Proportional Fairness in Non-Centroid Clustering

Ioannis Caragiannis
Aarhus University
iannis@cs.au.dk

Evi Micha
Harvard University
emicha@seas.harvard.edu

Nisarg Shah
University of Toronto
nisarg@cs.toronto.edu

Abstract

We revisit the recently developed framework of proportionally fair clustering, where the goal is to provide group fairness guarantees that become stronger for groups of data points (agents) that are large and cohesive. Prior work applies this framework to centroid clustering, where the loss of an agent is its distance to the centroid assigned to its cluster. We expand the framework to *non-centroid clustering*, where the loss of an agent is a function of the other agents in its cluster, by adapting two proportional fairness criteria — the core and its relaxation, fully justified representation (FJR) — to this setting.

We show that the core can be approximated only under structured loss functions, and even then, the best approximation we are able to establish, using an adaptation of the GreedyCapture algorithm developed for centroid clustering [1, 2], is unappealing for a natural loss function. In contrast, we design a new (inefficient) algorithm, GreedyCohesiveClustering, which achieves the relaxation FJR exactly under arbitrary loss functions, and show that the efficient GreedyCapture algorithm achieves a constant approximation of FJR. We also design an efficient auditing algorithm, which estimates the FJR approximation of any given clustering solution up to a constant factor. Our experiments on real data suggest that traditional clustering algorithms are highly unfair, whereas GreedyCapture is considerably fairer and incurs only a modest loss in common clustering objectives.

1 Introduction

Clustering is a fundamental task in unsupervised learning, where the goal is to partition a set of n points into k clusters $C = (C_1, \ldots, C_k)$ in such a way that points within the same cluster are close to each other (measured by a distance function d) and points in different clusters are far from each other. This goal is materialized through a variety of objective functions, the most popular of which is the k-means objective: $\sum_{i=1}^{k} \frac{1}{|C_i|} \cdot \sum_{x,y \in C_i} d(x,y)^2$.

When the points are in a Euclidean space, the k-means objective can be rewritten as $\sum_{i=1}^{k} \sum_{x \in C_i} d(x, \mu_i)^2$, where $\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$ is the mean (also called the *centroid*) of cluster C_i.[1] This gives rise to centroid clustering, where deciding where to place the k cluster centers is viewed as the task and the clusters are implicitly formed when each point is assigned to its nearest cluster center.

[1] Centroids can be defined in non-Euclidean spaces, e.g., as $\mu_i = \arg\min_y \frac{1}{|C_i|} \sum_{x \in C_i} d(x,y)^2$.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-0605

In the literature on fairness in centroid clustering, the loss of each data point (hereinafter, agent) is defined as the distance to the nearest cluster center [1]; here, the cluster centers do not merely help rewrite the objective, but play an essential role. This is a reasonable model for applications such as facility location, where the loss of an agent indeed depends on how much they have to travel to get to the nearest facility.

But in other applications of clustering, we simply partition the agents and agents prefer to be close to other agents in their cluster—there are no "cluster centers" that they prefer to be close to. For example, in clustered federated learning [3], the goal is to cluster the agents and have agents in each cluster collaboratively learn a model; naturally, agents would want other agents in their cluster to have similar data distributions, so the model learned is accurate on their own data distribution.[2] Other examples where we want to cluster nearby points together without defining cluster centers include document clustering [6], image segmentation for biomedical applications [7], and social network segmentation [8].

While there exist plenty of clustering objectives which do not require defining cluster centers (such as the first formulation of the k-means objective above), in order to reason about fairness we need to define the loss of each agent under a non-centroid clustering and explore the tradeoff between the losses of different agents. We initiate the study of proportional fairness in non-centroid clustering.

We follow the idea of proportional fairness outlined in a recent line of work [1, 2, 9, 10], which ensures that no group of at least n/k agents should "improve" (formalized later) by forming a cluster of its own.[3] Our main research questions are:

> *Can we obtain compelling proportional fairness guarantees for non-centroid clustering as with centroid clustering? Do the algorithms known to work well for centroid clustering also work well for non-centroid clustering? Can we audit the proportional fairness of a given algorithm?*

1.1 Our Contributions

In non-centroid clustering, we are given a set N of n points (agents) and the desired number of clusters k. The goal is to partition the agents into (at most) k clusters $C = (C_1, \ldots, C_k)$. Each agent i has a loss function ℓ_i, and her loss under clustering C is $\ell_i(C(i))$, where $C(i)$ denotes the cluster containing her. We study both the general case where the loss functions of the agents can be arbitrary, and structured cases where the loss of an agent for a cluster is the average or maximum of her distances — according to a given distance metric — to the agents in the cluster. In the latter case, our theoretical results hold for general metric spaces, as they rely solely on the satisfaction of the triangle inequality.

We study two proportional fairness guarantees, formally defined in Section 2: *the core* [11] and its relaxation, *fully justified representation* (FJR) [12]. Both have been studied for centroid clustering [1, 2, 10], but we are the first to study them in non-centroid clustering.

A summary of a selection of our results is presented in Table 1, with the cell values indicating approximation ratios (lower is better, 1 is optimal).

Loss Functions	Core UB	Core LB	FJR
Arbitrary	∞		1
Average	$O(n/k)$ (polytime)	1.3	1 (4 in polytime)
Maximum	2 (polytime)	1	1 (2 in polytime)

Table 1: The feasible core and FJR approximation guarantees, both existentially and in polynomial time. In each case, we can obtain a better FJR approximation than the core approximation.

[2] Prior work formulates this as centroid clustering [4, 5], where the principal also chooses a model for each cluster, but this goes against the federated learning setting.

[3] Groups with fewer than n/k agents are not deemed to be entitled to form a cluster.

Our results show the promise of FJR: while it is a slight relaxation of the core, it is satisfiable even under arbitrary loss functions, whereas the core can be unsatisfiable even under more structured loss functions. The existential result for FJR is achieved using a simple (but inefficient) algorithm we design, GreedyCohesiveClustering, which is an adaptation of the Greedy Cohesive Rule from social choice theory [12]. The core approximations as well as efficient FJR approximations are achieved using an efficient version of it, which turns out to be an adaptation of the GreedyCapture algorithm that has been introduced for centroid clustering [1, 2]. We show that the FJR approximation achieved by GreedyCapture stems from the fact that its key subroutine achieves a constant approximation of that of the GreedyCohesiveClustering algorithm.

Next, we turn to auditing the FJR approximation of a given clustering. Surprisingly, we show that the same technique that we use to algorithmically *achieve a constant approximation of FJR* can be used to also *estimate the FJR approximation of any given clustering*, up to a constant factor (4 for the average loss and 2 for the maximum loss).

We compare GreedyCapture to popular clustering methods, k-means++ and k-medoids, on three real datasets. We observe that in terms of both average and maximum loss, GreedyCapture provides significantly better approximations to both FJR and the core, and this fairness advantage comes at only a modest cost in terms of traditional clustering objectives, including those that k-means++ and k-medoids are designed to optimize.

1.2 Related Work

In recent years, there has been an active line of research related to fairness in clustering [13]. With a few exceptions, most of the work focuses on centroid-based clustering, where each agent cares about their distance from the closest cluster center. Mostly related to ours is the work by Chen et al. [1], who introduced the idea of proportionality through the core in centroid clustering. Their work has been revisited by Micha and Shah [2] for specific metric spaces. More recently, Aziz et al. [10] also introduced the relaxation of the core, fully justified representation, in centroid-based clustering. While one of our main algorithms, GreedyCapture, is a natural adaptation of the main algorithm used in all these works, there are significant differences between the two settings.

First, in centroid-based clustering, GreedyCapture provides a constant approximation to the core[1], while in the non-centroid case this approximation is not better than $O(n/k)$ for the average loss function. Second, in centroid-based clustering, GreedyCapture returns a solution that satisfies FJR exactly[10]. Here, for the non-centroid case, even though we know that an exact FJR solution always exists, GreedyCapture is shown to just provide an approximation better than 4 for the average loss and 2 for the maximum loss. In more specific metric spaces, Micha and Shah [2] show that a solution in the core always exists in the line. Here, we demonstrate that while this remains true for the maximum loss, it is not the case for the average loss, where the core can be empty. Finally, Chen et al. [1] conducted experiments using real data in which k-means++ performs better than GreedyCapture. However, for the same datasets, we found that GreedyCapture significantly outperforms k-means++ in the non-centroid setting.

Fairness in non-centroid clustering has received significantly less attention. Ahmadi et al. [14] recently introduced a notion of individual stability which indicates that no agent should prefer another cluster over the one they have been assigned to. Micha and Shah [2] studied the core when the goal is to create a balanced clustering (i.e. all clusters have almost equal size) and the agents have positive utilities for other agents. More generally, the hedonic games literature (e.g., see [15] for an early survey on the topic and [16] for a recent model that is close to the current paper) is also relevant to non-centroid clustering as it examines coalition formation. While the core concept has been extensively studied in hedonic games, there are two main differences with our work. First, subsets of any size can deviate to form their own cluster, rather than only proportionally eligible ones, and second, no approximate guarantees to the core have been provided, to the best of our knowledge.

https://doi.org/10.52202/079017-0605

2 Model

For $t \in \mathbb{N}$, let $[t] \triangleq \{1, \ldots, t\}$. We are given a set N of n agents, and the desired number of clusters k. Each agent $i \in N$ has an associated *loss function* $\ell_i : 2^N \setminus 2^{N \setminus \{i\}} \to \mathbb{R}_{\geqslant 0}$, where $\ell_i(S)$ is the cost to agent i for being part of group S. A k-clustering[4] $C = (C_1, \ldots, C_k)$ is a partition of N into k clusters,[5] where $C_t \cap C_{t'} = \emptyset$ for $t \neq t'$ and $\cup_{t=1}^{k} C_t = N$. With slight abuse of notation, denote by $C(i)$ the cluster that contains agent i. Then, the loss of agent i under this clustering is $\ell_i(C(i))$.

Loss functions. We study three classes of loss functions; for each class, we seek fairness guarantees that hold for any loss functions the agents may have from that class. A distance metric over N is given by $d : N \times N \to \mathbb{R}_{\geqslant 0}$, which satisfies: (i) $d(i, i) = 0$ for all $i \in N$, (ii) $d(i, j) = d(j, i)$ for all $i, j \in N$, and (iii) $d(i, j) \leqslant d(i, k) + d(k, j)$ for all $i, j, k \in N$ (triangle inequality).

- *Arbitrary losses.* In this most general class, the loss $\ell_i(S)$ can be an arbitrary non-negative number for each agent $i \in N$ and cluster $S \ni i$.
- *Average loss.* Here, we are given a distance metric d over N, and $\ell_i(S) = \frac{1}{|S|} \sum_{j \in S} d(i, j)$ for each agent $i \in N$ and cluster $S \ni i$. Informally, agent i prefers the agents in her cluster to be close to her on average.
- *Maximum loss.* Again, we are given a distance metric d over N, and $\ell_i(S) = \max_{j \in S} d(i, j)$ for each agent $i \in N$ and cluster $S \ni i$. Informally, agent i prefers that no agent in her cluster to be too far from her.

3 Core

Perhaps the most widely recognized proportional fairness guarantee is *the core*. Informally, an outcome is in the core if no group of agents $S \subseteq N$ can choose another (partial) outcome that (i) they are entitled to choose based on their proportion of the whole population ($|S|/|N|$), and (ii) makes every member of group S happier. The core was proposed and widely studied in the resource allocation literature from microeconomics [11, 17, 18], and it has been adapted recently to centroid clustering [1, 2]. When forming k clusters out of n agents, a group of agents S is deemed worthy of forming a cluster of its own if and only if $|S| \geqslant n/k$. In centroid clustering, such a group can choose any location for its cluster center. In the following adaptation to non-centroid clustering, no such consideration is required.

Definition 1 (α-Core)**.** For $\alpha \geqslant 1$, a k-clustering $C = (C_1, \ldots, C_k)$ is said to be in the α-core if there is no group of agents $S \subseteq N$ with $|S| \geqslant n/k$ such that $\alpha \cdot \ell_i(S) < \ell_i(C(i))$ for all $i \in S$. We refer to the 1-core simply as the core.

Given a clustering C, if there exists a group S that demonstrates a violation of the α-core guarantee, i.e., S has size at least n/k and the loss of each $i \in S$ for S is lower than $1/\alpha$ of her own loss under C, we say that S *deviates* under C and refer to it as the *deviating coalition*. We begin by proving a simple result that no finite approximation of the core can be guaranteed for arbitrary losses.

Theorem 1. *For arbitrary losses, there exists an instance in which no α-core clustering exists for any finite α.*

Next, for the more structured average loss function, we prove that the core can still be empty, albeit there is now room for a finite approximation. The proof, with an intricate construction, is delegated to Appendix A.

Theorem 2. *For the average loss, there exists an instance in which no α-core clustering exists for $\alpha < \frac{1+\sqrt{3}}{2} \approx 1.366$.*

To complement Theorems 1 and 2, we show the existence of a clustering in the $O(n/k)$-core (resp., 2-core) for the average (resp., maximum) loss. Despite significant effort, we are unable

[4] We simply call it clustering when the value of k is clear from the context.

[5] Technically, we have up to k clusters as C_t is allowed to be empty for any $t \in [k]$.

ALGORITHM 1: GreedyCohesiveClustering($\mathcal{A}$)

Input: Set of agents N, metric d, number of clusters k
Output: k-clustering $C = (C_1, \dots C_k)$
$N' \leftarrow N$; // Remaining set of agents
$j \leftarrow 1$; // Current cluster number
while $N' \neq \emptyset$ **do**
  $C_j \leftarrow \mathcal{A}(N', d, \lceil n/k \rceil)$; // Find and remove the next cohesive cluster
  $N' \leftarrow N' \setminus C_j$;
  $j \leftarrow j + 1$;
end
$C_j, C_{j+1}, \dots, C_k \leftarrow \emptyset$;
return $C = (C_1, \dots, C_k)$;

ALGORITHM 2: SmallestAgentBall

Input: Subset of agents $N' \subseteq N$, metric d, threshold integer τ
Output: Cluster S
if $|N'| \leqslant \tau$ **then return** $S \leftarrow N'$;
for $i \in N'$ **do**
  $\ell_i \leftarrow \tau$-th closest agent in N' to agent i; // Ties are broken arbitrarily
  $r_i \leftarrow d(i, \ell_i)$; // Smallest ball centered at agent i capturing at least τ agents
end
$i^* \leftarrow \arg\min_{i \in N'} r_i$;
return $S \leftarrow$ the set of τ closest agents in N' to agent i^*;

to determine whether the core is always non-empty for the maximum loss, or whether a constant approximation of the core can be guaranteed for the average loss, which we leave as tantalizing open questions.

Open Question 1: For the maximum loss, does there always exist a clustering in the core?

Open Question 2: For the average loss, does there always exist a clustering in the α-core for some constant α?

Our algorithms. For the positive result, we design a simple greedy algorithm, GreedyCohesiveClustering (Algorithm 1). It uses a subroutine $\mathcal{A}$, which, given a subset of agents $N' \subseteq N$, metric d, and threshold τ, finds a "cohesive" cluster S. Here, the term "cohesive" is informally used, but we will see a formalization in the next section. The threshold τ is meant to indicate the smallest size at which a group of agents deserve to form a cluster, but $\mathcal{A}$ can return a cluster of size greater, equal, or less than τ.

The algorithm we use as $\mathcal{A}$ in this section is given as SmallestAgentBall (Algorithm 2). It finds the smallest ball centered at agent that captures at least τ agents, and returns a set of τ agents from this ball. We call this algorithm with the natural choice of $\tau = \lceil n/k \rceil$, so GreedyCohesiveClustering(SmallestAgentBall) iteratively finds the smallest agent-centered ball containing $\lceil n/k \rceil$ agents and removes $\lceil n/k \rceil$ in that ball, until fewer than $\lceil n/k \rceil$ agents remain, at which point all remaining agents are put into one cluster and any remaining clusters are left empty.

Overall, GreedyCohesiveClustering(SmallestAgentBall) is an adaptation of the GreedyCapture algorithm proposed by Chen et al. [1] for centroid clustering with two key differences in our non-centroid case: (i) while they grow balls centered at feasible cluster center locations, we grow balls centered at the agents, and (ii) while they continue to grow a ball that already captured $\lceil n/k \rceil$ agents (and any agents captured by this ball in the future are placed in the same cluster), we stop a ball as soon as it captures $\lceil n/k \rceil$ agents, which

https://doi.org/10.52202/079017-0605

is necessary in our non-centroid case.[6] Nonetheless, due to its significant resemblance, we refer to the particular instantiation GreedyCohesiveClustering(SmallestAgentBall) as GreedyCapture hereinafter. The following result is one of our main results, with an intricate proof found in Appendix A.

Theorem 3. *For the average (resp., maximum) loss, the GreedyCapture algorithm is guaranteed to return a clustering in the* $(2 \cdot \lceil n/k \rceil - 3)$*-core (resp.,* 2*-core) in* $O(kn)$ *time complexity, and these bounds are (almost) tight.*

In many applications of clustering, such as clustered federated learning, the average loss is realistic because the agent's loss depends on all the agents in her cluster, and not just on a single most distant agent. Hence, it is a little disappointing that the only approximation to the core that we are able to establish in this case is $\alpha = O(n/k)$, which is rather unsatisfactory. We demonstrate two ways to circumvent this negative result. First, we consider demanding that any deviating coalitions be of size at least $\delta \cdot n/k$ for some $\delta > 1$. In Appendix B, we show that any *constant* $\delta > 1$ reduces the approximation factor α to a constant. In the next section, we explore a different approach: we relax the core to a slightly weaker fairness guarantee, which we show can be satisfied exactly, even under arbitrary losses.

4 Fully Justified Representation

Peters et al. [12] introduced *fully justified representation* (FJR) as a relaxation of the core in the context of approval-based committee selection. The following definition is its adaptation to non-centroid clustering. Informally, for a deviating coalition S, the core demands that the loss $\ell_i(S)$ of each member i after deviation be lower than *her own loss before deviation*, i.e., $\ell_i(C(i))$. FJR demands that it be lower than the *minimum loss of any member before deviation*, i.e., $\min_{j \in S} \ell_j(C(j))$.

Definition 2 (α-Fully Justified Representation (α-FJR))**.** For $\alpha \geqslant 1$, a k-clustering $C = (C_1, \ldots, C_k)$ satisfies α-fully justified representation (α-FJR) if there is no group of agents $S \subseteq N$ with $|S| \geqslant n/k$ such that $\alpha \cdot \ell_i(S) < \min_{j \in S} \ell_j(C(j))$ for each $i \in S$, i.e., if $\alpha \cdot \max_{i \in S} \ell_i(S) < \min_{j \in S} \ell_j(C(j))$. We refer to 1-FJR simply as FJR.

We easily see that α-FJR is a relaxation of α-core.

Proposition 1. *For* $\alpha \geqslant 1$*,* α*-core implies* α*-FJR for arbitrary loss functions.*

Proof. Suppose that a clustering C is in the α-core. Thus, for every $S \subseteq N$ with $|S| \geqslant n/k$, there exists $i \in S$ for which $\alpha \cdot \ell_i(S) \geqslant \ell_i(C(i)) \geqslant \min_{j \in S} \ell_j(C(j))$, so the clustering is also α-FJR. □

4.1 Arbitrary Loss Functions

We prove that an (exactly) FJR clustering is guaranteed to exist, even for arbitrary losses. For this, we need to define the following computational problem.

Definition 3 (Most Cohesive Cluster)**.** Given a set of agents N and a threshold τ, the Most Cohesive Cluster problem asks to find a cluster $S \subseteq N$ of size at least τ such that the maximum loss of any $i \in S$ for S is minimized, i.e., find $\arg\min_{S \subseteq N' : |S| \geqslant \tau} \max_{i \in S} \ell_i(S)$.

For $\lambda \geqslant 1$, a λ-approximate solution S satisfies $\max_{i \in S} \ell_i(S) \leqslant \lambda \cdot \max_{i \in S'} \lambda_i(S')$ for all $S' \subseteq N$ with $|S'| \geqslant \tau$, and a λ-approximation algorithm returns a λ-approximate solution on every instance.

We show that plugging in a λ-approximation algorithm $\mathcal{A}$ to the Most Cohesive Cluster problem into the GreedyCohesiveClustering algorithm designed in the previous section yields a λ-FJR clustering. In order to work with arbitrary losses, we need to consider a

[6] In centroid clustering, additional agents captured later on do not change the loss of the initial $\lceil n/k \rceil$ agents captured as loss is defined by the distance to the cluster center, which does not change. However, in non-centroid clustering, additional agents can change the loss of the initially captured agents, even from zero to positive, causing infinite core approximation when these agents deviate.

slightly generalized GreedyCohesiveClustering algorithm, which takes the loss functions ℓ_i as input instead of a metric d, and passes these loss functions to algorithm $\mathcal{A}$.

Theorem 4. *For arbitrary losses, $\alpha \geqslant 1$, and an α-approximation algorithm $\mathcal{A}$ for the Most Cohesive Cluster problem, GreedyCohesiveClustering($\mathcal{A}$) is guaranteed to return a α-FJR clustering. Hence, an (exactly) FJR clustering is guaranteed to exist.*

Proof. Suppose for contradiction that the k-clustering $C = \{C_1, \dots C_k\}$ returned by GreedyCohesiveClustering($\mathcal{A}$) on an instance is not α-FJR. Then, there exists a group $S \subseteq N$ with $|S| \geqslant n/k$ such that $\alpha \cdot \max_{i \in S} \ell_i(S) < \min_{i \in S} \ell_i(C(i))$. Let i^* be the first agent in S that was assigned to a cluster during the execution of GreedyCohesiveClustering, by calling $\mathcal{A}$ on a subset of agents N'. Note that $S \subseteq N'$. Then, we have that $\max_{i \in C(i^*)} \ell_i(C(i^*)) \geqslant \ell_{i^*}(C(i^*)) > \alpha \cdot \max_{i \in S} \ell_i(S)$, which contradicts $\mathcal{A}$ being an α-approximation algorithm for the Most Cohesive Cluster problem. Hence, GreedyCohesiveClustering($\mathcal{A}$) must return an α-FJR clustering.

Using an exact algorithm $\mathcal{A}$ for the Most Cohesive Cluster problem (e.g., the inefficient brute-force algorithm), we get that a 1-FJR clustering is guaranteed to exist. □

4.2 Average and Maximum Loss Functions

Let $\mathcal{A}^*$ be an exact algorithm for the Most Cohesive Cluster problem for the average (resp., maximum) loss. First, we notice that we cannot expect it to run in polynomial time, even for these structured loss functions. This is because it can be used to detect whether a given undirected graph admits a clique of at least a given size,[7] which is an NP-complete problem. Hence, GreedyCohesiveClustering($\mathcal{A}^*$) is an inefficient algorithm.

One can easily check that the proof of Theorem 3 extends to show that it achieves not only 1-FJR (Theorem 4), but also in the $O(n/k)$-core (resp., 2-core) for the average (resp., maximum) loss. For the core, GreedyCapture is an obvious improvement as it achieves the same approximation ratio but in polynomial time. For FJR, we show that GreedyCapture still achieves a constant approximation in polynomial time. We prove this by showing that the SmallestAgentBall algorithm used by GreedyCapture achieves the desired approximation to the Most Cohesive Cluster problem, and utilizing Theorem 4.

Lemma 1. *For the average (resp., maximum) loss, SmallestAgentBall is a 4-approximation (resp., 2-approximation) algorithm for the Most Cohesive Cluster problem, and this is tight.*

Plugging in Lemma 1 into Theorem 4, we get the following.

Corollary 1. *The (efficient) GreedyCapture algorithm is guaranteed to return a clustering that is 4-FJR (resp., 2-FJR) for the average (resp., maximum) loss.*

Determining the best FJR approximation achievable in polynomial time remains an open question.

Open Question 3: For the average (or maximum) loss, what is the smallest α for which an α-FJR clustering can be computed in polynomial time, assuming P $\neq$ NP?

Also, while Theorem 4 shows that exact FJR is achievable for the average and maximum losses, a single clustering may not achieve FJR for both losses simultaneously (the algorithm used in Theorem 4 depends on the loss function). In contrast, GreedyCapture does not depend on whether we are using the average or the maximum loss. Thus, the clustering it produces is simultaneously 4-FJR for the average loss and 2-FJR for the maximum loss (Corollary 1); this is novel even as an existential result, ignoring the fact that it can be achieved using an efficient algorithm GreedyCapture. We do not know how much this existential result can be improved upon.

[7] To detect whether an undirected graph $G = (V, E)$ has a clique of size at least t, we run $\mathcal{A}^*$ with each node being an agent, the distance between two agents being 1 if they are neighbors and 2 otherwise, and $k = n/t$. A clique of size at least t exists in G if and only if a cluster S exists with $|S| \geqslant n/k = t$ and $\max_{i \in S} \ell_i(S) = 1$.

https://doi.org/10.52202/079017-0605

Open Question 4: What is the smallest α such that there always exists a clustering that is simultaneously α-FJR for both the average loss and the maximum loss?

4.3 Auditing FJR

Next, we turn to the question of auditing the FJR approximation of a given clustering. In particular, the goal is to find the maximum FJR violation of a clustering C, i.e., the largest α for which there exists a group of agents of size at least n/k such that, if they were to form their own cluster, the loss of each of them would be lower, by a factor of at least α, than the minimum loss of any of them under clustering C. Because guarantees such as the core and FJR are defined with exponentially many constraints, it is difficult to determine the exact approximation ratio achieved by a given solution efficiently, which is why prior work has not studied auditing for proportional fairness guarantees. Nonetheless, we show that the same ideas that we used to *find* an (approximately) FJR clustering can also be used to (approximately) audit the FJR approximation of any given clustering.

Definition 4 (λ-Approximate FJR Auditing)**.** We say that algorithm $\mathcal{A}$ is a λ-approximate FJR auditing algorithm if, given any clustering C, it returns θ such that the exact FJR approximation of C (i.e., the smallest α such that C is α-FJR) is in $[\theta, \lambda \cdot \theta]$.

ALGORITHM 3: AuditFJR($\mathcal{A}$)

Input: Set of agents N, metric d, number of clusters k, clustering C
Output: Estimate θ of the FJR approximation of C

$N' \leftarrow N; \theta \leftarrow 0;$ // Remaining agents, current FJR apx estimate
while $|N'| \geqslant n/k$ **do**
  $S \leftarrow \mathcal{A}(N', d, n/k);$ // Find a cohesive group S
  $\theta \leftarrow \max\{\theta, \frac{\min_{i \in S} \ell_i(C(i))}{\max_{i \in S} \ell_i(S)}\};$ // Update θ using the FJR violation due to S
  $i^* \leftarrow \arg\min_{i \in S} \ell_i(C(i));$
  $N' \leftarrow N' \setminus \{i^*\};$ // Remove the agent with the smallest current loss
end
return θ;

We design another parametric algorithm, AuditFJR($\mathcal{A}$), presented as Algorithm 3, which iteratively calls $\mathcal{A}$ to find a 'cohesive' cluster S, similarly to GreedyCohesiveClustering. But while GreedyCohesiveClustering removes all the agents in S from further consideration, AuditFJR removes only the agent in S with the smallest loss under the given clustering C from further consideration. Thus, instead of finding up to k cohesive clusters, it finds up to n cohesive clusters. It returns the maximum FJR violation of C demonstrated by any of these n possible deviating coalitions (recall that the exact FJR approximation of C is the maximum FJR violation across all the exponentially many possible deviating coalitions of size at least $\lceil n/k \rceil$).

We show that if $\mathcal{A}$ was a λ-approximation algorithm for the Most Cohesive Cluster problem, then the resulting algorithm is a λ-approximate FJR auditing algorithm. In particular, if we were to solve the Most Cohesive Cluster problem exactly in each iteration (which would be inefficient), the maximum FJR violation across those n cohesive clusters found would indeed be the maximum FJR violation across all the exponentially many deviating coalitions, an apriori nontrivial insight. Fortunately, we can at least plug in the SmallestAgentBall algorithm, which we know achieves constant approximation to the Most Cohesive Cluster problem (Lemma 1).

Theorem 5. *For $\lambda \geqslant 1$, if $\mathcal{A}$ is a λ-approximation algorithm to the Most Cohesive Cluster problem, then AuditFJR($\mathcal{A}$) is a λ-approximate FJR auditing algorithm. Given Lemma 1, it follows that for the average (resp., maximum) loss, AuditFJR(SmallestAgentBall) is an efficient 4-approximate (resp., 2-approximate) FJR auditing algorithm.*

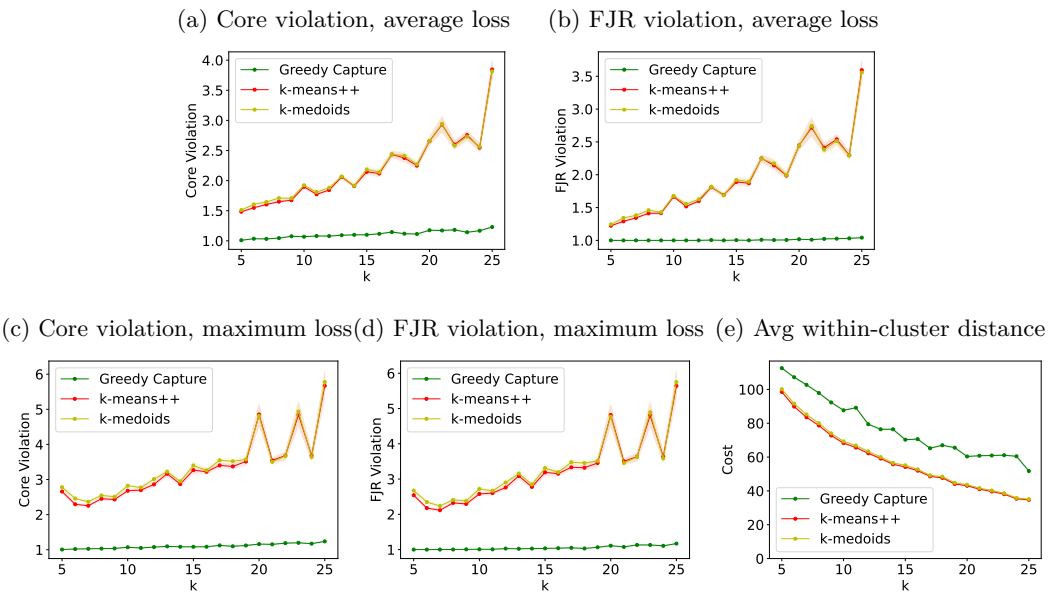


Figure 1: Census Income Dataset

Unfortunately, the technique from Theorem 5 does not extend to auditing the core. This is because it requires upper bounding $\min_{i \in S} \frac{\ell_i(C(i))}{\ell_i(S)}$ (instead of $\frac{\min_{i \in S} \ell_i(C(i))}{\max_{i \in S} \ell_i(S)}$); this can be upper bounded by $\frac{\ell_{i^*}(C(i^*))}{\ell_{i^*}(S)}$, but we cannot lower bound $\ell_{i^*}(S)$. The fact that $\mathcal{A}$ approximates the Most Cohesive Cluster problem only lets us lower bound $\max_{i \in S} \ell_i(S)$. We leave it as an open question whether an efficient, constant-approximate core auditing algorithm can be devised.

Open Question 5: Does there exist a polynomial-time, α-approximate core auditing algorithm for some constant α?

For the maximum loss, we can show that our 2-approximate FJR auditing algorithm is the best one can hope for in polynomial time; the proof is in Appendix A. The case of average loss remains open.

Theorem 6. *Assuming $P \neq NP$, there does not exist a polynomial-time λ-approximate FJR auditing algorithm for the maximum loss, for any $\lambda < 2$.*

5 Experiments

In this section, we empirically compare GreedyCapture with the popular clustering algorithms k-means++ and k-medoids on real data. Our focus is on the tradeoff between fairness (measured by the core and FJR) and accuracy (measured by traditional clustering objectives) they achieve.

Datasets. We consider three different datasets from the UCI Machine Learning Repository [19], namely Census Income, Diabetes, and Iris. For the first two datasets, each data point corresponds to a human being, and it is reasonable to assume that each individual prefers to be clustered along with other similar individuals. We also consider the third dataset for an interesting comparison with the empirical work of Chen et al. [1], who compared the same algorithms but for centroid clustering.

The Census Income dataset contains demographic and economic characteristics of individuals, which are used to predict whether an individual's annual income exceeds a threshold. For our experiments, we keep all the numerical features (i.e. `age`, `education-num`, `capital-gain`,

https://doi.org/10.52202/079017-0605

`capital-loss`, and `hours-per-week`) along with `sex`, encoded as binary values. There are in total 32,561 data points, each with a sample weight attribute (`fnlwgt`). The Diabetes dataset contains numerical features, such as age and blood pressure, for about 768 diabetes patients. The Iris dataset consists of 150 records of numerical features related to the petal dimensions of different types of iris flowers.

Measures. For fairness, we measure the true FJR and core approximations of each algorithm with respect to both the average and maximum losses. For accuracy, we use three traditional clustering objectives: the average within-cluster distance $\sum_{t\in[k]} \frac{1}{|C_t|} \cdot \sum_{i,j\in C_t} d(i,j)$, termed *cost* by Ahmadi et al. [14], as well as the popular k-means and k-medoids objectives.

Experimental setup. We implement the standard k-means++ and k-medoids clustering algorithms from the Scikit-learn project[8], averaging the values for each measure over 20 runs, as their outcomes depend on random initializations. The computation of GreedyCapture neither uses randomization nor depends on the loss function with respect to which the core or FJR approximation is measured. Since calculating core and FJR approximations requires considerable time, for both the Census Income and Diabetes datasets, we sample 100 data points and plot the means and standard deviations over 40 runs. For the former, we conduct weighted sampling according to the `fnlwgt` feature.

Results. In Figure 1, we see the results for the Census Income dataset; the results for k-means and k-medoids objectives for this dataset, along with results for the other two datasets, are relegated to Appendix D due to being qualitatively similar to the results presented here. According to all four fairness metrics, GreedyCapture is significantly fairer than both k-means++ and k-medoids, consistently across different values of k. Notably, the FJR approximation of GreedyCapture empirically stays very close to 1 in all cases, in contrast to the higher worst-case bounds (Corollary 1). Remarkably, the significant fairness advantage of GreedyCapture comes at a modest cost in accuracy: all three objective values (average within-cluster distance, k-means, and k-medoids) achieved by GreedyCapture are less than *twice* those of k-means++ and k-medoids, across all values of k and all three datasets!

Lastly, our results are in contrast to the experiments of Chen et al. [1] for centroid clustering, where GreedyCapture provides a worse core approximation than k-means++ on Iris and Diabetes datasets; as demonstrated in Appendix D, this is not the case in non-centroid clustering.

6 Discussion

We have initiated the study of proportional fairness in non-centroid clustering. Throughout the paper, we highlight several intriguing open questions. Probably the most important of these are whether we can achieve a better approximation than $O(n/k)$ of the core for the average loss, and whether the core is always non-empty for the maximum loss. In an effort to answer the latter question, in Appendix C.1 we show that the core is always non-empty for the maximum loss when the metric space is 1-dimensional (i.e., a line). This contrasts with the average loss, for which the core remains empty even on the line (see Appendix C.2).

In our work, we have shown that there are remarkable differences between centroid and non-centroid clustering settings. One can consider a more general model, where the loss of an agent depends on both her cluster center and the other agents in her cluster. Investigating what proportional fairness guarantees can be achieved in this case is an exciting direction. Another intriguing question is whether we can choose the number of clusters k intrinsically; this seems challenging as proportional fairness guarantees seem to depend on fixing k in advance to define which coalitions can deviate. Lastly, while classical algorithms such as k-means and k-centers are incompatible with the core and FJR in the worst case (see Appendix E), it is interesting to explore conditions under which they may be more compatible, and whether a fair clustering can be computed efficiently in such cases.

[8] https://scikit-learn.org

Acknowledgements

Caragiannis was partially supported by the Independent Research Fund Denmark (DFF) under grant 2032-00185B. Shah was partially supported by an NSERC Discovery grant.

References

[1] Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 1032–1041, 2019.

[2] Evi Micha and Nisarg Shah. Proportionally fair clustering revisited. In *Proceedings of the 47th International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 85:1–85:16, 2020.

[3] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3710–3722, 2020.

[4] Bhaskar Ray Chaudhury, Linyi Li, Mintong Kang, Bo Li, and Ruta Mehta. Fairness in federated learning via core-stability. In *Proceedings of the 35th Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 5738–5750, 2022.

[5] Bhaskar Ray Chaudhury, Aniket Murhekar, Zhuowen Yuan, Bo Li, Ruta Mehta, and Ariel D Procaccia. Fair federated learning via the proportional veto core. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024. Forthcoming.

[6] R Janani and S Vijayarani. Text document clustering using spectral clustering algorithm with particle swarm optimization. *Expert Systems with Applications*, 134:192–200, 2019.

[7] Hong Huang, Fanzhi Meng, Shaohua Zhou, Feng Jiang, and Gunasekaran Manogaran. Brain image segmentation based on fcm clustering algorithm and rough set. *IEEE Access*, 7:12386–12396, 2019.

[8] Hsuan-Wei Lee, Nishant Malik, Feng Shi, and Peter J Mucha. Social clustering in epidemic spread on coevolving networks. *Physical Review E*, 99(6):062301, 2019.

[9] Leon Kellerhals and Jannik Peters. Proportional fairness in clustering: A social choice perspective. arXiv:2310.18162, 2023.

[10] Haris Aziz, Barton E Lee, Sean Morota Chu, and Jeremy Vollen. Proportionally representative clustering. arXiv:2304.13917, 2023.

[11] Donald Bruce Gillies. *Some theorems on n-person games*. Princeton University, 1953.

[12] Dominik Peters, Grzegorz Pierczyński, and Piotr Skowron. Proportional participatory budgeting with additive utilities. In *Proceedings of the 34th Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 12726–12737, 2021.

[13] Anshuman Chhabra, Karina Masalkovaitė, and Prasant Mohapatra. An overview of fairness in clustering. *IEEE Access*, 9:130698–130720, 2021.

[14] Saba Ahmadi, Pranjal Awasthi, Samir Khuller, Matthäus Kleindessner, Jamie Morgenstern, Pattara Sukprasert, and Ali Vakilian. Individual preference stability for clustering. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, pages 197–246, 2022.

[15] Haris Aziz and Rahul Savani. Hedonic games. In Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors, *Handbook of Computational Social Choice*, pages 356–376. Cambridge University Press, 2016.

[16] Haris Aziz, Florian Brandl, Felix Brandt, Paul Harrenstein, Martin Olsen, and Dominik Peters. Fractional hedonic games. *ACM Transactions on Economics and Computation*, 7(2):6:1–6:29, 2019.

 https://doi.org/10.52202/079017-0605

[17] Hal R Varian. Equity, envy and efficiency. *Journal of Economic Theory*, 9:63–91, 1974.

[18] Vincent Conitzer, Rupert Freeman, Nisarg Shah, and Jennifer Wortman Vaughan. Group fairness for the allocation of indivisible goods. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1853–1860, 2019.

[19] D. Dua and C. Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Appendix

A Missing Proofs

A.1 Proof of Theorem 1

Theorem 1. *For arbitrary losses, there exists an instance in which no α-core clustering exists for any finite α.*

Proof. Consider an instance with a set of $n = 4$ agents $\{0, 1, 2, 3\}$ and $k = 2$. Note that any group of at least 2 agents deserves to form a cluster. For $i \in \{0, 1, 2\}$, the loss function of agent i is given by

$$\ell_i(S) = \begin{cases} \infty & \text{if } S = \{0, 1, 2\} \text{ or } 3 \in S, \\ 1 & \text{if } |S| = 2 \text{ and } i + 1 \bmod 3 \notin S, \\ 0 & \text{if } S = \{i, i + 1 \bmod 3\}. \end{cases}$$

In words, agent $i \in \{0, 1, 2\}$ only wants to be in a cluster of size 2 that does *not* include the undesirable agent 3; any other cluster has infinite loss. In an ideal such cluster (loss 0), agent 0 prefers to be with agent 1, agent 1 prefers to be with agent 2, and agent 2 prefers to be with agent 0. The remaining clusters of size 2 have loss 1.

Consider any clustering $C = (C_1, C_2)$. Without loss of generality, say $3 \in C_1$. We take three cases.

1. If $|C_1| = 1$, then $C_2 = \{0, 1, 2\}$. Then, a group S containing any two agents from C_2 can deviate, and each $i \in S$ would improve from infinite loss to finite loss.
2. If $|C_1| \geqslant 3$, then a group S containing two agents from C_1 other than agent 3 can deviate, and each $i \in S$ would improve from infinite loss to finite loss.
3. Suppose $|C_1| = 2$ and let $C_1 \cap \{0, 1, 2\} = \{i\}$. Then, the group $S = \{i, (i-1) \bmod 4\}$ can deviate: agent i would improve from infinite loss to finite loss, and agent $(i - 1) \bmod 4$ would improve from a loss of 1 to a loss of 0.

In each case, every deviating agent improves by an infinite factor, yielding the desired result. □

A.2 Proof of Theorem 2

Theorem 2. *For the average loss, there exists an instance in which no α-core clustering exists for $\alpha < \frac{1+\sqrt{3}}{2} \approx 1.366$.*

Proof. Let us construct an instance with an even number $k \geqslant 2$ of clusters. Let $\varepsilon = \frac{1+\sqrt{3}}{2} - \alpha$. We set the number of agents to be a multiple of k such that $n \geqslant k \cdot \max\left\{\frac{1}{2\varepsilon} + \frac{1}{2}, 4\alpha^2\right\}$. Our construction has $k/2+1$ *areas*, each consisting of a few locations (points), with several agents placed on each of them. In particular, area 0 has a single location M_0 with $k/2$ agents. For $i = 1, 2, ..., k/2$, area i consists of location M_i hosting a single agent, a left location L_i and a right location R_i each hosting $n/k - 1$ agents. We use L_i, R_i, and M_i to denote both the corresponding points as well as the set of agents located in them. For $i = 1, 2, ..., k/2$, the distance between points L_i and R_i is 1 while both points are at distance $\frac{n}{2k\alpha}$ from point M_i. The distance between any two points in different areas is infinite.

Consider a k-clustering C of the agents. We call *bad* any cluster of C that contains agents from different areas; notice that all points in such a cluster have infinite cost. A *good* cluster has all its points in the same area and, hence, all the agents contained in it have finite cost. Notice that C has at most $k - 1$ good clusters that contain points from areas 1, 2, ..., $k/2$. Among these areas, let t be the one with the minimum number of good clusters. Thus, area t either has all its agents in bad clusters or contains one good cluster that includes some of

https://doi.org/10.52202/079017-0605

its agents. If at least n/k of its agents belong to bad clusters in C, a deviating coalition of them would improve their cost from infinite to finite. So, in the following, we assume that clustering C contains exactly one good cluster with at least n/k agents from area t.

We distinguish between three cases. The first one is when the good cluster does not contain the agent in M_t. Among R_t and L_t, assume that L_t has at most as many agents in the good cluster as R_t (the other subcase is symmetric). Then, the cost of all agents of L_t in the good cluster is at least $1/2$. The deviating coalition consisting of all agents in L_t and the agent of M_t (i.e., n/k agents in total) improves the cost of all agents by a multiplicative factor at least α. Indeed, the cost of the agent in M_t improves from infinite to finite while the cost of any agent in L_t improves from at least $1/2$ to $\frac{1}{2\alpha}$, since any such agent has distance $\frac{n}{2k\alpha}$ to the agent in M_t, and is colocated with the other agents in the deviating coalition.

The second case is when the good cluster contains all agents in area t. In this case, the cost of the agents in L_t and R_t is $\frac{\frac{n}{2k\alpha}+n/k-1}{2^{n/k}-1} \geqslant \frac{1}{4\alpha}+\frac{1}{2}-\frac{1}{2(2^{n/k}-1)} \geqslant \frac{1}{4\alpha}+\frac{1}{2}-\frac{\varepsilon}{2}$ (the second inequality follows by the definition of n). Then, each agent in the deviating coalition containing $\frac{n}{2k}$ agents from L_t and $\frac{n}{2k}$ agents from R_t improves their cost to $1/2$, i.e., by a factor of at least $\frac{1}{2\alpha}+1-\varepsilon$.

The third case is when the good cluster contains the agent in M_t but does not contain some agent i from L_t or R_t. We will assume that agent i belongs to L_t (the other subcase is symmetric). Notice that the cost of the agents in R_t is at least $\frac{\frac{n}{2k\alpha}}{2^{n/k}-2} \geqslant \frac{1}{4\alpha}$. To see why, notice that the claim is trivial for those agents of R_t that belong to bad clusters while each of the agents of R_t in the good cluster is at distance $\frac{n}{2k\alpha}$ to the agent in M_t and there are at most $2^{n/k}-2$ in the cluster. The deviating coalition of all agents in R_t together with i decreases their cost to just k/n, i.e., by a factor of at least $\frac{n}{4k\alpha} \geqslant \alpha$ (the inequality follows by the definition of n), while the cost of agent i improves from infinite to finite.

So, there is always a deviating coalition of at least n/k agents with each of them improving their cost by a multiplicative factor of $\min\left\{\alpha, 1+\frac{1}{2\alpha}-\varepsilon\right\} = \alpha$, as desired. The last equality follows by the definition of α and ε. □

A.3 Proof of Theorem 3

Theorem 3. *For the average (resp., maximum) loss, the* GreedyCapture *algorithm is guaranteed to return a clustering in the* $(2 \cdot \lceil n/k \rceil - 3)$*-core (resp.,* 2*-core) in* $O(kn)$ *time complexity, and these bounds are (almost) tight.*

Proof. Let $C = \{C_1, \dots C_k\}$ be the k-clustering returned by GreedyCapture. Let $S \subseteq N$ be any set of at least n/k agents such that their average loss satisfies

$$\ell_i(C(i)) > (2 \cdot \lceil n/k \rceil - 3) \cdot \ell_i(S), \tag{1}$$

for every $i \in S$.

Let i^* be the agent that was the first among the agents in S that was included in some cluster by the algorithm. Consider the time step before this happens and let $i' \in C(i^*)$ be the agent that had the minimum distance R from the $\lceil n/k \rceil$-th agent in $C(i^*)$ among all agents that had not been included to clusters by the algorithm before. Then,

$$\begin{aligned}
\ell_{i^*}(C(i^*)) &= \frac{1}{|C(i^*)|} \sum_{i \in C(i^*)} d(i^*, i) \\
&\leqslant \frac{1}{|C(i^*)|} \left(d(i^*, i') + \sum_{i \in C(i^*) \setminus \{i', i^*\}} (d(i^*, i') + d(i', i)) \right) \\
&\leqslant \left(2 - \frac{3}{\lceil n/k \rceil}\right) \cdot R.
\end{aligned} \tag{2}$$

The first inequality follows by applying the triangle inequality. The second inequality follows since $C(i^*)$ has $\lceil n/k \rceil$ agents and, thus, the RHS has $2\lceil n/k \rceil - 3$ terms representing distances of agents in $C(i^*)$ from agent i', each bounded by R.

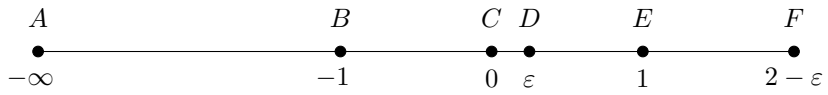


Figure 2: The instance used to show the lower bounds in Theorem 3 and Lemma 1.

Now, recall that, at the time step the algorithm includes cluster $C(i^*)$ in the clustering, none among the (at least $\lceil n/k \rceil$) agents of S have been included in any clusters. Then, S contains at most $\lceil n/k \rceil - 1$ agents located at distance less than R from agent i^*; if this were not the case, the algorithm would have included agent i^* together with $\lceil n/k \rceil - 1$ other agents of S in a cluster instead of the agents in $C(i^*)$. Thus, S contains at least $|S| - \lceil n/k \rceil + 1$ agents at distance at least R from agent i^*. Thus,

$$\ell_{i^*}(S) = \frac{1}{|S|} \sum_{i \in S} d(i^*, i) \geqslant \frac{|S| - \lceil n/k \rceil + 1}{|S|} \cdot R \geqslant \frac{1}{\lceil n/k \rceil} \cdot R. \tag{3}$$

The second inequality follows since $|S| \geqslant \lceil n/k \rceil$. Now, Equation (2) and Equation (3) yield $\ell_{i^*}(C(i^*)) \leqslant (2 \cdot \lceil n/k \rceil - 3) \cdot \ell_{i^*}(S)$, contradicting Equation (1).

Now, assume that there exists a set $S \subseteq N$ of at least n/k agents such that their maximum loss satisfies

$$\ell_i(C(i)) > 2 \cdot \ell_i(S), \tag{4}$$

for every $i \in S$. Again, let i^* be the agent that was the first among the agents in S that was included in some cluster by the algorithm. Consider the time step before this happens and let $i' \in C(i^*)$ be the agent that had the minimum distance R from the $\lceil n/k \rceil$-th agent in $C(i^*)$ among all agents that had not been included to clusters by the algorithm before. Then, the maximum loss of agent i^* for cluster $C(i^*)$ is

$$\ell_{i^*}(C(i^*)) = \max_{i \in C(i^*)} d(i^*, i) \leqslant \max_{i \in C(i^*)} (d(i^*, i') + d(i', i)) \leqslant 2 \cdot R. \tag{5}$$

The first inequality follows by applying the triangle inequality and the second one since all agents in $C(i^*)$ are at distance at most R from agent i'. We also have

$$\ell_{i^*}(S) = \max_{i \in S} d(i^*, i) \geqslant R, \tag{6}$$

otherwise, the algorithm would include a subset of $\lceil n/k \rceil$ agents from set S in the clustering instead of $C(i^*)$. Together, Equation (5) and Equation (6) contradict Equation (4). This completes the proof of the upper bounds.

We now show that the analysis is tight for both the average and the maximum loss functions using the instance depicted in Figure 2 with one agent at locations A, D, and E, two agents at location B, $n/2 - 3$ agents at location C, and $n/2 - 2$ agents at location F. Suppose that $k = 2$. It is easy to see that GreedyCapture returns a 2-clustering with the agents located at points A, B, and C in one cluster and the agents located at points D, E, and F in another. Notice that the agents at locations B and C have infinite loss under both loss functions, while the agent located at position D has maximum loss $2(1 - \varepsilon)$ and average loss $\frac{(n-3)(1-\varepsilon)}{n/2}$. Now, consider the deviating coalition of the $n/2$ agents at locations B, C, and D. The agents at B and C improve their loss from infinite to finite, while the agent located at C improves her maximum loss to $1 + \varepsilon$ and her average loss to $\frac{2+(n/2-1)\varepsilon}{n/2}$, for multiplicative improvements approaching 2 and $n/2 - 3/2$ as ε approaching 0.

Since GreedyCapture calls SmallestAgentBall at most k times and SmallestAgentBall does at most n iterations in each call, we easily see that the time complexity of GreedyCapture is $O(kn)$.

□

A.4 Proof of Lemma 1

Lemma 1. *For the average (resp., maximum) loss, SmallestAgentBall is a 4-approximation (resp., 2-approximation) algorithm for the Most Cohesive Cluster problem, and this is tight.*

https://doi.org/10.52202/079017-0605

Proof. For some $N' \subseteq N$, let S be the most cohesive cluster and let $S' \neq S$ be the cluster that SmallestAgentBall returns. Suppose that S' consists by the $\lceil n/k \rceil$ closest agents in N' to some agent i^* and the distance of i^* to her $\lceil n/k \rceil$-th closest agent in N' is equal to R. From the triangle inequality, we get that every two agents in S' have distance at most $2R$, and therefore, under both loss functions, we get that $max_{i \in S'} \ell_i(S') \leqslant 2 \cdot R$.

We show that there are two individuals in S, i_1 and i_2, such that the $d(i_1, i_2) \geqslant R$. Indeed if for each $i, i' \in S$, $d(i, i') < R$, then i^* would not be the agent in N' with the smallest distance to her $\lceil n/k \rceil$-th closest agent in N' and SmallestAgentBall would not return S. From this fact, we immediately get a 2-approximation for the maximum loss, since $max_{i \in S} \ell_i(S) \geqslant R$.

Now, for the average cost, note that

$$|S| \cdot d(i_1, i_2) = \sum_{i \in S} d(i_1, i_2) \leqslant \sum_{i \in S} \left(d(i_1, i) + d(i, i_2) \right).$$

From this we get, that either $\sum_{i \in S} d(i_1, i) \geqslant |S| \cdot d(i_1, i_2)/2$ or $\sum_{i \in S} d(i_2, i) \geqslant |S| \cdot d(i_1, i_2)/2$. Therefore, either $\ell_{i_1}(S) \geqslant d(i_1, i_2)/2 \geqslant R/2$ or $\ell_{i_2}(S) \geqslant d(i_1, i_2)/2 \geqslant R/2$. This means that $max_{i \in S} \ell_i(S) \geqslant R/2$ and the lemma follows.

Next, we show that there are instances for which SmallestAgentBall achieves exactly these bounds. Consider the instance showing in Figure 2, For $k = 2$, suppose there are $= 1$ point at position A, $n/4$ points at position B, $n/4 - 1$ at position C, 1 point at position D, 1 point at position E at position D, and $n/2 - 2$ points at position F. It is not hard to see that SmallestAgentBall will return the cluster $S = \{D, E, F\}$. But $S' = \{B, C, D\}$ can reduce the average loss by a factor equal to 4 and the maximum loss by a factor equal to 2 as n grows and ϵ goes to 0. □

A.5 Proof of Theorem 5

Theorem 5. *For $\lambda \geqslant 1$, if $\mathcal{A}$ is a λ-approximation algorithm to the Most Cohesive Cluster problem, then AuditFJR($\mathcal{A}$) is a λ-approximate FJR auditing algorithm. Given Lemma 1, it follows that for the average (resp., maximum) loss, AuditFJR(SmallestAgentBall) is an efficient 4-approximate (resp., 2-approximate) FJR auditing algorithm.*

Proof. Suppose $\mathcal{A}$ is a λ-approximation algorithm for the Most Cohesive Cluster problem. Consider any clustering C on which AuditFJR($\mathcal{A}$) returns θ. Let $\rho = \max_{S \subseteq N : |S| \geqslant \lceil n/k \rceil} \frac{\min_{i \in S} \ell_i(C(i))}{\max_{i \in S} \ell_i(S)}$ be the exact FJR approximation of C. First, it is easy to check that $\rho \geqslant \theta$ because θ is computed by taking the maximum of the same expression as ρ is, but over only some (instead of all possible) S. Hence, it remains to prove that $\rho \leqslant \lambda \cdot \theta$.

Consider any group $S \subseteq N$ with $|S| \geqslant n/k$. Let i^* be the first agent in S that was removed by AuditFJR, say when $\mathcal{A}$ returned a group S' containing it; there must be one such agent because $|S| \geqslant n/k$ and when AuditFJR stops, fewer than n/k agents remain in N'. Now, we have that

$$\frac{\min_{i \in S} \ell_i(C(i))}{\max_{i \in S} \ell_i(S)} \leqslant \frac{\ell_{i^*}(C(i^*))}{\max_{i \in S} \ell_i(S)} \leqslant \lambda \cdot \frac{\ell_{i^*}(C(i^*))}{\max_{i \in S'} \ell_i(S')} = \lambda \cdot \frac{\min_{i \in S'} \ell_i(C(i))}{\max_{i \in S'} \ell_i(S')} \leqslant \lambda \cdot \theta,$$

where the second inequality holds because $\mathcal{A}$ is a λ-approximation algorithm for the Most Cohesive Cluster problem, which implies $\max_{i \in S'} \ell_i(S') \leqslant \lambda \cdot \max_{i \in S} \ell_i(S)$; the next equality holds because agent i^* was selected for removal when S' was returned, which implies $i^* \in \arg\min_{i \in S'} \ell_i(C(i))$; and the final inequality holds because θ is updated to be the maximum of all FJR violations witnessed by the algorithm, and violation due to S' is one of them.

Finally, using the approximation ratio bound of SmallestAgentBall for the Most Cohesive Cluster problem from Lemma 1, we obtain the desired approximate auditing guarantee of AuditFJR(SmallestAgentBall). □

A.6 Proof of Theorem 6

Theorem 6. *Assuming $P \neq NP$, there does not exist a polynomial-time λ-approximate FJR auditing algorithm for the maximum loss, for any $\lambda < 2$.*

Proof. We show that such an algorithm can be used to solve the CLIQUE problem, which asks whether a given undirected graph $G = (V, E)$ admits a clique of size at least t. The problem remains hard with $t \geqslant 3$, so we can assume this without loss of generality. Given (G, t), we first modify $G = (V, E)$ into $G' = (V', E')$ as follows. To each $v \in V$, we attach $t - 2$ new (dummy) nodes, and to one of those dummy nodes, we attach yet another dummy node. In total, for each $v \in V$, we are adding $t - 1$ dummy nodes, so the final number of nodes is $|V'| = |V| \cdot t$.

Next, we create an instance of non-centroid clustering with $n = |V'|$ agents, one for each $v \in V'$. The distance $d(u, v)$ is set as the length of the shortest path between u and v. Set $k = |V|$.

Consider a clustering C in which each real node $v \in V$ is put into a separate cluster, along with the $t - 1$ dummy nodes created for it. Note that $\ell_v(C(v)) = 2$ for each real node $v \in V$ (due to the dummy node attached to a dummy node attached to v) and $\ell_v(C(v)) \in \{2, 3\}$ for each dummy node $v \in V' \setminus V$. Let us now consider possible deviating coalitions S.

If a dummy node v is included in S, then in order for an FJR violation, its maximum loss would have to be strictly reduced. If $\ell_v(C(v)) = 2$, then we must have $\ell_v(S) = 1$, but no dummy node has at least $t \geqslant 3$ nodes within a distance of 1. If $\ell_v(C(v)) = 3$, then in order to find at least t nodes within a distance of at most 2 (and the set not be identical to one of the clusters), S must include at least one real node v' that is not associated with the dummy node v. However, in this case, $\ell_{v'}(S) \geqslant 2$ whereas $\ell_{v'}(C(v')) = 2$, so no FJR violation is possible.

The only remaining case is when S consists entirely of real nodes. Since $\ell_v(C(v)) = 2$ for every real node $v \in V$, an FJR violation exists if an only if $\max_{v \in S} \ell_v(S) = 1$, which happens if and only if S is a clique of real nodes size at least t.

Thus, we have established that the FJR approximation of C is 2 if there exists a clique of size at least t in G, and 1 otherwise. Since a λ-approximate auditing algorithm with $\lambda < 2$ can distinguish between these two possibilities, it can be used to solve the CLIQUE problem. □

B Bicriteria Approximation of the Core

Here, we consider a more general definition of the core.

Definition 5 ((α, δ)-Core)**.** For $\alpha \geqslant 1$, a k-clustering $C = (C_1, \ldots, C_k)$ is said to be in the (α, δ)-core if there is no group of agents $S \subseteq N$ with $|S| \geqslant \delta \cdot n/k$ such that $\alpha \cdot \ell_i(S) < \ell_i(C(i))$ for all $i \in S$.

Theorem 7. *GreedyCapture returns a clustering solution in the $(\delta, \frac{2\delta}{\delta - 1})$-core, for any $\delta > 1$.*

Proof. Let $C = \{C_1, \ldots C_k\}$ be a solution that GreedyCapture returns. Suppose for contradiction that there exists $S \subseteq N$ with $|S| \geqslant \delta \cdot n/k$ such that

$$\forall i \in S, \qquad \ell_i(C(i)) > \frac{2\delta}{\delta - 1} \cdot \ell_i(S).$$

Let i^* be the agent that was the first among the agents in S that was included in some cluster by the algorithm. Consider the time step before this happens and let $i' \in C(i^*)$ be the agent that had the minimum distance R from the $\lceil n/k \rceil$-th agent in $C(i^*)$ among all agents that had not been included to clusters by the algorithm before. With similar arguments as in the proof of Theorem 3, we conclude that

$$\ell_{i^*}(C(i^*)) \leqslant \left(2 - \frac{3}{\lceil n/k \rceil}\right) \cdot R \leqslant 2 \cdot R.$$

https://doi.org/10.52202/079017-0605

Again, with very similar arguments as in the proof of Theorem 3, we can conclude that that S contains at least $|S| - \lceil n/k \rceil + 1$ agents at distance at least R from agent i^*. Thus,

$$\ell_{i^*}(S) = \frac{1}{|S|} \sum_{i \in S} d(i^*, i) \geqslant \frac{|S| - \lceil n/k \rceil + 1}{|S|} \cdot R \geqslant \frac{|S| - n/k}{|S|} \cdot R \geqslant \frac{\delta - 1}{\delta} \cdot R.$$

where the second inequality follows since $|S| \geqslant \delta \cdot n/k$ and the theorem follows. □

C Line

C.1 Non-Emptiness of the Core for Maximum Loss

ALGORITHM 4: SmallestDiameter

Input: $N' \subseteq N$, metric d, k, t
Output: S
if $|N'| < t$ **then**
  $S \leftarrow N'$;
else
  Label the agents from 1 to n', starting with the leftmost agent and moving to the right;
  $d_{min} \leftarrow d(1, n')$;
  $i^* \leftarrow 1$;
  for $i = 1$ *to* $n' - t$ **do**
    if $d(i, i + t) < d_{min}$ **then**
      $d_{min} \leftarrow d(i, i + t)$;
      $i^* \leftarrow i$;
    end
  end
end
$S \leftarrow \{i^*, \ldots, i^* + t\}$;

Theorem 8. *For the maximum loss in the line,* GreedyCohesiveClustering *(*SmallestDiameter*) returns a solution in the core in $O(kn)$ time complexity.*

Proof. Let $C = \{C_1, \ldots, C_k\}$ be the solution that the algorithm returns. Suppose for contradiction that there exists a group $S \subseteq N$, with $|S| \geqslant n/k$ such that $\ell_i(C(i)) > \ell_i(S)$ for all $i \in S$. We denote the leftmost and rightmost agents in S by L and R, respectively. Let i^* be the first agent in S that was assigned to some cluster. If we denote with N' the set of agents that have not been disregarded before this happens, this means that $S \subseteq N'$. We denote the leftmost and rightmost agents in $C(i^*)$ by L^* and R^*, respectively.

Note that i^* has incentives to deviate if and only if either $d(i^*, L) < d(i^*, L^*)$ or $d(i^*, R) < d(i^*, R^*)$, since otherwise $\ell_{i^*}(S) = \max\{d(i^*, L), d(i^*, R)\} \geqslant \ell_{i^*}(C(i^*)) = \max\{d(i^*, L^*), d(i^*, R^*)\}$. Without loss of generality, assume that $d(i^*, L) < d(i^*, L^*)$. Given the way that the algorithm operates, it is not hard to see that since L^* and i^* are included in $C(i^*)$ and L is located between L^* and i^*, then L is also included in $C(i^*)$. Denote with R' the $\lceil n/k \rceil$-th agent to the right of L in N'. We notice that the algorithm returns $C(i^*)$ instead of S, because $d(L^*, R^*) \leqslant d(L, R') \leqslant d(L, R)$. But this means that

$$\ell_L(C(i^L)) = \ell_L(C(i^*)) \leqslant d(L^*, R^*) \leqslant d(L, R) = \ell_L(S)$$

and we reach in a contradiction.

Since GreedyCohesiveClustering calls SmallestDiameter at most k times and SmallestDiameter does at most n iterations in each call, we easily see that the time complexity of GreedyCohesiveClustering (SmallestDiameter) is $O(kn)$. □

C.2 Emptiness of the Core for Average Loss

Theorem 9. *For $k = 2$ and the average loss, there exists an instance in the line where the core is empty.*

Proof. Consider the instance with even $n > 24$, where one agent, denoted by a, is located at position 0, $n/2 - 1$ agents, denoted by the set S_1, are located at position 2, $n/2 - 1$ agents, denoted by the set S_2, are located at position 3 and the last agent denoted by b is located at position $+\infty$.

Let $C = (C_1, C_2)$ be any clustering solution. Without loss of generality, suppose that C_1 contains b. This means that all the agents that are part of C_1 have loss equal to infinity. Note that if $|C_2| \leqslant n/2 - 1$, then $|C_1 \cap (S_1 \cup S_2 \cup \{a\})| \geqslant n/2 + 1$ which means that n/k agents from $S_1 \cup S_2 \cup \{a\}$ could reduce their loss arbitrary much by deviating to their own cluster. Hence, $|C_2| \geqslant n/2$. Next, we distinguish to two cases:

Case I: $|C_2 \cap S_2| \geqslant n/4$**.** In this case, for each agent i in S_1, we have that $\ell_i(C(i)) \geqslant \frac{n/4}{n-1} \geqslant 1/4$. Moreover, note that a always prefers to be in a cluster that consists by agents in S_1. Therefore, we have that if a and S_1 deviate to their own cluster, then for each $i \in S_1$ $\ell_i(S_1 \cup \{a\}) = \frac{1}{n/2} < 1/4$, where the last inequality follows from the fact that $n > 8$.

Case II: $|C_2 \cap S_2| < n/4$**.**

Since $|C_2| \geqslant n/2$, we have that $|C_2 \cap S_1| \geqslant n/4$. In this case, for each agent i in S_2, we have that $\ell_i(C(i)) \geqslant \frac{n/4}{n-1} \geqslant 1/4$.

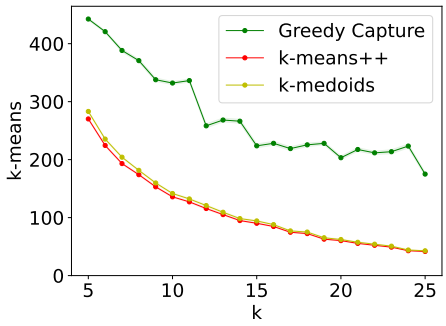
Now, we distinguish to two further subcases. First, suppose that a belongs to C_1. Then, if the agents in S_2 and a deviate to their own cluster, we have that for each $i \in S_2$, $\ell_i(S_2 \cup \{a\}) = \frac{3}{n/2} < 1/4$, where the last inequality follows from the fact that $n > 24$. If a is assigned to C_2, then all the agents in S_1 have incentives to deviate with an agent from S_2 that is assigned to C_1.

□

D More Experimental Results

We conducted our experiments on a server with 32 cores / 64 threads at 4.2 GHz and 128 GB of RAM.

(a) k-means objective



(b) k-medoids objective

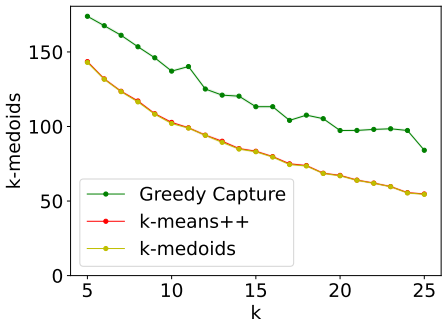


Figure 3: Remaining figures for the Census Income Dataset

E Incompatibility of FJR and Core with Classical Objectives

Consider Example 1 from Chen et al. [1]. Classic algorithms such as k-center, k-means++, and k-median would cluster all points at positions a and b together. But if the points at

https://doi.org/10.52202/079017-0605

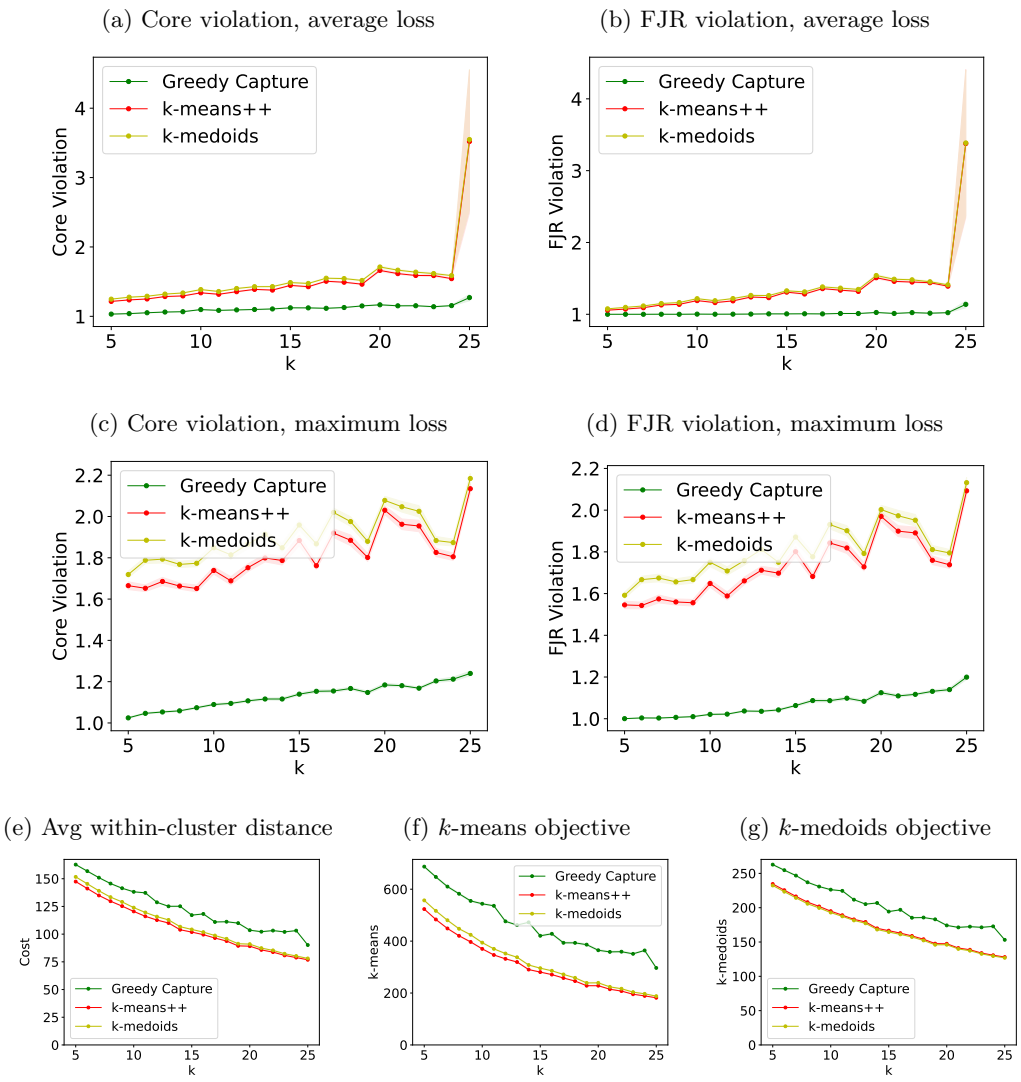
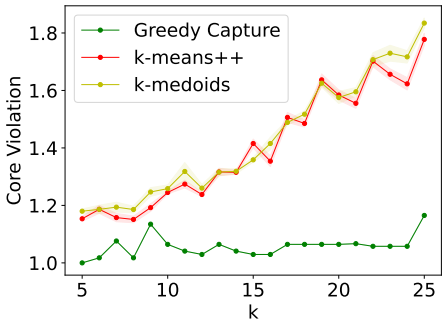


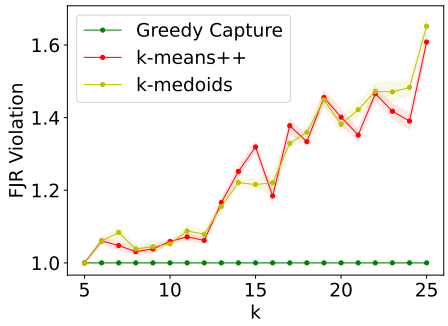
Figure 4: Diabetes dataset

a deviate by forming a cluster, each of them improves from infinite loss to a finite loss. Therefore, these algorithms do not provide a finite approximation to the core or FJR.

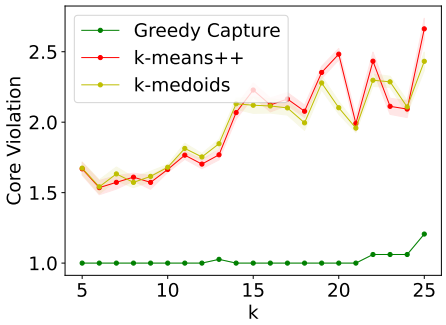
(a) Core violation, average loss



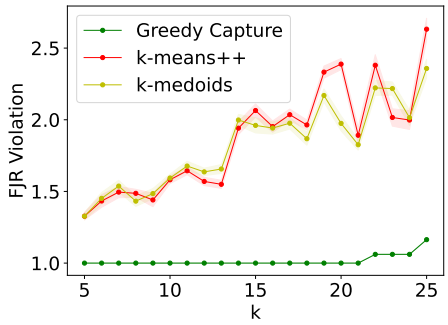
(b) FJR violation, average loss



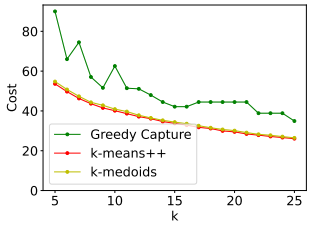
(c) Core violation, maximum loss



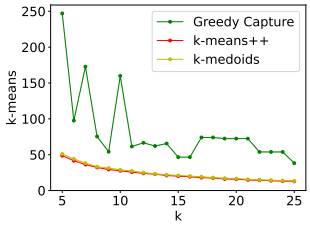
(d) FJR violation, maximum loss



(e) Avg within-cluster distance



(f) k-means objective



(g) k-medoids objective

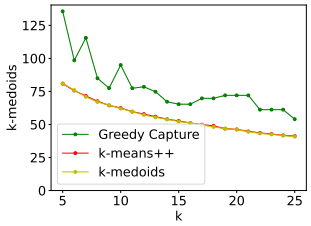


Figure 5: Iris dataset

https://doi.org/10.52202/079017-0605

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

https://doi.org/10.52202/079017-0605

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification:

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

https://doi.org/10.52202/079017-0605

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

https://doi.org/10.52202/079017-0605

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.