Fairness, Semi-Supervised Learning, and More: A General Framework for Clustering with Stochastic Pairwise Constraints

Brian Brubach [1], **Darshan Chakrabarti** [2], **John P. Dickerson** [3], **Aravind Srinivasan** [3], **Leonidas Tsepenekas** [3]

[1] Wellesley College, [2] Carnegie Mellon University, [3] University of Maryland, College Park
bb100@wellesley.edu, darshanc@alumni.cmu.edu, {john, srin, ltsepene}@cs.umd.edu

Abstract

Metric clustering is fundamental in areas ranging from Combinatorial Optimization and Data Mining, to Machine Learning and Operations Research. However, in a variety of situations we may have additional requirements or knowledge, distinct from the underlying metric, regarding which pairs of points should be clustered together. To capture and analyze such scenarios, we introduce a novel family of *stochastic pairwise constraints*, which we incorporate into several essential clustering objectives (radius/median/means). Moreover, we demonstrate that these constraints can succinctly model an intriguing collection of applications, including among others *Individual Fairness* in clustering and *Must-link* constraints in semi-supervised learning. Our main result consists of a general framework that yields approximation algorithms with provable guarantees for important clustering objectives, while at the same time producing solutions that respect the stochastic pairwise constraints. Furthermore, for certain objectives we devise improved results in the case of Must-link constraints, which are also the best possible from a theoretical perspective. Finally, we present experimental evidence that validates the effectiveness of our algorithms.

1 Introduction

In a generic metric clustering problem, there is a set of points $\mathcal{C}$, requiring service from a set of locations $\mathcal{F}$, where both $\mathcal{C}$ and $\mathcal{F}$ are embedded in some metric space. The sets $\mathcal{C}, \mathcal{F}$ do not need to be disjoint, and we may very well have $\mathcal{C} = \mathcal{F}$. The goal is then to choose a set of locations $S \subseteq \mathcal{F}$, where S might have to satisfy additional problem-specific requirements and an assignment $\phi : \mathcal{C} \mapsto S$, such that a metric-related objective function over $\mathcal{C}$ is minimized.

However, in a variety of situations there may be external and metric-independent constraints imposed on ϕ, regarding which pairs of points $j, j' \in \mathcal{C}$ should be clustered together, i.e., constraints forcing a linkage $\phi(j) = \phi(j')$. In this work, we generalize this deterministic requirement, by introducing a novel family of *stochastic pairwise constraints*. Our input is augmented with multiple sets P_q of pairs of points from $\mathcal{C}$ ($P_q \subseteq \binom{\mathcal{C}}{2}$ for each q), and values $\psi_q \in [0, 1]$. Given these, we ask for a randomized solution, which ensures that in expectation at most $\psi_q |P_q|$ pairs of P_q are separated in the returned assignment. In Sections 1.1-1.2, we discuss how these constraints have enough expressive power to capture a wide range of applications such as extending the notion of *Individual Fairness* from classification to clustering, and incorporating elements of Semi-Supervised clustering.

Another constraint we address is when $\mathcal{C} = \mathcal{F}$ and every chosen point $j \in S$ must serve as an exemplar of the cluster it defines (the set of all points assigned to it). The subtle difference here, is that an exemplar point should be assigned to its own cluster, i.e., $\phi(j) = j$ for all $j \in S$. This constraint is highly relevant in strict classification settings, and is trivially satisfied in vanilla clustering variants where each point is always assigned to its nearest point in S. However, the presence of additional requirements on ϕ makes its satisfaction more challenging. Previous literature, especially in the context of fairness in clustering (Anderson et al. 2020; Esmaeili et al. 2020; Bera et al. 2019; Bercea et al. 2019), does not address this issue, but in our framework we explicitly offer the choice of whether or not to enforce it.

1.1 Formal Problem Definitions

We are given a set of points $\mathcal{C}$ and a set of locations $\mathcal{F}$, in a metric space characterized by the distance function $d : \mathcal{C} \cup \mathcal{F} \times \mathcal{C} \cup \mathcal{F} \mapsto \mathbb{R}_{\geq 0}$, which satisfies the triangle inequality. Moreover, the input includes a concise description of a set $\mathcal{L} \subseteq 2^{\mathcal{F}}$, that captures the allowable configurations of location openings. The goal of all problems we consider, is to find a set $S \subseteq \mathcal{F}$, with $S \in \mathcal{L}$, and an efficiently-sampleable distribution $\mathcal{D}$ over assignments $\mathcal{C} \mapsto S$, such that for a randomly drawn $\phi \sim \mathcal{D}$ we have: (i) an objective function being minimized, and (ii) depending on the variant at hand, further constraints are satisfied by ϕ. We study two types of additional constraints imposed on ϕ.

- *Stochastic Pairwise Constraints* (SPC): We are given a family of sets $\mathcal{P} = \{P_1, P_2, \ldots\}$, where each $P_q \subseteq \binom{\mathcal{C}}{2}$ is a set of pairs of points from $\mathcal{C}$, and a sequence $\psi = (\psi_1, \psi_2, \ldots)$ with $\psi_q \in [0, 1]$. We then want $\sum_{\{j,j'\} \in P_q} \Pr_{\phi \sim \mathcal{D}}[\phi(j) \neq \phi(j')] \leq \psi_q |P_q|$, $\forall P_q \in \mathcal{P}$.
- *Centroid Constraint* (CC): When this is imposed on any of our problems, we must first have $\mathcal{C} = \mathcal{F}$. In addition, we should ensure that $\Pr_{\phi \sim \mathcal{D}}[\phi(i) = i] = 1$ for all $i \in S$.

Special Cases of SPC: When each $P_q \in \mathcal{P}$ has $|P_q| = 1$, we get two interesting resulting variants.

- $\psi_q = 0, \forall q$: For each $P_q = \{\{j, j'\}\}$ we must ensure that j, j' have $\Pr_{\phi\sim\mathcal{D}}[\phi(j) = \phi(j')] = 1$, and hence we call such constraints *must-link* (ML). Further, since there is no actual randomness involved in these constraints, we assume w.l.o.g. that $|\mathcal{D}| = 1$, and only solve for a single $\phi : \mathcal{C} \mapsto S$ instead of a distribution over assignments.
- $\psi_q \geq 0, \forall q$: For each $P_q = \{\{j, j'\}\}$ we must have $\Pr_{\phi\sim\mathcal{D}}[\phi(j) \neq \phi(j')] \leq \psi_q$, and therefore we call this constraint *probabilistic-bounded-separation* (PBS).

The objective functions we consider are:

- **$\mathcal{L}$-center/$\mathcal{L}$-supplier:** Here we aim for the minimum τ ("radius"), such that $\Pr_{\phi\sim\mathcal{D}}[d(\phi(j), j) \leq \tau] = 1$ for all $j \in \mathcal{C}$. Further, in the $\mathcal{L}$-center setting, we have $\mathcal{C} = \mathcal{F}$.
- **$\mathcal{L}$-median ($p = 1$)/$\mathcal{L}$-means ($p = 2$):** Here the goal is to minimize $(\sum_{j\in\mathcal{C}} \mathbb{E}_{\phi\sim\mathcal{D}}[d(\phi(j), j)^p])^{1/p}$.

There are four types of location specific constraints that we study in this paper. In the first, which we call *unrestricted*, $\mathcal{L} = 2^{\mathcal{F}}$ and hence any set of locations can serve our needs. In the second, we have $\mathcal{L} = \{S \subseteq \mathcal{F} \mid |S| \leq k\}$ for some given positive integer k. This variant gives rise to the popular *k-center/k-supplier/k-median/k-means* objectives. In the third, we assume that each $i \in \mathcal{F}$ has an associated cost $w_i \geq 0$, and for some given $W \geq 0$ we have $\mathcal{L} = \{S \subseteq \mathcal{F} \mid \sum_{i\in S} w_i \leq W\}$. In this case the resulting objectives are called *knapsack-center/knapsack-supplier/knapsack-median/knapsack-means*. Finally, if the input also consists of a matroid $\mathcal{M} = (\mathcal{F}, \mathcal{I})$, where $\mathcal{I} \subseteq 2^{\mathcal{F}}$ the family of independent sets of $\mathcal{M}$, we have $\mathcal{L} = \mathcal{I}$, and the objectives are called *matroid-center/matroid-supplier/matroid-median/matroid-means*.

To specify the problem at hand, we use the notation **Objective-List of Constraints**. For instance, **$\mathcal{L}$-means-SPC-CC** is the $\mathcal{L}$-means problem, where we additionally impose the SPC and CC constraint. We could also further specify $\mathcal{L}$, by writing for example **k-means-SPC-CC**. Moreover, observe that when no constraints on ϕ are imposed, we get the vanilla version of each objective, where the lack of any stochastic requirement implies that the distribution $\mathcal{D}$ once more has support of size 1, i.e., $|\mathcal{D}| = 1$, and we simply solve for just an assignment $\phi : \mathcal{C} \mapsto S$.

1.2 Motivation

In this section we present a wide variety of applications, that can be effectively modeled by our newly introduced SPCs.

Fairness: With machine-learning clustering approaches being ubiquitous in everyday decision making, a natural question that arises and has recently captured the interest of the research community, is how to avoid clusterings which perpetuate existing social biases.

The *individual* approach to fair classification introduced in the seminal work of (Dwork et al. 2012) assumes that we have access to an additional metric, separate from the feature space, which captures the true "similarity" between points (or some approximation of it). This similarity metric may be quite different from the feature space d (e.g., due to redundant encodings of features such as race), and its ultimate purpose is to help "treat similar candidates similarly". Note now that the PBS constraint introduced earlier, can succinctly capture this notion. For two points j, j', we may have $\psi_{j,j'} \in [0, 1]$ as an estimate of their true similarity (with 0 indicating absolute identity), and interpret unfair treatment as deterministically separating these two points in the final solution. Hence, a fair randomized approach would cluster j and j' apart with probability at most $\psi_{j,j'}$.

A recent work that explores individual fairness in clustering is (Anderson et al. 2020). Using our notation, the authors in that paper require a set $S \in \mathcal{L}$, and for all $j \in \mathcal{C}$ a distribution ϕ_j that assigns j to each $i \in S$ with probability $\phi_{i,j}$. Given that, they seek solutions that minimize the clustering objectives, while ensuring that for given pairs j, j', their assignment distributions are statistically similar based on some metric D that captures distributional proximity (e.g., total variation and KL-divergence). In other words, they interpret individual fairness as guaranteeing $D(\phi_j, \phi_{j'}) \leq p_{j,j'}$ for all provided pairs $\{j, j'\}$ and values $p_{j,j'}$. Although this work is interesting in terms of initiating the discussion on individual fair clustering, it has a significant modeling issue. To be more precise, suppose that for j, j' the computed $\phi_j, \phi_{j'}$ are both the uniform distribution over S. Then according to that paper's definition a fair solution is achieved. However, the actual probability of placing j, j' in different clusters (hence treating them unequally) is almost 1 if we do not consider any correlation between ϕ_j and $\phi_{j'}$. On the other hand, our definition which instead asks for a distribution $\mathcal{D}$ over assignments $\phi : \mathcal{C} \mapsto S$, always provides meaningful results, since it bounds the quantity that really matters, i.e., the probability of separating j and j' in a random $\phi \sim \mathcal{D}$.

Another closely related work in the context of individual fair clustering is (Brubach et al. 2020). The authors of that paper study a special case of PBS, where for each $j, j' \in \mathcal{C}$ we have $\psi_{j,j'} = d(j, j')/\tau^*$, with τ^* the objective value of the optimal solution. They then provide a $\log k$-approximation for the k-center objective under the above constraints. Compared to that, our framework 1) can handle the median and means objectives as well, 2) can incorporate further requirements on the set of chosen locations (unrestricted/knapsack/matroid), 3) allows for arbitrary values for the separation probabilities $\psi_{j,j'}$, and 4) provides smaller constant-factor approximations for the objective functions.

Semi-Supervised Clustering: A common example of ML constraints is in semi-supervised learning (Wagstaff et al. 2001; Basu, Davidson, and Wagstaff 2008; Van Engelen and Hoos 2020). There we assume that pairs of points have been annotated (e.g., by human experts) with additional information about their similarity (Zhang and Yan 2007), or that some points may be explicitly labeled (Zhu, Ghahramani, and Lafferty 2003; Bilenko, Basu, and Mooney 2004), allowing pairwise relationships to be inferred. Then these extra requirements are incorporated in the algorithmic setting in the form of ML constraints. Further, our SPCs capture the scenario where the labeler generating the constraints is assumed to make some bounded number of errors (by associating each labeler with a set P_q and an accuracy ψ_q), and also allow for multiple labelers (e.g., from crowdsourcing labels) with different accuracies. Similar settings have been studied by (Chang et al. 2017; Luo et al. 2018) as well.

OTU Clustering: The field of metagenomics involves analyzing environmental samples of genetic material to explore the vast array of bacteria that cannot be analyzed through traditional culturing approaches. A common practice in the study of these microbial communities is the *de novo* clustering of genetic sequences (e.g., 16S rRNA marker gene sequences) into Operational Taxonomic Units (OTUs) (Edgar 2013; Westcott and Schloss 2017), that ideally correspond to clusters of closely related organisms. One of the most ubiquitous approaches to this problem involves taking a fixed radius (e.g., 97% similarity based on string alignment (Stackebrandt and Goebel 1994)) and outputting a set of center sequences, such that all points are assigned to a center within the given radius (Edgar 2013; Ghodsi, Liu, and Pop 2011). In this case, we do not know the number of clusters a priori, but we may be able to generate pairwise constraints based on a distance/similarity threshold as in (Westcott and Schloss 2017) or reference databases of known sequences. Thus, the "unrestricted" variant of our framework is appropriate here, where the number of clusters should be discovered, but radius and pairwise information is known or estimated. Other work in this area has considered *conspecific probability*, a given probability that two different sequences belong to the same species (easily translated to PBS) and *adverse triplets*; sets of ML constraints that cannot all be satisfied simultaneously (an appropriate scenario for a set P_q as defined in Section 1.1)(Edgar 2018).

Community Preservation: There are scenarios where clustering a *group/community* of points together is beneficial for the coherence and quality of the final solution. Examples of this include assigning students to schools such that students living in the same neighborhood are not placed into different schools, vaccinating people with similar demographics in a community (e.g., during a pandemic), and drawing congressional districts with the intent to avoid the practice of gerrymandering. Given such a group of points G, we let $P_G = \binom{G}{2}$, and set a tolerance parameter $\psi_G \in [0, 1]$. Then, our SPCs will make sure that in expectation at most $\psi_G|G|$ pairs from G are separated, and thus a $(1-\psi_G)$ fraction of the community is guaranteed to be preserved. Finally, Markov's inequality also gives tail bounds on this degree of separation for all G.

1.3 Our Contribution

In Section 2 we present our main algorithmic result, which is based on the two-step approach of (Bercea et al. 2019; Chierichetti et al. 2017). Unlike previous works utilizing this technique, the most serious technical difficulty we faced was not in the LP-rounding procedure, but rather in the formulation of an appropriate assignment-LP relaxation. Letting $P_{\mathcal{L}}$ be any problem in $\{\mathcal{L}$-center, $\mathcal{L}$-supplier, $\mathcal{L}$-median, $\mathcal{L}$-means$\}$ and $\mathcal{L}$ any of the four location settings, we get:

Theorem 1. *Let τ^* the optimal value of a $P_{\mathcal{L}}$-**SPC** instance, and ρ the best approximation ratio for $P_{\mathcal{L}}$. Then our algorithm chooses a set $S_{P_{\mathcal{L}}}$ and constructs an appropriate distribution over assignments $\mathcal{D}$, such that $S_{P_{\mathcal{L}}} \in \mathcal{L}$, $\sum_{\{j,j'\}\in P_q} \Pr_{\phi\sim\mathcal{D}}[\phi(j) \neq \phi(j')] \leq 2\psi_q|P_q|\ \forall P_q \in \mathcal{P}$, and*

1. *$P_{\mathcal{L}}$ is $\mathcal{L}$-center$(\alpha = 1)$/$\mathcal{L}$-supplier$(\alpha = 2)$: Here we get $\Pr_{\phi\sim\mathcal{D}}[d(\phi(j), j) \leq (\alpha + \rho)\tau^*] = 1$, for all $j \in \mathcal{C}$.*
2. *$P_{\mathcal{L}}$ is $\mathcal{L}$-median$(p = 1)$/$\mathcal{L}$-means$(p = 2)$: Here we get $(\sum_{j\in\mathcal{C}} \mathbb{E}_{\phi\sim\mathcal{D}}[d(\phi(j), j)^p])^{1/p} \leq (2 + \rho)\tau^*$.*

Finally, sampling a $\phi \sim \mathcal{D}$ can be done in polynomial time.

Given that the value ρ is a small constant for all variations of $P_{\mathcal{L}}$ that we consider, we see that our algorithmic framework gives indeed good near-optimal guarantees. Moreover, a tighter analysis when $\mathcal{L} = 2^{\mathcal{F}}$ yields the next result.

Theorem 2. *When $\mathcal{L} = 2^{\mathcal{F}}$, our algorithm has the same guarantees as those in Theorem 1, but this time the cost of the returned solution for $P_{unrestricted}$-**SPC** is at most τ^*.*

Although imposing no constraint on the set of chosen locations yields trivial problems in vanilla settings, the presence of SPCs makes even this variant NP-hard. Specifically, we show the following theorem (full version Appendix).

Theorem 3. *The problem **unrestricted-$\mathcal{O}$-SPC** is NP-hard, where $\mathcal{O} \in \{$center, supplier, median, means$\}$.*

In Section 3 we consider settings where each $j \in S$ must serve as an exemplar of its defining cluster. Hence, we incorporate the Centroid constraint in our problems. As mentioned earlier, previous work in the area of fair clustering had ignored this issue. Our first result follows.

Theorem 4. *Let τ^* the optimal value of a k-**center-SPC-CC** instance. Then our algorithm chooses $S_k \subseteq \mathcal{C}$ and constructs a distribution $\mathcal{D}$, such that sampling $\phi \sim \mathcal{D}$ can be done efficiently, $|S_k| \leq k$, $\sum_{\{j,j'\}\in P_q} \Pr_{\phi\sim\mathcal{D}}[\phi(j) \neq \phi(j')] \leq 2\psi_q|P_q|\ \forall P_q \in \mathcal{P}$, $\Pr_{\phi\sim\mathcal{D}}[d(\phi(j), j) \leq 3\tau^*] = 1$ for all $j \in \mathcal{C}$, and $\Pr_{\phi\sim\mathcal{D}}[\phi(i) = i] = 1$ for all $i \in S_k$.*

To address all objective functions under the Centroid constraint, we demonstrate (again in Section 3) a reassignment procedure that gives the following result.

Theorem 5. *Let λ the approximation ratio for the objective of $P_{\mathcal{L}}$-**SPC** achieved in Theorem 1. Then, our reassignment procedure applied to the solution produced by the algorithm mentioned in Theorem 1, gives an approximation ratio 2λ for $P_{\mathcal{L}}$-**SPC-CC**, while also preserving the SPC guarantees of Theorem 1 and satisfying the CC, when $\mathcal{L} = 2^{\mathcal{C}}$ or $\mathcal{L} = \{S' \subseteq \mathcal{C} : |S'| \leq k\}$ for some given positive integer k.*

As for ML constraints, since they are a special case of SPCs, our results for the latter also address the former. However, in Section 4 we provide improved approximation algorithms for a variety of problem settings with ML constraints. Our main result is summarized in the following theorem.

Theorem 6. *There exists a $2/3/3/3$-approximation algorithm for k-**center-ML/knapsack-center-ML/k-supplier-ML/knapsack-supplier-ML**. This algorithm is also the best possible in terms of the approximation ratio, unless $P = NP$. In addition, it satisfies without any further modifications the Centroid constraint.*

Although ML constraints have been extensively studied in the semi-supervised literature (Basu, Davidson, and Wagstaff 2008), to the extent of our knowledge we are the first to tackle them purely from a Combinatorial Optimization perspective, with the exception of (Davidson, Ravi, and Shamis 2010). This paper provides a $(1 + \epsilon)$ approximation for k-center-ML, but only in the restricted $k = 2$ setting.

1.4 Further Related Work

Clustering problems have been a longstanding area of research in Combinatorial Optimization, with all important settings being thoroughly studied (Hochbaum and Shmoys 1986; Gonzalez 1985; Harris et al. 2017; Byrka et al. 2017; Ahmadian et al. 2017; Chakrabarty and Negahbani 2019).

The work that initiated the study of fairness in clustering is (Chierichetti et al. 2017). That paper addresses a notion of demographic fairness, where points are given a certain color indicating some protected attribute, and then the goal is to compute a solution that enforces a fair representation of each color in every cluster. Further work on similar notions of demographic fairness includes (Bercea et al. 2019; Bera et al. 2019; Esmaeili et al. 2020; Huang, Jiang, and Vishnoi 2019; Backurs et al. 2019; Ahmadian et al. 2019).

Finally, a separation constraint similar to PBS is found in (Davidson, Ravi, and Shamis 2010). In that paper however, the separation is deterministic and also depends on the underlying distance between two points. Due to their stochastic nature, our PBS constraints allow room for more flexible solutions, and also capture more general separation scenarios, since the ψ_p values can be arbitrarily chosen.

1.5 An LP-Rounding Subroutine

We present an important subroutine developed by (Kleinberg and Tardos 2002), which we repeatedly use in our results, and call it **KT-Round**. Suppose we have a set of elements V, a set of labels L, and a set of pairs $E \subseteq \binom{V}{2}$. Consider the following Linear Program (LP).

$$\sum_{l \in L} x_{l,v} = 1, \qquad \forall v \in V \quad (1)$$

$$z_{e,l} \geq x_{l,v} - x_{l,w}, \qquad \forall e = \{v,w\} \in E,\ \forall l \in L \quad (2)$$

$$z_{e,l} \geq x_{l,w} - x_{l,v}, \qquad \forall e = \{v,w\} \in E,\ \forall l \in L \quad (3)$$

$$z_e = \frac{1}{2} \sum_{l \in L} z_{e,l}, \qquad \forall e = \{v,w\} \in E \quad (4)$$

$$0 \leq x_{l,v}, z_e, z_{e,l} \leq 1, \qquad \forall v \in V, \forall e \in E, \forall l \in L \quad (5)$$

Theorem 7. *(Kleinberg and Tardos 2002) Given a feasible solution (x, z) of (1)-(5), there exists a randomized rounding approach **KT-Round**(V, L, E, x, z), which in polynomial expected time assigns each $v \in V$ to a $\phi(v) \in L$, such that:*

1. $\Pr[\phi(v) \neq \phi(w)] \leq 2z_e,\ \ \forall e = \{v,w\} \in E$
2. $\Pr[\phi(v) = l] = x_{l,v}, \quad \forall v \in V,\ \forall l \in L$

2 A General Framework for Approximating Clustering Problems with SPCs

In this section we show how to achieve approximation algorithms with provable guarantees for **$\mathcal{L}$-center-SPC/$\mathcal{L}$-supplier-SPC/$\mathcal{L}$-median-SPC/$\mathcal{L}$-means-SPC** using a general two-step framework. At first, let $P_{\mathcal{L}}$ denote any of the vanilla versions of the objective functions we consider, i.e., $P_{\mathcal{L}} \in \{\mathcal{L}\text{-center}, \mathcal{L}\text{-supplier}, \mathcal{L}\text{-median}, \mathcal{L}\text{-means}\}$.

To tackle a $P_{\mathcal{L}}$-SPC instance, we begin by using on it any known ρ-approximation algorithm $A_{P_{\mathcal{L}}}$ for $P_{\mathcal{L}}$. This gives a set of locations $S_{P_{\mathcal{L}}}$ and an assignment $\phi_{P_{\mathcal{L}}}$, which yield an objective function cost of $\tau_{P_{\mathcal{L}}}$ for the corresponding $P_{\mathcal{L}}$ instance. In other words, we drop the SPC constraints from the $P_{\mathcal{L}}$-SPC instance, and simply treat it as its vanilla counterpart. Although $\phi_{P_{\mathcal{L}}}$ may not satisfy the SPCs, we are going to use the set $S_{P_{\mathcal{L}}}$ as our chosen locations. The second step in our framework would then consist of constructing the appropriate distribution over assignments. Toward that end, consider the following LP, where $P' = \cup_{P_q \in \mathcal{P}} P_q$.

$$\sum_{i \in S_{P_{\mathcal{L}}}} x_{i,j} = 1 \qquad \forall j \in \mathcal{C} \quad (6)$$

$$z_{e,i} \geq x_{i,j} - x_{i,j'} \qquad \forall e = \{j,j'\} \in P',\ \forall i \in S_{P_{\mathcal{L}}} \quad (7)$$

$$z_{e,i} \geq x_{i,j'} - x_{i,j} \qquad \forall e = \{j,j'\} \in P',\ \forall i \in S_{P_{\mathcal{L}}} \quad (8)$$

$$z_e = \frac{1}{2} \sum_{i \in S_{P_{\mathcal{L}}}} z_{e,i} \qquad \forall e \in P' \quad (9)$$

$$\sum_{e \in P_q} z_e \leq \psi_q |P_q| \qquad \forall P_q \in \mathcal{P} \quad (10)$$

$$0 \leq x_{i,j}, z_e, z_{e,i} \leq 1 \qquad \forall i \in S_{P_{\mathcal{L}}}, \forall j \in \mathcal{C}, \forall e \in P' \quad (11)$$

The variable $x_{i,j}$ can be interpreted as the probability of assigning point j to location $i \in S_{P_{\mathcal{L}}}$. To understand the meaning of the z variables, it is easier to think of the integral setting, where $x_{i,j} = 1$ iff j is assigned to i and 0 otherwise. In this case, $z_{e,i}$ is 1 for $e = \{j,j'\}$ iff exactly one of j and j' are assigned to i. Thus, z_e is 1 iff j and j' are separated. We will later show that in the fractional setting z_e is a lower bound on the probability that j and j' are separated. Therefore, constraint (6) simply states that every point must be assigned to a center, and given the previous discussion, (10) expresses the provided SPCs.

Depending on which exact objective function we optimize, we must augment LP (6)-(11) accordingly.

- **$\mathcal{L}$-center ($\alpha = 1$)/$\mathcal{L}$-supplier ($\alpha = 2$):** Here we assume w.l.o.g. that the optimal radius τ^*_{SPC} of the original $P_{\mathcal{L}}$-**SPC** instance is known. Observe that this value is always the distance between some point and some location, and hence there are only polynomially many alternatives for it. Thus, we execute our algorithm for each of those, and in the end keep the outcome that resulted in a feasible solution of minimum value. Given now τ^*_{SPC}, we add the following constraint to the LP.
$$x_{i,j} = 0,\ \forall i,j :\ d(i,j) > \tau_{P_{\mathcal{L}}} + \alpha \cdot \tau^*_{SPC} \qquad (12)$$
- **$\mathcal{L}$-median ($p = 1$)/$\mathcal{L}$-means ($p = 2$):** In this case, we augment the LP with the following objective function.
$$\min \sum_{j \in \mathcal{C}} \sum_{i \in S_{P_{\mathcal{L}}}} x_{i,j} \cdot d(i,j)^p \qquad (13)$$

The second step of our framework begins by solving the appropriate LP for each variant of $P_{\mathcal{L}}$, in order to acquire a fractional solution $(\bar{x}, \bar{z})$ to that LP. Finally, the distribution $\mathcal{D}$ over assignments $\mathcal{C} \mapsto S_{P_{\mathcal{L}}}$ is constructed by running **KT-Round**$(\mathcal{C}, S_{P_{\mathcal{L}}}, P', \bar{x}, \bar{z})$. Notice that this will yield an assignment $\phi \sim \mathcal{D}$, where $\mathcal{D}$ results from the internal randomness of **KT-Round**. Our overall approach for solving $P_{\mathcal{L}}$-**SPC** is presented in Algorithm 1.

Algorithm 1: Approximating $P_{\mathcal{L}}$**-SPC**

$(S_{P_{\mathcal{L}}}, \phi_{P_{\mathcal{L}}}) \leftarrow A_{P_{\mathcal{L}}}(\mathcal{C}, \mathcal{F}, \mathcal{L})$;
Solve LP (6)-(11) with (12) for $\mathcal{L}$**-center/**$\mathcal{L}$**-supplier**, and with (13) for $\mathcal{L}$**-median/**$\mathcal{L}$**-means**, and get a fractional solution $(\bar{x}, \bar{z})$;
$\phi \leftarrow$ **KT-Round**$(\mathcal{C}, S_{P_{\mathcal{L}}}, P', \bar{x}, \bar{z})$;

Theorem 8. *Let τ^*_{SPC} the optimal value of the given $P_{\mathcal{L}}$-**SPC** instance. Then Algorithm 1 guarantees that $S_{P_{\mathcal{L}}} \in \mathcal{L}$, $\sum_{\{j,j'\}\in P_q} \Pr_{\phi\sim\mathcal{D}}[\phi(j) \neq \phi(j')] \leq 2\psi_q |P_q| \ \forall P_q \in \mathcal{P}$ and*

1. *$P_{\mathcal{L}}$ is $\mathcal{L}$-center$(\alpha = 1)$/$\mathcal{L}$-supplier$(\alpha = 2)$: Here we get $\Pr_{\phi\sim\mathcal{D}}[d(\phi(j), j) \leq \alpha \cdot \tau^*_{SPC} + \tau_{P_{\mathcal{L}}}] = 1$, for all $j \in \mathcal{C}$.*
2. *$P_{\mathcal{L}}$ is $\mathcal{L}$-median$(p = 1)$/$\mathcal{L}$-means$(p = 2)$: Here we get $(\sum_{j\in\mathcal{C}} \mathbb{E}_{\phi\sim\mathcal{D}}[d(\phi(j), j)^p])^{1/p} \leq 2\tau^*_{SPC} + \tau_{P_{\mathcal{L}}}$.*

Since $P_{\mathcal{L}}$ is a less restricted version of $P_{\mathcal{L}}$-SPC, the optimal solution value $\tau^*_{P_{\mathcal{L}}}$ for $P_{\mathcal{L}}$ in the original instance where we dropped the SPCs, should satisfy $\tau^*_{P_{\mathcal{L}}} \leq \tau^*_{SPC}$. Therefore, because $A_{P_{\mathcal{L}}}$ is a ρ-approximation algorithm for $P_{\mathcal{L}}$, we get $\tau_{P_{\mathcal{L}}} \leq \rho \cdot \tau^*_{SPC}$. The latter implies the following.

Corollary 9. *The approximation ratio achieved through Algorithm 1 is $(\rho + 1)$ for $\mathcal{L}$-center-SPC, and $(\rho + 2)$ for $\mathcal{L}$-supplier-SPC/$\mathcal{L}$-median-SPC/$\mathcal{L}$-means-SPC.*

Tighter analysis for the *unrestricted* **($\mathcal{L} = 2^{\mathcal{F}}$) case:** For this case, a more careful analysis leads to the following.

Theorem 10. *When $\mathcal{L} = 2^{\mathcal{F}}$, Algorithm 1 achieves an objective value of at most τ^*_{SPC} for all objectives we study (center/supplier/median/means).*

3 Addressing the Centroid Constraint

In this section we present results that incorporate the Centroid Constraint (CC) to a variety of the settings we study. Moreover, recall that for this case $\mathcal{C} = \mathcal{F}$, and hence the supplier objective reduces to the center one.

3.1 Approximating k-center-SPC-CC

Our approach for solving this problem heavily relies on Algorithm 1 with two major differences.

The first difference compared to Algorithm 1 lies in the approximation algorithm A_k used to tackle k-center. For k-center there exists a 2-approximation which given a target radius τ, it either returns a solution where each $j \in \mathcal{C}$ gets assigned to a location i_j with $d(i_j, j) \leq 2\tau$, or outputs an "infeasible" message, indicating that there exists no solution of radius τ ((Hochbaum and Shmoys 1986)).

Recall now that w.l.o.g. the optimal radius τ^*_C for the k-center-SPC-CC instance is known. In the first step of our framework we will use the variant of A_k mentioned earlier with τ^*_C as its target radius, and get a set of chosen locations S_k. The second step is then the same as in Algorithm 1, with the addition of the next constraint to the assignment LP:

$$x_{i,i} = 1, \ \forall i \in S_k \tag{14}$$

The overall process is presented in Algorithm 2.

Algorithm 2: Approximating k**-center-SPC-CC**

$(S_k, \phi_k) \leftarrow A_k(\mathcal{C}, \mathcal{F}, \mathcal{L}, \tau^*_C)$;
Solve LP (6)-(11) with (12), (14) and S_k as the chosen locations, and get a solution $(\bar{x}, \bar{z})$;
$\phi \leftarrow$ **KT-Round**$(\mathcal{C}, S_k, P', \bar{x}, \bar{z})$;

Algorithm 3: Approximating $P_{\mathcal{L}}$**-SPC-CC**

Run Algorithm 1 to solve $P_{\mathcal{L}}$-SPC, and get $S \subseteq \mathcal{C}$ and an assignment $\phi : \mathcal{C} \mapsto S$ in return;
while *there exists $i \in S$ with $\phi(i) \neq i$* **do**
  $S \leftarrow S \setminus \{i\}$;
  $i' \leftarrow \arg\min_{j\in\mathcal{C}:\phi(j)=i} d(i, j)$;
  $S \leftarrow S \cup \{i'\}$;
  for *all $j \in \mathcal{C}$ with $\phi(j) = i$* **do**
    $\phi(j) \leftarrow i'$;

Theorem 11. *Let τ^*_C the optimal value of the given k-**center-SPC-CC** instance, and $\mathcal{D}$ the distribution over assignments given by **KT-Round**. Then Algorithm 2 guarantees $|S_k| \leq k$, $\sum_{\{j,j'\}\in P_q} \Pr_{\phi\sim\mathcal{D}}[\phi(j) \neq \phi(j')] \leq 2\psi_q |P_q| \ \forall P_q \in \mathcal{P}$, $\Pr_{\phi\sim\mathcal{D}}[d(\phi(j), j) \leq 3\tau^*_C] = 1$ for all $j \in \mathcal{C}$, and $\Pr_{\phi\sim\mathcal{D}}[\phi(i) = i] = 1$ for all $i \in S_k$.*

3.2 A Reassignment Step for the Unrestricted and k-Constrained Location Setting

We now demonstrate a reassignment procedure that can be used to correct the output of Algorithm 1, in a way that satisfies the CC. Again, let $P_{\mathcal{L}}$ be any of the vanilla objective functions, and consider Algorithm 3.

Theorem 12. *Let λ the approximation ratio of Algorithm 1 for $P_{\mathcal{L}}$-**SPC** with respect to the objective function. Then, Algorithm 3 gives an approximation ratio 2λ for the objective of $P_{\mathcal{L}}$-**SPC-CC**, while satisfying the CC and preserving the guarantees of Algorithm 1 on SPCs, when $\mathcal{L} = 2^{\mathcal{C}}$ or $\mathcal{L} = \{S' \subseteq \mathcal{C} : \ |S'| \leq k\}$ for some integer k.*

4 Improved Results for Problems with Must-Link Constraints

Since must-link constraints (ML) are a special case of SPCs, Algorithm 1 provides approximation results for the former as well (also note that due to $\psi_p = 0 \ \forall p$, we have no pairwise constraint violation when using Algorithm 1 purely for ML). However, in this section we demonstrate how we can get improved approximation guarantees for some of the problems we consider. Specifically, we provide a $2/3/3/3$-approximation for k-center-ML/knapsack-center-ML/k-supplier-ML/knapsack-supplier-ML, which constitutes a clear improvement over the $3/4/5/5$-approximation, given when Algorithm 1 is executed using the best approximation algorithm for the corresponding vanilla variant.

First of all, recall that in the ML case we are only looking for a set of locations S and an assignment $\phi : \mathcal{C} \mapsto S$, and not for a distribution over assignments. Also, notice that the

Algorithm 4: Approximating ML Constraints

1 $C \leftarrow \emptyset$, $S \leftarrow \emptyset$;
2 Initially all $C_1, C_2, \ldots, C_t$ are considered uncovered;
3 **while** *there exists an uncovered* C_q **do**
4 Pick an uncovered C_q;
5 Pick an arbitrary point $j_q \in C_q$;
6 $C \leftarrow C \cup \{j_q\}$;
7 C_q and all neighboring cliques C_p of it, are now considered covered;
8 **for** *all* $j_q \in C$ **do**
9 **if** *k-center/k-supplier* **then**
10 $i_q \leftarrow \arg\min_{i \in \mathcal{F}} d(i, j_q)$;
11 $S \leftarrow S \cup \{i_q\}$;
12 **if** *knapsack-center/knapsack-supplier* **then**
13 $i_q \leftarrow \arg\min_{i \in \mathcal{F}: d(i,j_q) \leq \tau^*} w_i$;
14 $S \leftarrow S \cup \{i_q\}$;
15 **for** *all* $j \in \mathcal{C}$ **do**
16 Let $j_q \in C$ the point whose clique C_q covered j's clique in the first while loop;
17 $\phi(j) \leftarrow i_q$;

must-link relation is transitive. If for j, j' we want $\phi(j) = \phi(j')$, and for j', j'' we also require $\phi(j') = \phi(j'')$, then $\phi(j) = \phi(j'')$ is necessary as well. Given that, we view the input as a partition $C_1, C_2, \ldots, C_t$ of the points of $\mathcal{C}$, where all points in C_q, with $q \in \{1, \ldots, t\}$, must be assigned to the same location of S. We call each part C_i of this partition a clique. Finally, for the problems we study, we can once more assume w.l.o.g. that the optimal radius τ^* is known.

Definition 13. Two cliques $C_q,\ C_p$ are called **neighboring** if $\forall j \in C_q,\ \forall j' \in C_p$ we have $d(j, j') \leq 2\tau^*$.

Algorithm 4 captures k-center-ML, knapsack-center-ML, k-supplier-ML and knapsack-supplier-ML at once, yielding improved approximations for each of them.

Theorem 14. *Algorithm 4 is a $2/3/3/3$-approximation algorithm for k-**center-ML/knapsack-center-ML/k-supplier-ML/knapsack-supplier-ML**.*

Observation 15. Algorithm 4 is a $2/3$-approximation for k-center-ML-CC/knapsack-center-ML-CC. This directly follows from steps 10, 13 and 17 of it.

Observation 16. Due to known hardness results for the vanilla version of the corresponding problems (Hochbaum and Shmoys 1986), Algorithm 4 gives the best possible approximation ratios, assuming that $P \neq NP$.

5 Experimental Evaluation

We implement our algorithms in Python 3.8 and run our experiments on AMD Opteron 6272 @ 2.1 GHz with 64 cores and 512 GB 1333 MHz DDR3 memory. We focus on fair clustering applications with PBS constraints and evaluate against the most similar prior work. Comparing to (Anderson et al. 2020) using k-means-PBS, shows that our algorithm violates fewer constraints while achieving a comparable cost of fairness. Similarly, our comparison with (Brubach et al. 2020) using k-center-PBS-CC (that prior algorithm also satisfies the CC constraint) reveals that we are better able to balance fairness constraints and the objective value. Our code is publicly available at https://github.com/chakrabarti/pairwise_constrained_clustering.

	k	4	6	8	10
Adult	Alg-1	2.27	4.73	12.53	21.81
	ALG-IF	84.87	91.76	100.00	100.00
Bank	Alg-1	0.16	0.44	0.34	0.54
	ALG-IF	55.84	71.48	92.85	99.93
Credit	Alg-1	1.34	2.58	9.03	14.76
	ALG-IF	80.25	100.00	100.00	100.00

Table 1: Percentage of constraints that are violated on average for metric F_2

	k	4	6	8	10
Adult	Alg-1	1.88	2.41	3.09	3.48
	ALG-IF	1.88	2.38	3.21	3.44
Bank	Alg-1	2.34	3.28	4.09	4.62
	ALG-IF	2.36	3.34	4.67	4.93
Credit	Alg-1	1.82	2.12	2.46	2.71
	ALG-IF	1.80	2.20	2.43	2.66

Table 2: Cost of fairness for metric F_2

Datasets: We use 3 datasets from the UCI ML Repository (Dua and Graff 2017): **(1)** Bank-4,521 points (Moro, Cortez, and Rita 2014), **(2)** Adult-32,561 points (Kohavi 1996), and **(3)** Creditcard-30,000 points (Yeh and Lien 2009).

Algorithms: In all of our experiments, $\mathcal{C} = \mathcal{F}$ at first. When solving k-means-PBS, we use Lloyd's algorithm in the first step of Algorithm 1 and get a set of points L. The set of chosen locations S is constructed by getting the nearest point in $\mathcal{C}$ for every point of L. This is exactly the approach used in (Anderson et al. 2020), where their overall algorithm is called ALG-IF. To compare Algorithm 1 to ALG-IF, we use independent sampling for ALG-IF, in order to fix the assignment of each $j \in \mathcal{C}$ to some $i \in S$, based on the distribution ϕ_j produced by ALG-IF. For k-center-PBS-CC, we use Algorithm 2 with a binary search to compute τ^*_C.

Fairness Constraints: We consider three similarity metrics (F_1, F_2, F_3) for generating PBS constraints. We use F_1 for k-center-PBS-CC and F_2, F_3 for k-means-PBS. F_1 is the metric used for fairness in the simulations of (Brubach et al. 2020) and F_2, F_3 are the metrics used in the experimental evaluation of the algorithms in (Anderson et al. 2020).

F_1 involves setting the separation probability between a pair of points j and j' to $d(j, j')/R_{Scr}$ if $d(j, j') \leq R_{Scr}$, where R_{Scr} is the radius given by running the Scr algorithm (Mihelic and Robic 2005) on the provided input.

F_2 is defined so that the separation probability between a pair j,j' is given by $d(j, j')$, scaled linearly to ensure all such probabilities are in $[0, 1]$. Adopting the approach taken by (Anderson et al. 2020) when using this metric, we only consider pairwise constraints between each j and its closest m neighbors. For our experiments, we set $m = 100$.

	k	4	6	8	10
Adult	Alg-1	0.14	0.25	0.58	0.77
	ALG-IF	7.98	7.07	8.90	9.42
Bank	Alg-1	0.02	0.16	0.20	0.50
	ALG-IF	4.25	5.09	5.58	6.37
Credit	Alg-1	0.00	0.07	0.21	0.25
	ALG-IF	0.97	3.80	4.17	3.90

Table 3: Percentage of constraints that are violated on average for metric F_3

	k	4	6	8	10
Adult	Alg-1	1.13	1.20	1.23	1.24
	ALG-IF	1.13	1.20	1.23	1.23
Bank	Alg-1	1.22	1.36	1.41	1.44
	ALG-IF	1.24	1.41	1.46	1.54
Credit	Alg-1	1.12	1.11	1.10	1.10
	ALG-IF	1.10	1.09	1.10	1.11

Table 4: Cost of fairness for metric F_3

Again in order to compare our Algorithm 1 with (Anderson et al. 2020), we need the metric F_3. For any $j \in \mathcal{C}$, let r_j the minimum distance such that $|j' \in \mathcal{C} : d(j, j') \leq r_j| \geq |\mathcal{C}|/k$. Then the separation probability between j and any j' such that $d(j, j') \leq r_j$, is set to $d(j, j')/r_j$.

Implementation Details: As performed in (Anderson et al. 2020; Brubach et al. 2020), we uniformly sample N points from each dataset and run all algorithms on those sets, while only considering a subset of the numerical attributes and normalizing the features to have zero mean and unit variance. In our comparisons with (Anderson et al. 2020) we use $N = 1000$, while in our comparisons with (Brubach et al. 2020) N is set to 250. For the number of clusters k, we study the values $\{4, 6, 8, 10\}$ when comparing to (Anderson et al. 2020), and $\{10, 20, 30, 40, 50, 60\}$ when comparing to (Brubach et al. 2020) (theoretically Algorithm 2 is better for larger k). Finally, to estimate the empirical separation probabilities and the underlying objective function cost, we run 5000 trials for each randomized assignment procedure, and then compute averages for the necessary performance measures we are interested in.

Comparison with (Anderson et al. 2020): In Tables 1 and 3, we show what percentage of fairness constraints are violated by ALG-IF and our algorithm, for the fairness constraints induced by F_2 and F_3, allowing for an $\epsilon = 0.05$ threshold on the violation of a separation probability bound; we only consider a pair's fairness constraint to be violated if the empirical probability of them being separated exceeds that set by the fairness metric by more than ϵ. It is clear that our algorithm outperforms ALG-IF consistently across different values of k, different datasets, and both types of fairness constraints considered by (Anderson et al. 2020).

In order to compare the objective value achieved by both algorithms, we first compute the average connection costs over the 5000 runs. Since the cost of the clustering returned by Lloyd's algorithm contributes to both Algorithm 1 and ALG-IF, we utilize that as an approximation of the cost of fairness. In other words, we divide the objective value of the final solutions by the cost of the clustering produced by Lloyd, and call this quantity cost of fairness. The corresponding comparisons are presented in Tables 2, 4. The cost of fairness for both algorithms is very similar, demonstrating a clear advantage of Algorithm 1, since it dominates ALG-IF in the percentage of fairness constraints violated.

	k	10	20	30	40	50	60
Adult	Alg-2	.01	.01	.01	.02	.02	.04
	Alg-F(A)	.00	.00	.00	.00	.00	.00
	Alg-F(B)	.19	.18	.23	.30	.27	.30
Bank	Alg-2	.00	.01	.01	.06	.09	.03
	Alg-F(A)	.00	.00	.00	.00	.00	.00
	Alg-F(B)	.18	.20	.16	.15	.20	.23
Credit	Alg-2	.00	.01	.01	.01	.01	.02
	Alg-F(A)	.00	.00	.00	.00	.00	.00
	Alg-F(B)	.03	.05	.05	.05	.08	.08

Table 5: Percentage of constraints that are violated on average for metric F_1

	k	10	20	30	40	50	60
Adult	Alg-2	.39	.28	.23	.20	.17	.16
	Alg-F(A)	.54	.48	.46	.46	.43	.42
	Alg-F(B)	.31	.24	.17	.17	.12	.14
Bank	Alg-2	.17	.11	.08	.07	.06	.05
	Alg-F(A)	.21	.20	.17	.16	.14	.13
	Alg-F(B)	.12	.07	.06	.05	.04	.03
Credit	Alg-2	.38	.29	.25	.24	.21	.19
	Alg-F(A)	.45	.45	.43	.42	.41	.41
	Alg-F(B)	.28	.25	.21	.20	.18	.17

Table 6: Objective achieved for metric F_1

Comparison with (Brubach et al. 2020): In Table 5 we show what percentage of fairness constraints are violated by the algorithm of (Brubach et al. 2020) (named Alg-F) and Algorithm 2, using an $\epsilon = 0$; if the empirical probability of separation of a pair exceeds the bound set by the fairness metric by any amount, it is considered a violation. We run ALG-F with two different choices of the scale parameter used in that prior work: $\frac{1}{R_{Scr}}$ (Alg-F(A)) and $\frac{16}{R_{Scr}}$ (Alg-F(B)), where R_{Scr} is the value achieved using the Scr algorithm. The reason for doing so is that (Brubach et al. 2020) consider multiple values for the separation probabilities, and we wanted to have a more clear comparison of our results against all of those. Alg-F(A) leads to 0 violations, while our algorithm produces a small number of violations in a few cases, and Alg-F(B) leads to a significant number of violations. In Table 6, we show the cost of the clusterings produced by ALG-F and Algorithm 2, measured in the normalized metric space by taking the average of the maximum radius of any cluster over the 5000 runs. Alg-F(b) leads to the lowest objective, followed relatively closely by our algorithm, and then finally Alg-F(A) has significantly higher objective values.

Acknowledgements

The authors would like to sincerely thank Samir Khuller for useful discussions that led to some of the technical results of this work. In addition, we thank Bill Gasarch for his devotion to building a strong REU program, which facilitated coauthor Chakrabarti's collaboration. Finally, we thank the anonymous referees for multiple useful suggestions.

Brian Brubach was supported in part by NSF award CCF-1749864. John Dickerson was supported in part by NSF CAREER Award IIS-1846237, NSF Award CCF-1852352, NSF D-ISN Award #2039862, NIST MSE Award #20126334, NIH R01 Award NLM-013039-01, DARPA GARD Award #HR00112020007, DoD WHS Award #HQ003420F0035, DARPA Disruptioneering Award (SI3-CMD) #S4761 and Google Faculty Research Award. Aravind Srinivasan was supported in part by NSF awards CCF-1422569, CCF-1749864, and CCF-1918749, as well as research awards from Adobe, Amazon, and Google. Leonidas Tsepenekas was supported in part by NSF awards CCF-1749864 and CCF-1918749, and by research awards from Amazon and Google.

Ethics Statement

Our primary contribution is general and theoretical in nature, so we do not foresee any immediate and direct negative ethical impacts of our work. That said, one use case of our framework—that we highlight prominently both in the general discussion of theoretical results as well as through experimental results performed on standard and commonly-used datasets—is as a tool to operationalize notions of *fairness* in a broad range of clustering settings. Formalization of fairness as a mathematical concept, while often grounded in legal doctrine (see, e.g., Feldman et al. 2015; Barocas, Hardt, and Narayanan 2019), is still a morally-laden and complicated process, and one to which there is no one-size-fits-all "correct" approach.

Our method supports a number of commonly-used fairness definitions; thus, were tools built based on our framework that operationalized those definitions of fairness, then the ethical implications—both positive and negative—of that decision would also be present. Our framework provides strong theoretical guarantees that would allow decision-makers to better understand the performance of systems built based on our approach. Yet, we also note that any such guarantees should, in many domains, be part of a larger conversation with stakeholders—one including understanding the level of comprehension (e.g., Saha et al. 2020; Saxena et al. 2020) and specific wants of stakeholders (e.g., Holstein et al. 2019; Madaio et al. 2020).

References

Ahmadian, S.; Epasto, A.; Kumar, R.; and Mahdian, M. 2019. Clustering without Over-Representation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19.

Ahmadian, S.; Norouzi-Fard, A.; Svensson, O.; and Ward, J. 2017. Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms. In *FOCS 2017*, 61–72.

Anderson, N.; Bera, S. K.; Das, S.; and Liu, Y. 2020. Distributional Individual Fairness in Clustering. *CoRR* abs/2006.12589. URL https://arxiv.org/abs/2006.12589.

Backurs, A.; Indyk, P.; Onak, K.; Schieber, B.; Vakilian, A.; and Wagner, T. 2019. Scalable Fair Clustering. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 405–413.

Barocas, S.; Hardt, M.; and Narayanan, A. 2019. *Fairness and Machine Learning*. fairmlbook.org, august 2020 edition. http://www.fairmlbook.org.

Basu, S.; Davidson, I.; and Wagstaff, K. 2008. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*. Chapman & Hall/CRC, 1 edition. ISBN 1584889969.

Bera, S.; Chakrabarty, D.; Flores, N.; and Negahbani, M. 2019. Fair Algorithms for Clustering. In *Advances in Neural Information Processing Systems 32*, 4954–4965.

Bercea, I. O.; Groß, M.; Khuller, S.; Kumar, A.; Rösner, C.; Schmidt, D. R.; and Schmidt, M. 2019. On the Cost of Essentially Fair Clusterings. In *APPROX/RANDOM 2019*, volume 145, 18:1–18:22.

Bilenko, M.; Basu, S.; and Mooney, R. J. 2004. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the twenty-first international conference on Machine learning*, 11.

Brubach, B.; Chakrabarti, D.; Dickerson, J. P.; Khuller, S.; Srinivasan, A.; and Tsepenekas, L. 2020. A Pairwise Fair and Community-preserving Approach to k-Center Clustering. In *International Conference on Machine Learning (ICML)*.

Byrka, J.; Pensyl, T.; Rybicki, B.; Srinivasan, A.; and Trinh, K. 2017. An Improved Approximation for k-Median and Positive Correlation in Budgeted Optimization. *ACM Trans. Algorithms* .

Chakrabarty, D.; and Negahbani, M. 2019. Generalized Center Problems with Outliers. *ACM Trans. Algorithms* .

Chang, Y.; Chen, J.; Cho, M. H.; Castaldi, P. J.; Silverman, E. K.; and Dy, J. G. 2017. Multiple clustering views from multiple uncertain experts. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 674–683. JMLR. org.

Chierichetti, F.; Kumar, R.; Lattanzi, S.; and Vassilvitskii, S. 2017. Fair Clustering Through Fairlets. In *Advances in Neural Information Processing Systems 30*.

Davidson, I.; Ravi, S. S.; and Shamis, L. 2010. A SAT-based Framework for Efficient Constrained Clustering. In *Proceedings of the 2010 SIAM International Conference on Data Mining*.

Dua, D.; and Graff, C. 2017. UCI Machine Learning Repository. URL http://archive.ics.uci.edu/ml. Last accessed August 2020.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness Through Awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12.

Edgar, R. C. 2013. UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nat Methods* 10(10): 996–998. URL http://www.ncbi.nlm.nih.gov/pubmed/23955772.

Edgar, R. C. 2018. Updating the 97% identity threshold for 16S ribosomal RNA OTUs. *Bioinformatics* 34(14): 2371–2375.

Esmaeili, S. A.; Brubach, B.; Tsepenekas, L.; and Dickerson, J. P. 2020. Probabilistic Fair Clustering. In *Neural Information Processing Systems (NeurIPS)*.

Feldman, M.; Friedler, S. A.; Moeller, J.; Scheidegger, C.; and Venkatasubramanian, S. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 259–268.

Ghodsi, M.; Liu, B.; and Pop, M. 2011. DNACLUST: accurate and efficient clustering of phylogenetic marker genes. *BMC bioinformatics* 12(1): 271.

Gonzalez, T. F. 1985. Clustering to minimize the maximum intercluster distance. *Theoretical computer science* 38: 293–306.

Harris, D. G.; Pensyl, T.; Srinivasan, A.; and Trinh, K. 2017. A Lottery Model for Center-Type Problems with Outliers. In *APPROX/RANDOM 2017*, volume 81, 10:1–10:19.

Hochbaum, D. S.; and Shmoys, D. B. 1986. A Unified Approach to Approximation Algorithms for Bottleneck Problems. *J. ACM* 33(3).

Holstein, K.; Wortman Vaughan, J.; Daumé III, H.; Dudik, M.; and Wallach, H. 2019. Improving fairness in machine learning systems: What do industry practitioners need? In *Conference on Human Factors in Computing Systems (CHI)*, 1–16.

Huang, L.; Jiang, S.; and Vishnoi, N. 2019. Coresets for Clustering with Fairness Constraints. In *Advances in Neural Information Processing Systems 32*, 7589–7600. Curran Associates, Inc.

Kleinberg, J.; and Tardos, E. 2002. Approximation Algorithms for Classification Problems with Pairwise Relationships: metric labeling and markov random fields. *J.ACM* .

Kohavi, R. 1996. Scaling up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, 202–207. AAAI Press.

Luo, Y.; Tian, T.; Shi, J.; Zhu, J.; and Zhang, B. 2018. Semi-crowdsourced clustering with deep generative models. In *Advances in Neural Information Processing Systems*, 3212–3222.

Madaio, M. A.; Stark, L.; Wortman Vaughan, J.; and Wallach, H. 2020. Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI. In *Conference on Human Factors in Computing Systems (CHI)*, 1–14.

Mihelic, J.; and Robic, B. 2005. Solving the k-center Problem Efficiently with a Dominating Set Algorithm. *J. Comput. Inf. Technol.* 13: 225–234.

Moro, S.; Cortez, P.; and Rita, P. 2014. A Data-Driven Approach to Predict the Success of Bank Telemarketing. *Decision Support Systems* 62. doi:10.1016/j.dss.2014.03.001.

Saha, D.; Schumann, C.; McElfresh, D. C.; Dickerson, J. P.; Mazurek, M. L.; and Tschantz, M. C. 2020. Measuring Non-Expert Comprehension of Machine Learning Fairness Metrics. In *International Conference on Machine Learning (ICML)*.

Saxena, N. A.; Huang, K.; DeFilippis, E.; Radanovic, G.; Parkes, D. C.; and Liu, Y. 2020. How do fairness definitions fare? Testing public attitudes towards three algorithmic definitions of fairness in loan allocations. *Artificial Intelligence* 283: 103238.

Stackebrandt, E.; and Goebel, B. M. 1994. Taxonomic Note: A Place for DNA-DNA Reassociation and 16S rRNA Sequence Analysis in the Present Species Definition in Bacteriology. *International Journal of Systematic and Evolutionary Microbiology* 44(4): 846–849.

Van Engelen, J. E.; and Hoos, H. H. 2020. A survey on semi-supervised learning. *Machine Learning* 109(2): 373–440.

Wagstaff, K.; Cardie, C.; Rogers, S.; and Schrödl, S. 2001. Constrained K-means Clustering with Background Knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, 577–584.

Westcott, S.; and Schloss, P. 2017. OptiClust, an improved method for assigning amplicon-based sequence data to operational taxonomic units. In *mSphere*, volume 2.

Yeh, I.; and Lien, C.-H. 2009. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications* 36: 2473–2480. doi:10.1016/j.eswa.2007.12.020.

Zhang, J.; and Yan, R. 2007. On the Value of Pairwise Constraints in Classification and Consistency. In *Proceedings of the 24th International Conference on Machine Learning*.

Zhu, X.; Ghahramani, Z.; and Lafferty, J. 2003. Semi-supervised Learning Using Gaussian Fields and Harmonic Functions. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03.