.; and Tetek, J. 2023. A Nearly Tight Analysis of Greedy k-means++. In *Proceedings of the 34th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1012–1070.

Jain, K.; Mahdian, M.; and Saberi, A. 2002. A New Greedy Approach for Facility Location Problems. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing (STOC)*, 731–740.

Jain, K.; and Vazirani, V. V. 2001. Approximation Algorithms for Metric Facility Location and k-Median Problems using the Primal-Dual Schema and Lagrangian Relaxation. *Journal of the ACM*, 48(2): 274–296.

Kempe, D. 2020. An Analysis Framework for Metric Voting based on LP Duality. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2079–2086.

Lattanzi, S.; and Sohler, C. 2019. A Better k-means++ Algorithm via Local Search. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 3662–3671.

Li, S. 2013. A 1.488 Approximation Algorithm for the Uncapacitated Facility Location Problem. *Information and Computation*, 222: 45–58.

Li, S.; and Svensson, O. 2016. Approximating k-Median via Pseudo-Approximation. *SIAM Journal on Computing*, 45(2): 530–547.

Makarychev, K.; Reddy, A.; and Shan, L. 2020. Improved Guarantees for k-means++ and k-means++ Parallel. In *Proceedings of the 33rd Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Meyerson, A. 2001. Online Facility Location. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science (FOCS)*, 426–431.

Munagala, K.; and Wang, K. 2019. Improved Metric Distortion for Deterministic Social Choice Rules. In *Proceedings of the 20th ACM Conference on Economics and Computation (EC)*, 245–262.

Pierczynski, G.; and Skowron, P. 2019. Approval-Based Elections and Distortion of Voting Rules. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, 543–549.

Procaccia, A. D.; and Rosenschein, J. S. 2006. The Distortion of Cardinal Preferences in Voting. In *Proceedings of the 10th International Workshop on Cooperative Information Agents (CIA)*, 317–331.

Pulyassary, H. 2022. *Algorithm Design for Ordinal Settings*. Master's thesis, University of Waterloo.

Rozhon, V. 2020. Simple and Sharp Analysis of k-means$||$. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 8266–8275.

Wei, D. 2016. A Constant-Factor Bi-Criteria Approximation Guarantee for k-means++. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NeurIPS)*, 604–612.