Carnegie Mellon University
Research Showcase @ CMU

Computer Science Department | School of Computer Science

1-2009

Approximate Clustering without the Approximation

Maria-Florina Balcan
Carnegie Mellon University

Avrim Blum
Carnegie Mellon University, avrim@cs.cmu.edu

Anupam Gupta
Carnegie Mellon University, anupamg@cs.cmu.edu

Follow this and additional works at: http://repository.cmu.edu/compsci

Published In
Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, .

This Conference Proceeding is brought to you for free and open access by the School of Computer Science at Research Showcase @ CMU. It has been accepted for inclusion in Computer Science Department by an authorized administrator of Research Showcase @ CMU. For more information, please contact research-showcase@andrew.cmu.edu.

Approximate Clustering without the Approximation

Maria-Florina Balcan* Avrim Blum† Anupam Gupta‡

Abstract

Approximation algorithms for clustering points in metric spaces is a flourishing area of research, with much research effort spent on getting a better understanding of the approximation guarantees possible for many objective functions such as k-median, k-means, and min-sum clustering.

This quest for better approximation algorithms is further fueled by the implicit hope that these better approximations also give us more accurate clusterings. E.g., for many problems such as clustering proteins by function, or clustering images by subject, there is some unknown "correct" target clustering and the implicit hope is that approximately optimizing these objective functions will in fact produce a clustering that is close (in symmetric difference) to the truth.

In this paper, we show that if we make this implicit assumption explicit—that is, if we assume that any c-approximation to the given clustering objective F is ϵ-close to the target—then we can produce clusterings that are $O(\epsilon)$-close to the target, *even for values c for which obtaining a c-approximation is NP-hard*. In particular, for k-median and k-means objectives, we show that we can achieve this guarantee for any constant $c > 1$, and for min-sum objective we can do this for any constant $c > 2$.

Our results also highlight a somewhat surprising conceptual difference between assuming that the *optimal* solution to, say, the k-median objective is ϵ-close to the target, and assuming that any *approximately optimal* solution is ϵ-close to the target, even for approximation factor say $c = 1.01$. In the former case, the problem of finding a solution that is $O(\epsilon)$-close to the target remains computationally hard, and yet for the latter we have an efficient algorithm.

*School of Computer Science, Carnegie Mellon University. Supported in part by NSF grant CCF-0514922, by an IBM Graduate Fellowship, and by a Google Research Grant. ninamf@cs.cmu.edu

†School of Computer Science, Carnegie Mellon University. Supported in part by NSF grant CCF-0514922, and by a Google Research Grant. avrim@cs.cmu.edu

‡Computer Science Department, Carnegie Mellon University. Supported in part by NSF awards CCF-0448095 and CCF-0729022, and an Alfred P. Sloan Fellowship. anupamg@cs.cmu.edu

1 Introduction

The field of approximation algorithms for clustering points in metric spaces is a very active one, with a large number of algorithms having been developed for clustering objectives like k-median, k-means, and min-sum clustering. The k-median problem has a $3+\epsilon$-approximation [AGK+04], and it is NP-hard to approximate to better than $1+2/e$ [JMS02]. The k-means problem for general metric spaces has a constant-factor approximation, and admits a PTAS in Euclidean spaces for constant number of clusters k [KSS04]. The min-sum clustering problem admits an $O(\log^{1+\delta} n)$-approximation for general metric spaces, and admits a PTAS when k is a constant [dlVKKR03]. For most of these problems, the approximation guarantees do not match the known hardness results, and much effort is spent on obtaining tighter approximation guarantees.

However, this search for better approximation algorithms is motivated not just by the desire to pin down the tractability threshold for these objectives: there is the underlying hope that better approximations will give more meaningful clusterings of the underlying data. Indeed, for many clustering problems, such as clustering proteins by function, or clustering images by subject, the real goal is to classify the points correctly, and these objectives are only a proxy. That is, there is some unknown "correct" target clustering—such as grouping the proteins by their actual functions, or grouping the images by who is actually in them—and the implicit hope is that approximately optimizing these objectives will in fact produce a clustering that is close in symmetric difference to the truth. In other words, implicit in taking the approximation-algorithms approach is the hope that any c-approximation to our given objective will be pointwise close to the true answer, and our motivation for improving a c_2-approximation to a c_1-approximation (for $c_1 < c_2$) is that perhaps this closeness property holds for c_1 but not c_2.

In this paper, we show that if we make this implicit assumption explicit, and assume that any c-approximation to the given clustering objective F is ϵ-close[1] to the target clustering, then we can in fact produce a clustering that is $O(\epsilon)$-close to the target, *even for values c for which obtaining a c-approximation is provably NP-hard*. In particular, for k-median and k-means objectives, we show that we can achieve this guarantee for any constant $c > 1$, and for min-sum objective when the target clusters are "large" we can do this for any constant $c > 2$. Moreover, if clusters are sufficiently large compared to $\frac{\epsilon n}{c-1}$ then for k-median we can actually get ϵ-close to the target (rather than $O(\epsilon)$-close).

Thus, we show that we do not need to find a better approximation algorithm in order to get the properties that such algorithms would imply: we can approximate the target without approximating the objective (up to a constant factor loss in the misclassification rate ϵ in some cases). Moreover, the problem of finding a c-approximation to these objectives even with this assumption is as hard as finding a c-approximation to the objectives without it (see Theorem 17 in Appendix A) so we *must* bypass the objective to do so.

Our results also show that there is a perhaps unexpected conceptual difference between assuming that the *optimal* solution to, say, the k-median objective is ϵ-close to the target, and assuming that any *approximately optimal* solution is ϵ-close to the target, even for approximation factor $c = 1.01$ (say). In the former case, the problem of finding a solution that is $O(\epsilon)$-close to the target remains computationally hard (see Section 2.1 and Appendix A), and yet for the latter case we give efficient algorithms.

1.1 Related Work

Work on approximation algorithms: For *k-median*, $O(1)$-approximations were given by [CGTS99, JV01, CG99] and the best approximation guarantee known is $(3+\epsilon)$ due to [AGK+04]. A reduction from max-k-coverage shows an easy $(1+2/e)$-hardness of approximation [GK99, JMS02]. The k-median problem on constant-dimensional Euclidean spaces admits a PTAS [ARR99].

For *k-means* on general metric spaces, one can derive a constant approximation using ideas from

[1] ϵ-close means only an ϵ fraction of the points are misclassified; see Section 2 for a formal definition.

k-median—the squared distances do not form a metric, but are close enough for the proofs to go through; an approximation-hardness of $1+8/e$ follows from the ideas of [GK99, JMS02]. This problem is very often studied in Euclidean space, where a near-linear time $(1+\epsilon)$-approximation algorithm is known for the case of constant k and ϵ [KSS04]. Lloyd's local search algorithm [Llo82] is often used in practice, despite having poor worst-case performance [AV06]. Ostrovsky et al. [ORSS06] study ways of seeding Lloyd's local search algorithm: they show that on instances satisfying an *ϵ-separation* property, this seeding results in solutions with provable approximation guarantees. We show in Section 2.2 that their assumption can be quite a bit stronger than ours—though, in part, they need a stronger assumption because their goal is different (they want to approximate the objective whereas we want to approximate the target).

Min-sum k-clustering on general metric spaces admits a PTAS for the case of constant k by Fernandez de la Vega et al. [dlVKKR03] (see also [Ind99]). For the case of arbitrary k there is an $O(\delta^{-1}\log^{1+\delta} n)$-approximation algorithm in time $n^{O(1/\delta)}$ due to Bartal et al. [BCR01]. The problem has also been studied in geometric spaces for constant k by Schulman [Sch00] who gave an algorithm for (R^d, ℓ_2^2) that either output a $(1+\epsilon)$-approximation, or a solution that agreed with the *optimum* clustering on $(1-\epsilon)$-fraction of the points (but could have much larger cost than optimum); the runtime is $O(n^{\log\log n})$ in the worst case and linear for sublogarithmic dimension d.

Related work that considers a target clustering: There has been significant work in the machine learning and theoretical computer science communities on clustering or learning with mixture models [AM05, AK05, DHS01, DGL96, KSV05, VW04, Das99]. That work, like ours, has an explicit notion of a correct ground-truth clustering of the data points; however, they all make strong probabilistic assumptions about the data.

In recent work, Balcan et al. [BBV08] investigated the goal of approximating a desired target clustering without probabilistic assumptions. They analyzed what properties of a pairwise similarity function are sufficient to produce a tree such that some unknown pruning is close to the target, or a small list of clusterings such that the target is close to one of them. In relation to implicit assumptions about approximation algorithms, [BBV08] showed that for k-median, the assumption that any 2-approximation is ϵ-close to the target implies that most of the data satisfies a certain separation property, which they then use to construct a hierarchical clustering such that the target clustering is close to some pruning of the hierarchy. In this paper, we initiate a systematic investigation of the interplay between such assumptions and approximation algorithms. Additionally, the goals in this paper are stronger — we want to output a *single* approximately correct clustering (as opposed to a list of clusterings or a hierarchy), and moreover, we want to succeed for any $c > 1$.

The work of Meila [Mei06] is complementary to ours: it shows sufficient conditions under which k-means instances satisfy the property that near-optimal solutions are ϵ-close to the *optimal* k-means solution.

Other work on clustering: There is a large body of other work which does not assume the existence of a target clustering. For example there has been working on defining measures of clusterability of data sets [ABD08] for telling which data sets are more clusterable, on formulating definitions of good clusterings [GKS06], on axiomatizing clustering (in the sense of postulating what natural axioms should a "good clustering algorithm" satisfy), both with possibility and impossibility results [Kle02], on "comparing clusterings" [Mei03, Mei05], or on efficiently testing if a given data set has a clustering satisfying certain properties [ADPR00]. The main difference between this type of work and our work is that we have an explicit notion of a correct ground-truth clustering of the data points, and indeed the results we are trying to prove are quite different. There is also other interesting work addressing *stability* of various clustering algorithms with connections to model selection [BDPS07, BDvLP06, RC06].

2 Definitions and Preliminaries

The clustering problems in this paper fall into the following general framework: we are given a metric space $\mathcal{M} = (X, d)$ with point set X and a distance function $d : \binom{X}{2} \to R_{\geq 0}$ satisfying the triangle inequality—this is the ambient space. We are also given the actual point set $S \subseteq X$ we want to cluster; we use n to denote the cardinality of S. A k-clustering $\mathcal{C}$ is a partition of S into k sets $C_1, C_2, \ldots, C_k$. In this paper, we always assume that there is a *true* or *target* k-clustering $\mathcal{C}_T$ for the point set S.

Commonly used clustering algorithms seek to minimize some objective function or "score"; e.g., the *k-median* clustering objective assigns to each cluster C_i a "median" $c_i \in X$ and seeks to minimize $\Phi_1(\mathcal{C}) = \sum_{i=1}^{k} \sum_{x \in C_i} d(x, c_i)$, *k-means* clustering minimizes $\Phi_2(\mathcal{C}) = \sum_{i=1}^{k} \sum_{x \in C_i} d(x, c_i)^2$, and *min-sum clustering* minimizes $\Phi_\Sigma = \sum_{i=1}^{k} \sum_{x,y \in C_i} d(x, y)$. Given a function Φ and instance $(\mathcal{M}, S)$, let $\mathrm{OPT}_\Phi = \min_{\mathcal{C}} \Phi(\mathcal{C})$, where the minimum is over all k-clusterings of $(\mathcal{M}, S)$.

We define the distance $dist(\mathcal{C}, \mathcal{C}')$ between two k-clusterings $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ and $\mathcal{C}' = \{C'_1, C'_2, \ldots, C'_k\}$ as the fraction of points on which they disagree under the optimal matching of clusters in $\mathcal{C}$ to clusters in $\mathcal{C}'$; i.e., $dist(\mathcal{C}, \mathcal{C}') = \min_{\sigma \in S_k} \frac{1}{n} \sum_{i=1}^{k} |C_i - C'_{\sigma(i)}|$, where S_k is the set of bijections $\sigma : [k] \to [k]$. We say that two clusterings $\mathcal{C}$ and $\mathcal{C}'$ are *ϵ-close* if $dist(\mathcal{C}, \mathcal{C}') \leq \epsilon$ and we say that a clustering has *error* ϵ if it is ϵ-close to the target. Note that if $\mathcal{C}$ and $\mathcal{C}'$ are ϵ-close and all clusters C_i have size greater than $2\epsilon n$, then the bijection σ minimizing $\frac{1}{n} \sum_{i=1}^{k} |C_i - C'_{\sigma(i)}|$ is unique; in this case we call this the *optimal bijection* σ and we say that $\mathcal{C}$ and $\mathcal{C}'$ *agree* on x if $x \in C_i \cap C'_{\sigma(i)}$ for some i.

The following definition is central to our discussion:

Property 1 (The (c, ϵ)-property) *Given an objective function Φ (such as k-median, k-means, or min-sum), we say that instance $(\mathcal{M}, S)$* satisfies the (c, ϵ)-property for Φ *if all near-optimal clusterings $\mathcal{C}$—those with $\Phi(\mathcal{C}) \leq c \cdot \mathrm{OPT}_\Phi$—are ϵ-close to the target clustering $\mathcal{C}_T$ for $(\mathcal{M}, S)$.*

The above assumption is often implicitly made when proposing to use a c-approximation for the objective Φ to solve a clustering problem in which the true goal is to classify data points correctly; similarly, the motivation for improving a c_2 approximation to a $c_1 < c_2$ approximation is that perhaps the data satisfies the (c_1, ϵ) property for Φ but not the (c_2, ϵ) property.

Note that for any $c > 1$, the (c, ϵ)-property does not require that the target clustering $\mathcal{C}_T$ exactly coincide with the optimal clustering $\mathcal{C}^*$ under objective Φ. However, it does imply the following simple facts:

Fact 2 *Given any instance $(\mathcal{M}, S)$ and objective Φ such that $(\mathcal{M}, S)$ satisfies the (c, ϵ)-property for Φ for $c \geq 1$, the following facts hold:*

(a) The target clustering $\mathcal{C}_T$, and the optimal clustering $\mathcal{C}^*$ are ϵ-close.

(b) The distance between k-clusterings is a metric, and hence a (c, ϵ) property with respect to the target clustering $\mathcal{C}_T$ implies a $(c, 2\epsilon)$ property with respect to the optimal clustering $\mathcal{C}^*$.

Proof: Since all c-approximate solutions are ϵ-close to the target, the optimal clustering $\mathcal{C}^*$ (which is 1-approximate) is ϵ-close to $\mathcal{C}_T$. For the second fact, consider any c-approximate clustering $\mathcal{C}$: by the (c, ϵ)-property w.r.t. $\mathcal{C}_T$, $\mathcal{C}$ is ϵ-close to $\mathcal{C}_T$, which in turn is ϵ-close to $\mathcal{C}^*$. And hence every c-approximate solution is 2ϵ-close to $\mathcal{C}^*$, proving the claim. ■

Thus, in our analysis we can act as if the optimal clustering is indeed the target up to a constant factor loss in the error rate.

2.1 Two Strawman Solutions, and Why They Fail

Before proceeding to our results, we first consider two "strawman" approaches to achieving our goals, and indicate why they do not work.

- First, suppose that the (c, ϵ)-property for some objective Φ implied, say, the $(2c, 2\epsilon)$ property. Then it would be sufficient to simply apply an $O(c)$ approximation in order to have error $O(\epsilon)$ with respect to the target. However, we show in Theorem 16 in the Appendix, for any $c_1 < c_2$ and any $\epsilon, \alpha > 0$, for each of the three objectives we consider (k-median, k-means, and min-sum), there exists a family of metric spaces and target clusterings satisfying the (c_1, ϵ) property for that objective, and yet that do not satisfy even the $(c_2, 1/2 - \alpha)$ property. Thus, the result of a direct application of a c_2-approximation in this case is nearly as poor as possible.

- Second, perhaps the (c, ϵ) assumption implies that finding a c-approximation is somehow trivial. However, this is not the case either: for any $c > 1$, the problem of finding a c-approximation to any of the three objectives we consider under the (c, ϵ) assumption is as hard as finding a c-approximation in general (see Appendix, Theorem 17).

It is also interesting to note that results of the form we are aiming for are *not possible given only the* $(1, \epsilon)$ *property*. Indeed, because the standard hardness-of-approximation reduction for k-median produces a metric in which all pairwise distances lie in a bounded range, the reduction also implies that it is NP-hard, given a data set satisfying the $(1, \epsilon)$ property, to find a clustering of error $O(\epsilon)$; see Theorem 18 in the Appendix.

2.2 Relationship to Similar Concepts

Schulman [Sch00] studies the problem of min-sum clustering in ℓ_2^2 and other geometric spaces, for the case of a constant number of clusters k. For any $\delta > 0$, his algorithm either gives a $(1 + \delta)$ approximation to the optimal cost, or else outputs a clustering that is δ-close to the target clustering; of course, the running time of the algorithm depends on δ. Given a $(1 + \alpha, \epsilon)$-assumption for min-sum clustering, his algorithm can be used—as can the PTASs for metric min-sum clustering [Ind99, dlVKKR03]—to output clusterings that are $O(\epsilon)$-close to the target *for the case of constant k and α*. Our algorithms are somewhat simpler, and they work for non-constant k as well.

Ostrovsky et al. [ORSS06] study k-means in Euclidean space; they call a k-means instance ϵ-*separated* if the optimal k-means cost is at most ϵ^2 times the cost of optimally opening $k - 1$ means; under this assumption on the input, they show how to seed Lloyd's method to obtain a $1 + O(\epsilon^2)$ approximation in d-dimensional Euclidean space in time $O(nkd + k^3 d)$, and a $(1+\delta)$-PTAS with run-time $nd2^{k(1+\epsilon^2)/\delta}$. In Theorem 5.1 of their paper, they show that their ϵ-separatedness assumption implies that any near-optimal solution to k-means is $O(\epsilon^2)$-close to the optimal clustering. *However, the converse is not true*, since an instance could satisfy our property without being ϵ-separated: here is an example[2]. Consider the case $k = 2$ where target cluster C_1 has $(1 - \alpha)n$ points and target cluster C_2 has αn points. Any two points inside the same cluster have distance 1 and any two points inside different clusters have distance $1 + 1/\epsilon$. For any $\alpha \in (\epsilon, 1 - \epsilon)$, this satisfies the $(2, \epsilon)$ property for k-median (and the $(2, \epsilon^2)$ property for k-means for any $\alpha \in (\epsilon^2, 1 - \epsilon^2)$). However, it does not necessarily satisfy the ϵ-separation property. E.g., for $\alpha = 2\epsilon$, the optimal 2-median solution has cost $n - 1$, but the optimal 1-median has cost $< 3n$. Likewise for $\alpha = 2\epsilon^2$, the optimal 2-means solution has cost $n - 1$, but the optimal 1-means has cost $< (3 + 4\epsilon)n$. Thus, the ratio of costs for $k = 1$ and $k = 2$ is not so large.

3 The k-Median Problem

We first study k-median clustering under the (c, ϵ)-property. Our main results are that for any constant $c > 1$, (1) if all clusters are "large", then this property allows us to efficiently find a clustering that is ϵ-close to the target clustering, and (2) for any cluster sizes, we can efficiently

[2]The paper [ORSS06] also shows an implication in the other direction (Theorem 5.2 of their paper shows that their separatedness condition implies a certain closeness property); however, the notion of closeness used there is much stronger.

find a clustering that is $O(\epsilon)$-close to the target. Note that we can do this without obtaining a c-approximation algorithm for the k-median problem. (Recall that obtaining a c-approximation for $c < 1 + 2/\mathrm{e}$ is not even possible unless P=NP.) To prove these results, we first investigate the implications of the (c, ϵ)-property in Section 3.1. We then give our algorithm for the case that all clusters are large in Section 3.2, and our algorithm for arbitrary cluster sizes in Section 3.3.

3.1 Implications of the (c, ϵ)-Property

Given an instance of k-median specified by a metric space $\mathcal{M} = (X, d)$ and a set of points $S \subseteq X$, fix an optimal k-median clustering $\mathcal{C}^* = \{C_1^*, \ldots, C_k^*\}$, and let c_i^* be the center point for C_i^*. Let $w(x) = \min_i d(x, c_i^*)$ be the contribution of x to the k-median objective in $\mathcal{C}^*$ (i.e., x's "weight"), and let $w_2(x)$ be x's distance to the second-closest center point among $\{c_1^*, c_2^*, \ldots, c_k^*\}$. Also, let $w = \frac{1}{n}\sum_{i=1}^{n} w(x) = \frac{\mathrm{OPT}}{n}$ be the average weight of the points. Finally, let $\epsilon^* = dist(\mathcal{C}_T, \mathcal{C}^*)$; so, by our assumption we have $\epsilon^* \leq \epsilon$.

Lemma 3 *If the k-median instance $(\mathcal{M}, S)$ satisfies the $(1+\alpha, \epsilon)$-property with respect to $\mathcal{C}_T$, and each cluster in $\mathcal{C}_T$ has size greater than $2\epsilon n$, then*

(a) at most $(\epsilon - \epsilon^*)n$ points $x \in S$ on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree have $w_2(x) - w(x) < \frac{\alpha w}{\epsilon}$, and

(b) at most $5\epsilon n/\alpha$ points $x \in S$ have $w(x) \geq \frac{\alpha w}{5\epsilon}$.

For the case of general cluster sizes in $\mathcal{C}_T$ we replace (a) and (b) with:

(a') at most $6\epsilon n$ points $x \in S$ have $w_2(x) - w(x) < \frac{\alpha w}{2\epsilon}$.

(b') at most $10\epsilon n/\alpha$ points $x \in S$ have $w(x) \geq \frac{\alpha w}{10\epsilon}$.

Proof: To prove Property (a), assume to the contrary. Then one could take $\mathcal{C}^*$ and move more than $(\epsilon - \epsilon^*)n$ points x on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree to their second-closest clusters, increasing the objective by at most $\alpha\mathrm{OPT}$. Moreover, this new clustering $\mathcal{C}' = \{C_1', \ldots, C_k'\}$ has distance greater than ϵ from $\mathcal{C}_T$, because we begin at distance ϵ^* from $\mathcal{C}_T$ and each move increases this distance by $\frac{1}{n}$ (here we use the fact that because each cluster in $\mathcal{C}_T$ has size greater than $2\epsilon n$, the optimal bijection between $\mathcal{C}_T$ and $\mathcal{C}'$ remains the same as the optimal bijection between $\mathcal{C}_T$ and $\mathcal{C}^*$). Hence we have a clustering that is not ϵ-close to $\mathcal{C}_T$ with cost only $(1+\alpha)\mathrm{OPT}$, a contradiction. Property (b) follows from the definition of the average weight w, and Markov's inequality. For Property (a'), we use Lemma 19 in the Appendix which addresses the case of small clusters. Specifically, assuming for contradiction that more than $6\epsilon n$ points satisfy the condition (a'), Lemma 19 states that we can find a subset of more than $2\epsilon n$ of them such that starting from $\mathcal{C}^*$, for each one that we move to its second-closest cluster, the distance from $\mathcal{C}^*$ increases by $\frac{1}{n}$. Therefore, by increasing the objective by at most $\alpha\mathrm{OPT}$ we can create a clustering $\mathcal{C}'$ that is more than distance 2ϵ from $\mathcal{C}^*$, and so is not ϵ-close to $\mathcal{C}_T$. Property (b') again follows from Markov's inequality. ■

For the case that each cluster in $\mathcal{C}_T$ has size greater than $2\epsilon n$, define the *critical distance* $d_{crit} = \frac{\alpha w}{5\epsilon}$, else define $d_{crit} = \frac{\alpha w}{10\epsilon}$; i.e., these are the values in properties (b) and (b') respectively of Lemma 3. We call point x *good* if both $w(x) < d_{crit}$ and $w_2(x) - w(x) \geq 5d_{crit}$, else x is called *bad*; by Lemma 3 and the definition of ϵ^*, if all clusters in the target have size greater than $2\epsilon n$ then at most a $(1 + 5/\alpha)\epsilon$ fraction of points are bad, and in general at most a $(6 + 10/\alpha)\epsilon$ fraction of points are bad. Let X_i be the *good* points in the optimal cluster C_i^*, and let $B = S \setminus \cup X_i$ be the bad points. Let $b = |B|$.

Lemma 4 (Threshold Graph) *Define the τ-threshold graph $G_\tau = (S, E_\tau)$ by connecting all pairs $\{x, y\} \in \binom{S}{2}$ with $d(x, y) \leq \tau$. For an instance satisfying the $(1+\alpha, \epsilon)$-property and $\tau = 2d_{crit}$, the threshold graph G_τ has the following properties:*

(i) For all x, y in the same X_i, the edge $\{x, y\} \in E(G_\tau)$.

(ii) For $x \in X_i$ and $y \in X_{j \neq i}$, $\{x, y\} \notin E(G_\tau)$. Moreover, such points x, y do not share any neighbors in G_τ.

Proof: For part (i), since x, y are both good, they are at distance less than d_{crit} to their cluster center, by Lemma 3 (b or b'). By the triangle inequality, the distance $d(x,y) \leq d(x,c_i^*) + d(c_i^*,y) \leq 2 \times d_{crit} = \tau$. For part (ii), the distance from x to y's cluster center c_j^* is at least $5d_{crit}$, by Lemma 3 (a or a'). Again by the triangle inequality, $d(x,y) \geq d(x,c_j^*) - d(y,c_j^*) > 5d_{crit} - d_{crit} = 2\tau$. Since each edge in G_τ is between points at distance at most τ, the points x, y cannot share any common neighbors. ■

Hence, the graph G_τ for the above value of τ is fairly simple to describe: each X_i forms a clique, and the neighborhood $N_{G_\tau}(X_i)$ of X_i lies entirely in the bad bucket B with no edges going between X_i and $X_{j\neq i}$, or between X_i and $N_{G_\tau}(X_{j\neq i})$. We now show how we can use this to find a clustering of error at most ϵ if the size of each X_i is large (Section 3.2) and how we can get error $O(\epsilon)$ for general cluster sizes (Section 3.3).

3.2 An algorithm for large clusters

We begin with the following lemma. Recall that $b = |B|$.

Lemma 5 *Given a graph $G = (S, E)$ satisfying properties (i), (ii) of Lemma 4 and where each $|X_i| \geq b + 2$, there is an efficient algorithm that outputs a k-clustering with each X_i contained in a distinct cluster.*

Proof: Construct a graph $H = (S, E')$ where we place an edge $\{x, y\} \in E'$ if x and y have at least b common neighbors in G. By property *(i)* each X_i is a clique of size $\geq b+2$ in G, so each pair $x, y \in X_i$ has at least b common neighbors in G and hence $\{x, y\} \in E'$. Now consider $x \in X_i \cup N_G(X_i)$, and $y \notin X_i \cup N_G(X_i)$: we claim $\{x, y\} \notin E'$. Indeed, by property *(ii)*, x and y cannot share neighbors that lie in X_i (since $y \notin X_i \cup N_G(X_i)$), nor in some $X_{j\neq i}$ (since $x \notin X_j \cup N_G(X_j)$). Hence the common neighbors of x, y all lie in B, which has size b. Moreover, at least one of x and y must itself belong to B, else they would have no common neighbors by property *(ii)*; hence, the number of distinct common neighbors is at most $b - 1$, which implies that $\{x, y\} \notin E'$.

Thus each X_i is contained within a distinct component of the graph H; the remaining components of H contain vertices from the "bad bucket" B. Since the X_i's are larger than B, we can obtain the claimed clustering by taking the largest k components in H, adding the vertices of all other smaller components to any of these, and using this as the k-clustering. ■

We now show how we can use Lemma 5 to find a clustering that is ϵ-close to $\mathcal{C}_T$. For simplicity, we begin by assuming that we are given the value of $w = \frac{\text{OPT}}{n}$, and then we show how this assumption can be removed.

Theorem 6 (The "known w" Case) *If the k-median instance satisfies the $(1+\alpha, \epsilon)$-property and each cluster in the target clustering has size at least $(3+10/\alpha)\epsilon n+2$, then given w we can efficiently find a clustering that is ϵ-close to the target.*

Proof: Since each cluster in the target clustering has at least $(3 + 10/\alpha)\epsilon n + 2$ points, and the *optimal k-median clustering* $\mathcal{C}^*$ differs from the target clustering by $\epsilon^* n \leq \epsilon n$ points, each cluster in C^* must have at least $(2 + 10/\alpha)\epsilon n + 2$ points. Moreover, by Lemma 3, the bad points B constitute at most $(1 + 5/\alpha)\epsilon n$ points, and hence each $|X_i| = |C_i^* \setminus B| \geq (1 + 5/\alpha)\epsilon n + 2 = b + 2$.

Now, given w, we can construct the graph G_τ with $\tau = 2d_{crit}$ (which we can compute from the given value of w), and apply Lemma 5 to find a k-clustering $\mathcal{C}'$ where each X_i is contained within a distinct cluster. Note that this clustering $\mathcal{C}'$ differs from the optimal clustering $\mathcal{C}^*$ only in the bad points which constitute an $O(\epsilon/\alpha)$ fraction of the total. Hence, it is at distance $O(\epsilon/\alpha + \epsilon)$ from the target. However, our goal is to get ϵ-close to the target, which we do as follows.

Call a point x "red" if it satisfies condition (a) in Lemma 3 (i.e., $w_2(x) - w(x) < 5d_{crit}$), "yellow" if it is not red but satisfies condition (b) in Lemma 3 (i.e., $w(x) \geq d_{crit}$), and "green" otherwise. So,

the green points are those in the sets X_i, and we have partitioned the bad set B into red points and yellow points. Let $\mathcal{C}' = \{C'_1, \ldots, C'_k\}$ and recall that $\mathcal{C}'$ agrees with $\mathcal{C}^*$ on the green points, so without loss of generality we may assume $X_i \subseteq C'_i$. We now construct a new clustering $\mathcal{C}''$ that agrees with $\mathcal{C}^*$ on both the green and yellow points. Specifically, for each point x and each cluster C'_j, compute the median distance $d_{med}(x, j)$ between x and all points in C'_j; then insert x into the cluster C''_i for $i = \text{argmin}_j d_{med}(x, j)$. Since each non-red point x satisfies $w_2(x) - w(x) \geq 5d_{crit}$, and all green points g satisfy $w(g) < d_{crit}$, this means that any non-red point x must satisfy the following two conditions: (1) for a green point g_1 in the *same* cluster as x in $\mathcal{C}^*$ we have $d(x, g_1) \leq w(x) + d_{crit}$, and (2) for a green point g_2 in a *different* cluster than x in $\mathcal{C}^*$ we have $d(x, g_2) \geq w_2(x) - d_{crit} \geq w(x) + 4d_{crit}$. Therefore, $d(x, g_1) < d(x, g_2)$. Since each cluster in $\mathcal{C}'$ has a strict majority of green points (even with point x removed) all of which are clustered as in $\mathcal{C}^*$, this means that for a non-red point x, the median distance to points in its correct cluster with respect to $\mathcal{C}^*$ is less than the median distance to points in any incorrect cluster. Thus, $\mathcal{C}''$ agrees with $\mathcal{C}^*$ on all non-red points. Finally, since there are at most $(\epsilon - \epsilon^*)n$ red points on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree by Lemma 3, this implies that $dist(\mathcal{C}'', \mathcal{C}_T) \leq (\epsilon - \epsilon^*) + \epsilon^* = \epsilon$ as desired. ■

We now extend the above argument to the case where we are not given the value of w.

Theorem 7 (The "unknown w" Case) *If the k-median instance satisfies the $(1 + \alpha, \epsilon)$-property and each cluster in the target clustering has size at least $(4 + 15/\alpha)\epsilon n + 2$, then we can efficiently find a clustering that is ϵ-close to the target.*

Proof: If we are not given the value w, we instead run the algorithm of Lemma 5 repeatedly for different values of w, starting with $w = 0$ (so the graph G_τ is empty) and at each step increasing w to the next value such that G_τ contains at least one new edge (so we have at most n^2 different guesses to try). If some guess for w causes the k largest components of H to miss more than $b = (2+10/\alpha)\epsilon n$ points, or if any of these components have size $\leq b$, then we reject, and increase w. Otherwise, we define $\mathcal{C}'$ to be the k largest components in H (so up to b points may be unclustered) and continue to the second phase of the algorithm for the known-w case constructing clustering $\mathcal{C}''$.

Note that we still might have too small a guess for w, but this just means that the resulting graphs G_τ and H can only have fewer edges than the corresponding graphs for the correct w. Hence, some of the X_i's might not have fully formed into connected components in H. However, if the k largest components have size greater than b, then we never misclassify the good points lying in these largest components. We might misclassify all the bad points (at most b of these), and might fail to cluster at most b of the points in the actual X_i's (i.e., those not lying in the largest k components), but this nonetheless guarantees that each cluster $\mathcal{C}'_i$ contains at least $|X_i| - b \geq b + 2$ correctly clustered green points (with respect to $\mathcal{C}^*$) and at most b misclassified points. Therefore, as shown in the proof of Theorem 7, the resulting clustering $\mathcal{C}''$ will correctly cluster all non-red points as in $\mathcal{C}^*$ and so is at distance at most $(\epsilon - \epsilon^*) + \epsilon^* = \epsilon$ from $\mathcal{C}_T$. ■

3.3 An Algorithm for the General Case

The algorithm in the previous section required the minimum cluster size in the target to be large (of size $\Omega(\epsilon n)$). In this section, we show how this requirement can be removed using a different algorithm that finds a clustering that is $O(\epsilon/\alpha)$-close to the target; while the algorithm is just as simple, we need to be a bit more careful in the analysis. (Again, we will assume we know $w = \frac{\text{OPT}}{n}$, and discharge this assumption later.)

Theorem 8 (k-Median: General Case) *If the k-median instance satisfies the $(1 + \alpha, \epsilon)$-property and we are given the value of w, the above algorithm produces a clustering which is $O(\epsilon/\alpha)$-close to the target.*

Algorithm 1 k-median Algorithm: General Case

Input: w, $\epsilon \leq 1$, $\alpha > 0$, k.
Step 1: Construct the τ-threshold graph G_τ with $\tau = 2d_{crit} = \frac{1}{5}\frac{\alpha w}{\epsilon}$.
Step 2: For $j = 1$ to k do:
Pick the vertex v_j of highest degree in G_τ.
Remove v_j and its neighborhood from G_τ and call this cluster $C(v_j)$.
Step 3: Output the k clusters $C(v_1), \ldots, C(v_{k-1}), S - \cup_{i=1}^{k-1} C(v_i)$.

Proof: Recall the notation from Section 3.1: the graph G_τ satisfies properties *(i),(ii)* of Lemma 4. We show that the greedy method of Step 2 above correctly captures most of the cliques $X_1, X_2, \ldots, X_k$ in G_τ—in particular, we show there is a bijection $\sigma : [k] \to [k]$ such that $\sum_i |X_{\sigma(i)} \setminus C(v_i)| = O(b)$. Since the b bad points (i.e., those in $B = S \setminus \cup_{i=1}^{k} X_i$) may potentially all be misclassified, this gives an additional error of b.

Let us think of each clique X_i as initially "unmarked", and then "marking" it the first time we choose a cluster $C(v_j)$ that intersects it. We now consider two cases. If the jth cluster $C(v_j)$ intersects some *unmarked* clique X_i, we will assign $\sigma(j) = i$. (Note that it is not possible for $C(v_j)$ to intersect two cliques X_i and $X_{j \neq i}$, since by Lemma 4*(ii)* these cliques have no common neighbors.) If $C(v_j)$ misses r_i points from X_i, then since the vertex v_j defining this cluster had maximum degree and X_i is a clique, we must have picked at least r_i elements from B in $C(v_j)$. Therefore the total sum of these r_i can be at most $b = |B|$, and hence $\sum_j |X_{\sigma(j)} \setminus C(v_j)| \leq b$, where the sum is over j's that correspond to the first case.

The other case is if $C(j)$ intersects a previously marked clique X_i. In this case we assign $\sigma(j)$ to any arbitrary clique $X_{i'}$ that is not marked by the end of the process. Note that the total number of points in such $C(j)$'s must be at most the number of points remaining in the marked cliques (i.e., $\sum_j r_j$), and possibly the bad points (at most b of them). Since the cliques $X_{i'}$ were unmarked at the end, their sizes must be bounded by the size of the $C(j)$'s, and hence by $|B| + \sum_i r_i \leq 2b$. This shows that the sum over such j's, $\sum_j |X_{\sigma(j)} \setminus C(v_j)| \leq 2b$. Therefore, overall, the total error over all $C(v_j)$ with respect to the k-median optimal is the two sums above, plus potentially the bad points, which gives us at most $4b$ points. Adding in the extra ϵn to account for the distance between the k-median optimum and the target clustering yields the claimed $4b + \epsilon n = O(\epsilon/\alpha)n$ result for the case that we are given the value of w. ■

Not Knowing the Value of w. If we do not know the value of w (and hence of τ), unfortunately the method used in the proof of Theorem 7 may not work, because we might split some large cluster causing substantial error, and not be able to recognize our mistake (because we only miss small clusters which do not result in very many points being left over). However, we can instead run an off-the-shelf k-median approximation algorithm to produce an estimate for w that is off by only a constant factor, and use this estimate instead. In particular, if we have a β-approximation $\tilde{w}$ (i.e., say $w \leq \tilde{w} \leq \beta\tilde{w}$, an analog of Lemma 4 holds for the threshold graph $G_{\tau'}$ with the altered threshold $\tau' = \frac{1}{5}\frac{\alpha\tilde{w}}{\epsilon\beta}$, with the number of bad points now bounded by $b' = (6 + 10\beta/\alpha)\epsilon$. The rest of the proof follows unchanged with all bs replaced by b's, to give us a final bound of $O(\beta\epsilon/\alpha)$ on the number of misclassified points.

4 The k-Means Problem

The algorithm in Section 3.3 for the k-median problem can be easily altered to work for the k-means problem as well. Indeed, if we can prove the existence of a structure like that promised by Lemma 3 and Lemma 4 (albeit with different parameters), the same algorithm and proof would give a good clustering for any objective function. Given some optimal solution for k-means define $w(x) = \min_i d(x, c_i)$ to be the distance of x to its center, which is *the square root of* x's contribution

to the k-means objective function; hence $\mathrm{OPT} = \sum_x w(x)^2$. Again, let $w_2(x) = \min_{j \neq i} d(x, c_j)$ be the distance to the second-closest center, and let $\epsilon^* = dist(\mathcal{C}_T, \mathcal{C}^*)$.

Lemma 9 *If the k-means instance $(\mathcal{M}, S)$ satisfies the $(1+\alpha, \epsilon)$-property and each cluster in the target has size greater than $2\epsilon n$, then*

(a) at most $(\epsilon - \epsilon^*)n$ points $x \in S$ on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree have $w_2(x) < (\frac{\alpha \mathrm{OPT}}{\epsilon n})^{1/2}$, and

(b) at most $25\epsilon n/\alpha$ points $x \in S$ have $w(x) > \frac{1}{5}(\frac{\alpha \mathrm{OPT}}{\epsilon n})^{1/2}$.

For the case of general cluster sizes we replace (a) and (b) with:

(a') at most $6\epsilon n$ points $x \in S$ have $w_2(x) < (\frac{\alpha \mathrm{OPT}}{2\epsilon n})^{1/2}$.

(b') at most $50\epsilon n/\alpha$ points $x \in S$ have $w(x) > \frac{1}{5}(\frac{\alpha \mathrm{OPT}}{2\epsilon n})^{1/2}$.

Proof: The proof is the same as for Lemma 3. For part (a), beginning with $\mathcal{C}^*$ and moving more than $(\epsilon - \epsilon^*)n$ points x on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree would create a clustering $\mathcal{C}'$ at distance more than ϵ from the target (because we begin at distance ϵ^* and the optimal bijection doesn't change, since each cluster in $\mathcal{C}_T$ has size greater than $2\epsilon n$). Hence, at most $(\epsilon - \epsilon^*)n$ points on which $\mathcal{C}_T$ and $\mathcal{C}^*$ agree can have $w_2(x)^2 - w(x)^2 < \frac{\alpha \mathrm{OPT}}{\epsilon n}$, and even fewer points can have $w_2(x)^2 < \frac{\alpha \mathrm{OPT}}{\epsilon n}$ or $w_2(x) < (\frac{\alpha \mathrm{OPT}}{\epsilon n})^{1/2}$. To prove part (b), recall that $\sum_x w(x)^2 = \mathrm{OPT}$, and hence Markov's inequality implies that the number of x with $w(x)^2 > \frac{\alpha \mathrm{OPT}}{25\epsilon n}$ is at most $25\epsilon n/\alpha$. Part (b') follows similarity. Finally, for part (a') we again use Lemma 19 as in the proof of part (a') of Lemma 3 to show that if the condition is violated we can find a subset of more than $2\epsilon n$ points we can move from $\mathcal{C}^*$ while maintaining cost at most $(1+\alpha)\mathrm{OPT}$. ■

Note that the threshold for $w_2(x)$ in part (a) above is again 5 times the threshold for $w(x)$ in part (b), and similarly for (a') and (b'). We can thus define the critical distance d_{crit} as the value in (b) or (b') respectively, and define the $b = (1 + 25/\alpha)\epsilon n$ points that satisfy either (a) or (b) above (in the large-cluster case) or the $b = (6 + 50/\alpha)\epsilon n$ points satisfying (a') or (b') (in the general case) as *bad*. The rest of the proof for achieving an $O(\epsilon/\alpha)$-close clustering for k-medians now goes through unchanged in the k-means case as well. Note that k-means also has a constant-factor approximation, so the results for the case of unknown w go through similarly, with different constants. Unfortunately, the argument for exact ϵ-closeness breaks down because property (a) in Lemma 9 is weaker than property (a) in Lemma 3. We therefore have the following theorem.

Theorem 10 *If the instance satisfies the $(1+\alpha, \epsilon)$-property for the k-means objective, we can efficiently produce a clustering which is $O(\epsilon/\alpha)$-close to the target.*

5 The Min-sum Clustering Problem

Recall that the min-sum k-clustering problem asks to find a k-clustering $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ to minimize the objective function

$$\Phi(\mathcal{C}) = \sum_{i=1}^{j} \sum_{x,y \in C_i} d(x, y).$$

In this section, we show how assuming a $(2+\alpha, \epsilon)$-property for the min-sum clustering problem, and assuming that all the clusters in the target are "large", allows us to find a clustering that is $O(\epsilon)$-close to the target clustering. The general idea is reduce to a problem known as "balanced k-median" (which is related to min-sum clustering to within a multiplicative factor of 2); and extend the techniques from the previous sections to handle the balanced-k-median problem.

5.1 Properties of Min-Sum Clustering

The *balanced k-median* clustering objective assigns to each cluster C_i a "median" $c_i \in X$ and seeks to minimize $\Psi(\mathcal{C}) = \sum_{i=1}^{k} |C_i| \sum_{x \in C_i} d(x, c_i)$. We begin with a useful lemma, which shows that the two objective functions Φ (for min-sum clustering) and Ψ (for balanced-k-median) are related to within a factor of 2.

Lemma 11 ([BCR01]) *Let Ψ be the balanced k-median objective and let Φ be the min-sum objective. For any k-clustering $\mathcal{C}$ of S we have:*

$$\Psi(\mathcal{C})/2 \leq \Phi(\mathcal{C}) \leq \Psi(\mathcal{C}).$$

Lemma 12 *If the instance $(\mathcal{M}, S)$ satisfies the $(2(1+\alpha), \epsilon)$-property for the min-sum objective with respect to the target clustering, then $(\mathcal{M}, S)$ satisfies the $(1+\alpha, \epsilon)$-property for the balanced k-median objective with respect to the target clustering.*

Proof: Let OPT_Φ be an optimal clustering for the min-sum objective Φ, and let OPT_Ψ be an optimal clustering for the balanced-k-median objective Ψ. Lemma 11 implies that $\Psi(\mathrm{OPT}_\Phi) \leq 2\Phi(\mathrm{OPT}_\Phi)$, which further implies that $\Psi(\mathrm{OPT}_\Psi) \leq 2\Phi(\mathrm{OPT}_\Phi)$. So, any clustering $\mathcal{C}$ such that $\Psi(\mathcal{C}) \leq (1+\alpha)\Psi(\mathrm{OPT}_\Psi)$ satisfies $\Psi(\mathcal{C}) \leq 2(1+\alpha)\Phi(\mathrm{OPT}_\Phi)$ which further implies that $\Phi(\mathcal{C}) \leq 2(1+\alpha)\Phi(\mathrm{OPT}_\Phi)$. Since $(\mathcal{M}, S)$ satisfies the $(2(1+\alpha), \epsilon)$-property for the Φ objective with respect to the target clustering, we get that $\mathcal{C}$ is ϵ-close to the target, as desired. ■

Henceforth, we will work with the balanced k-median objective function. Let the balanced k-median optimal clustering be $\mathcal{C}^* = \{C_1^*, \ldots, C_k^*\}$ with objective function value $\mathrm{OPT} = \Psi(\mathcal{C}^*)$. For each cluster C_i^*, let c_i^* be the median point in the cluster. For $x \in C_i^*$, define $w(x) = |C_i^*| d(x, c_i^*)$ and let $w = \mathrm{avg}_x w(x) = \frac{\mathrm{OPT}}{n}$. Define $w_2(x) = \min_{j \neq i} d(x, c_j^*)|C_j^*|$. Let $d_{C_i^*} = \sum_{x \in C_i} d(x, c_i^*)$, and hence $\mathrm{OPT} = \sum_i |C_i^*| d_{C_i^*}$.

Lemma 13 *If the balanced k-median instance $(\mathcal{M}, S)$ satisfies the $(1+\alpha, \epsilon)$-property with respect to the target clustering, then as long as the minimum cluster size is at least $\max(6, 6/\alpha) \cdot \epsilon n$ we have:*
(a) at most 2ϵ-fraction of points $x \in S$ have $w_2(x) < \frac{\alpha w}{4\epsilon}$, and
(b) at most $60\epsilon/\alpha$-fraction of $x \in S$ have $w(x) > \frac{\alpha w}{60\epsilon}$.

Proof: To prove Property (a), assume to the contrary. Then one could move more than a 2ϵ fraction of points from their clusters in the optimal clustering $\mathcal{C}^*$ to the clusters that define their w_2 value. This may increase the sizes of the clusters; let the new clustering be $\mathcal{C}' = (C_1', \ldots, C_k')$, where $|C_i' \setminus C_i^*| = \delta_i n$, so that $\sum_i \delta_i = 2\epsilon$. If a point x moves to cluster C_i' from some other cluster, then it now contributes $w_2(x) \cdot \frac{|C_i'|}{|C_i|}$. Summing over all the points, we get that the cost $\Psi(\mathcal{C}')$ is at most

$$\Psi(\mathcal{C}') \leq \sum_{i=1}^{k} \left((|C_i^*| + \delta_i n) d_{C_i} + \delta_i n \cdot \frac{\alpha w}{4\epsilon} \cdot \frac{|C_i^*| + \delta_i n}{|C_i^*|} \right) \tag{5.1}$$

However, $\delta_i n \leq \sum_i \delta_i n \leq 2\epsilon n \leq \frac{\min(1,\alpha)}{3} |C_i^*|$ (since each cluster size is at least $\max(6, 6/\alpha) \cdot \epsilon n$). Hence, we have

$$\Psi(\mathcal{C}') \leq \sum_{i=1}^{k} \left(1 + \frac{\alpha}{3}\right) |C_i^*| d_{C_i^*} + \frac{4}{3} \sum_{i=1}^{k} \frac{\delta_i \, \alpha \, \mathrm{OPT}}{4\epsilon} \leq (1+\alpha)\mathrm{OPT}. \tag{5.2}$$

This would give a clustering with cost at most $(1+\alpha)\mathrm{OPT}$ that is not 2ϵ-close to the optimal clustering $\mathcal{C}^*$, which is impossible by Fact 2(b). Property (b) above follows from the definition of the average weight w, and Markov's inequality. ■

We call point x *good* if it both $w(x) \leq \frac{\alpha w}{60\epsilon}$ and $w_2(x) \geq \frac{\alpha w}{4\epsilon}$, else x is called *bad*; let X_i be the *good* points in the optimal cluster C_i, and let $B = S \setminus \cup X_i$ be the bad points.

Lemma 14 (Structure of the Balanced-k-median optimum) *If the balanced-k-median instance $(\mathcal{M}, S)$ satisfies the $(1+\alpha, \epsilon)$-property with respect to the target clustering, then as long as the minimum cluster size is at least $\max(6, 6/\alpha) \cdot \epsilon n$ we have:*

(i) For all x, y in the same X_i, we have $d(x,y) < \frac{\alpha}{30}\frac{1}{\epsilon}\frac{w}{|C_i^*|}$

(ii) For $x \in X_i$ and $y \in X_{j \neq i}$, $d(x,y) > \frac{\alpha}{5}\frac{1}{\epsilon}\frac{w}{\min(|C_i^*|,|C_j^*|)}$

(iii) The number of bad points $|B| = |S \setminus \cup X_i|$ is at most $b := (2 + 60/\alpha)\epsilon n$.

Proof: For part (i), since $x, y \in X_i \subseteq C_i^*$ are both good, they are at distance less than $\frac{\alpha}{60}\frac{1}{\epsilon}\frac{w}{|C_i^*|}$ to their cluster center (Lemma 13(a)), and hence at distance at most $\frac{\alpha}{30}\frac{1}{\epsilon}\frac{w}{|C_i^*|}$ to each other. For part (ii) assume without loss of generality that $|C_i^*| \geq |C_j^*|$; using both parts of Lemma 13 and the fact that both $x \in C_i^*, y \in C_j^*$ are good, we have $d(y, c_j) \leq \frac{\alpha}{60}\frac{1}{\epsilon}\frac{w}{|C_j^*|}$, and $d(x, c_j) \geq w_2(x) > \frac{\alpha}{4}\frac{1}{\epsilon}\frac{w}{|C_j^*|}$, so

$$d(x,y) \geq \alpha w \cdot \left(\frac{1}{4} - \frac{1}{60}\right)\frac{1}{\epsilon}\frac{w}{|C_j^*|} > \frac{\alpha}{5}\frac{1}{\epsilon}\frac{w}{\min(|C_i^*|,|C_j^*|)} \tag{5.3}$$

where we use that $|C_j^*| = \min(|C_i^*|, |C_j^*|)$. Part (iii) follows from Lemma 13 and the trivial union bound. ■

While Lemma 14 is similar in spirit to Lemma 4, there is a crucial difference: the distance between the good points in X_i and X_j is no longer lower bounded by some absolute value τ, but rather the bound depends on the sizes of X_i and X_j. However, a redeeming feature is that the separation between them is large compared to the diameters of both X_i and X_j; we will use this feature crucially in our algorithm.

5.2 The Algorithm for Min-Sum/Balanced-k-Median Clustering

For the algorithm below, define *critical thresholds* $\tau_0, \tau_1, \tau_2, \ldots$ as: $\tau_0 = 0$ and τ_i is the ith smallest distinct distance $d(x,y)$ for $x, y \in S$. Thus, $G_{\tau_0}, G_{\tau_1}, \ldots$ are the only distinct threshold graphs possible.

Theorem 15 *If the balanced k-median instance satisfies the $(1+\alpha, \epsilon)$-property and we are given the value of w, then so long as the smallest correct cluster has size greater than $(6 + 120/\alpha)\epsilon n$, Algorithm 2 produces a clustering that is $O(\epsilon/\alpha)$-close to the target. If we are not given w, then we can use Algorithm 2 as a subroutine to produce a clustering that is $O(\epsilon/\alpha)$-close to the target.*

Algorithm 2 Balanced k-median Algorithm

Input: $(\mathcal{M}, S)$, w, $\epsilon \leq 1$, $\alpha > 0$, k, $b := (2 + 60/\alpha)\epsilon n$.
Let the initial threshold $\tau = \tau_0$.

Step 1: If $k = 0$ or $S = \emptyset$, stop.
Step 2: Construct the τ-threshold graph G_τ on the current set S of points.
Step 3: Create a new graph H by connecting two points by in S an edge if they share at least b neighbors in common in G_τ.
Step 4: Let C be largest connected component in H. If $|C| \geq \frac{1}{20}\frac{\alpha w}{\epsilon \tau}$,
then output C as a cluster, set $k \leftarrow k-1$, $S \leftarrow S \setminus C$, and go to Step 1,
else increase τ to the next critical threshold and go to Step 1.

Proof: Since each cluster in the target clustering has more than a $(6 + 120/\alpha)\epsilon$ fraction of the points by the assumption, the *optimal balanced-k-median clustering* $\mathcal{C}^*$ must differ from the target clustering by fewer than ϵn points, and hence each cluster in $\mathcal{C}^*$ must have at least $(5 + 120/\alpha)\epsilon n$

points. Moreover, by Lemma 13, the bad points B constitute at most $(2+60/\alpha)\epsilon$ fraction of points, and hence each $|X_i| = |C_i^* \setminus B| > (3+60/\alpha)\epsilon n \geq (2+60/\alpha)\epsilon n + 2 = b+2$.

Assume we know w. Consider what happens in the execution of the algorithm: as we increase τ, the sizes of the H-components increase (since we are adding more edges in G_τ). This happens until the largest H-component is "large enough" (i.e., the condition in Step 4 gets satisfied) and we output a component whose size is large enough; and then we go back to raising τ.

We claim that every time we output a cluster in Step 4, this cluster completely contains some X_i and includes no points in any $X_{j\neq i}$. More specifically, we show that as we increase τ, the condition in Step 4 will be satisfied *after* all the good points in the some cluster have been fully connected, but *before* any edges appear between good points in different clusters. It suffices to show that the first cluster output by the algorithm contains some X_i entirely; the claim for the subsequent output clusters is the same. Assume that $|C_1^*| \geq |C_2^*| \geq \ldots \geq |C_k^*|$, and let $n_i = |C_i^*|$. Define $d_i = \frac{\alpha}{30}\frac{w}{\epsilon}\frac{1}{|C_i^*|}$ and recall that $\min_{x,y\in X_i} d(x,y) \leq d_i$.

We first claim that as long as $\tau \leq 3\,d_1$, no two points belonging to different X_i's can lie in the same H-component. Since the distance between points in any X_i and $X_{j\neq i}$ is strictly greater than $\frac{\alpha}{5}\frac{1}{\epsilon}\frac{w}{\min(|C_i^*|,|C_j^*|)} \geq 2\tau$ for any $\tau \leq 3\,d_1$, every $x \in X_i$ and $y \in X_j$ share no common neighbors, and hence, by an argument identical to that in Lemma 5, the nodes x, y belong to different components of H.

Next, we claim that for values of $\tau < \min\{d_i, 3d_1\}$, the H-component containing X_i cannot be output by Step 4. Indeed, since $\tau < 3d_1$, no X_i and X_j belong to the same H-component by the above claim, and hence any H-component containing points from X_i has size at most $|C_i^*|+|B| < \frac{3n_i}{2}$; however, the minimum size bound $\frac{1}{20}\frac{\alpha w}{\epsilon\tau} > \frac{3n_i}{2}$ for values of $\tau < d_i$, and hence the condition of Step 4 is not satisfied. Note that when $\tau \geq d_i$, all the points of X_i lie in the same H-component.

Finally, we show that the condition in Step 4 becomes true for some H-component fully containing some X_i for some value $\tau = [d_1, 3d_1]$. (By the argument in the previous paragraph, $\tau \geq d_i$, and hence the output component will fully contain X_i.) For the sake of contradiction, suppose not. But note at time $\tau = 3d_1$, at least the H-component containing X_1 has size at least $|C_1^*| - |B| > n_1/2$ and will satisfy the condition (which at time $\tau = 3d_1$ requires a cluster of size $\frac{1}{20}\frac{\alpha w}{\epsilon}\frac{30\epsilon n_i}{3\alpha w} = n_1/2$), giving the contradiction.

To recap, we showed that by time $3d_1$ none of the clusters have merged together, and the Step 4 condition was satisfied for at least the component containing X_1 (and hence for the largest component) at some time prior to that. Moreover, this largest component must fully contain some set X_i and no points in $X_{j\neq i}$. Finally, we can iterate the same argument on the set $S \setminus X_i$ to complete the proof for the case when we know w.

The case when we do not know w. In this case, we do not want to use a β-approximation algorithm for balanced k-median to obtain a clustering that is $O(\beta\epsilon/\alpha)$-close to the target, because the balanced-k-median (and minsum clustering) problems only have a logarithmic approximation for arbitrary k, and hence our error would blow up by a logarithmic factor. Instead, we use the idea of trying increasing values of w: we then stop the first time when we output k clusters that cover at least $n-b = (1-O(\epsilon/\alpha))n$ of the points in S. Clearly, if we reached the correct value of w we would succeed in covering all the good $n-b$ points using our k clusters; we now argue that we will never mistakenly output a high-error clustering.

The argument is as follows. Let us say we *mark* X_i the first time we output a cluster containing at least one point from it. There are three possible sources of mistakes: (a) we may output a cluster prematurely, it may contain some but not all points from X_i, (b) we may output a cluster which contains points from one or more previously marked sets X_j (but no unmarked X_i), or (c) we may output a cluster with points from an unmarked X_i and one or more previously marked X_j. In case (a), if we end up with all but $O(\epsilon/\alpha)$-fraction of the points, we did not miss too many points from the X_i's, so our error is $O(\epsilon/\alpha)$. In case (b), we use up too many clusters and would end

with missing some X_i completely, which would result in more than b unclustered points, and we would try a larger guess for w. The dangerous case is case (c), but we claim case (c) in fact cannot happen. Indeed, the value of τ at which we would form connected components containing points from both X_i and X_j is a constant times larger than the value $\tau_<$ at which all of X_i would be in a single H-component. Moreover, since our guess for w is too small, this H-component would certainly satisfy the condition of Step 4 and be output as a cluster instead. ■

6 Conclusions and Open Questions

A concrete open question is designing an efficient algorithm for the min-sum property which works in the presence of small target clusters. Another natural direction for investigation is designing faster algorithms for all the properties analyzed in this paper. The case of large clusters can be handled by using standard sampling ideas [MOP01, CS04, BD07], however these techniques do not seem to immediately apply in the case where the target clusters are small.

More broadly, it would be interesting to further explore and analyze in our framework other natural classes of commonly used clustering objective functions.

References

[ABD08] M. Ackerman and S. Ben-David. Which data sets are clusterable? - a theoretical study of clusterability. Manuscript, 2008.

[ADPR00] N. Alon, S. Dar, M. Parnas, and D. Ron. Testing of clustering. In *Proc. 41st Annual Symposium on Foundations of Computer Science*, 2000.

[AGK+04] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local search heuristics for k-median and facility location problems. *SIAM J. Comput.*, 33(3):544–562, 2004.

[AK05] S. Arora and R. Kannan. Learning mixtures of arbitrary gaussians. In *ACM Symposium on Theory of Computing*, 2005.

[AM05] D. Achlioptas and F. McSherry. On spectral learning of mixtures of distributions. In *COLT*, 2005.

[ARR99] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for Euclidean k-medians and related problems. In *STOC '98 (Dallas, TX)*, pages 106–113. ACM, New York, 1999.

[AV06] D. Arthur and S. Vassilvitskii. Worst-case and smoothed analyses of the icp algorithm, with an application to the k-means method. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, 2006.

[BBV08] M.-F. Balcan, A. Blum, and S. Vempala. A discrimantive framework for clustering via similarity functions. In *Proceedings of the 40th annual ACM symposium on Theory of Computing*, 2008.

[BCR01] Y. Bartal, M. Charikar, and D. Raz. Approximating min-sum k-clustering in metric spaces. In *Proceedings of the 33rd Symposium on Theory of Computation*, 2001.

[BD07] S. Ben-David. A framework for statistical clustering with constant time approximation for k-median and k-means clustering. *Machine Learning Journal*, 66(2-3):243 – 257, 2007.

[BDPS07] S. Ben-David, D. Pal, and H.U. Simon. Stability of k-means clustering. In *Proceedings of the Twentieth Annual Conference on Learning Theory*, 2007.

[BDvLP06] S. Ben-David, U. von Luxburg, and D. Pal. A sober look at clustering stability. In *Proceedings of the 19th Annual Conference on Learning Theory*, 2006.

[CG99] M. Charikar and S. Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, 1999.

[CGTS99] M. Charikar, S. Guha, E. Tardos, and D. B. Shmoy. A constant-factor approximation algorithm for the k-median problem. In *ACM Symposium on Theory of Computing*, 1999.

[CS04] A. Czumaj and C. Sohler. Sublinear-time approximation for clustering via random samples. In *Proceedings of the 31st International Colloquium on Automata, Language and Programming*, pages 396–407, 2004.

[Das99] S. Dasgupta. Learning mixtures of gaussians. In *Fortieth Annual IEEE Symposium on Foundations of Computer Science*, 1999.

[DGL96] L. Devroye, L. Gyorfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, 1996.

[DHS01] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, 2001.

[dlVKKR03] W. Fernandez de la Vega, Marek Karpinski, Claire Kenyon, and Yuval Rabani. Approximation schemes for clustering problems. In *STOC*, 2003.

[GK99] S. Guha and S. Khuller. Greedy strikes back: Improved algorithms for facility location. *Journal of Algorithms*, 31(1):228–248, 1999.

[GKS06] S. Gollapudi, R. Kumar, and D. Sivakumar. Programmable clustering. In *Symposium on Principles of Database Systems*, pages 348 – 354, 2006.

[Ind99] P. Indyk. Sublinear time algorithms for metric space problems. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, 1999.

[JMS02] K. Jain, M. Mahdian, and A. Saberi. A new greedy approach for facility location problems. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of Computing*, 2002.

[JV01] K. Jain and V. V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *JACM*, 48(2):274 – 296, 2001.

[Kle02] J. Kleinberg. An impossibility theorem for clustering. In *NIPS*, 2002.

[KSS04] A. Kumar, Y. Sabharwal, and S. Sen. A simple linear time $(1+\epsilon)$-approximation algorithm for k-means clustering in any dimensions. In *Proc. 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 454–462, Washington, DC, USA, 2004.

[KSV05] R. Kannan, H. Salmasian, and S. Vempala. The spectral method for general mixture models. In *Proc. 18th Annual Conference on Learning Theory*, 2005.

[Llo82] S.P. Lloyd. Least squares quantization in PCM. *IEEE Trans. Inform. Theory*, 28(2):129–137, 1982.

[Mei03] M. Meila. Comparing clusterings by the variation of information. In *Proceedings of the Annual Conference on Computational Learning Theory*, 2003.

[Mei05] M. Meila. Comparing clusterings – an axiomatic view. In *International Conference on Machine Learning*, 2005.

[Mei06] M. Meila. The uniqueness of a good clustering for k-means. In *Proceedings of the* 23*rd International Conference on Machine Learning*, 2006.

[MOP01] N. Mishra, D. Oblinger, and L. Pitt. Sublinear time approximate clustering. In *Proceedings of Syposium on Discrete Algorithms*, pages 439–447, 2001.

[ORSS06] R. Ostrovsky, Y. Rabani, L. Schulman, and C. Swamy. The effectiveness of lloyd-type methods for the k-means problem. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, 2006.

[RC06] A. Rakhlin and A. Caponnetto. Stability of k-means clustering. In *Neural Information Processing Systems Conference*, 2006.

[Sch00] L.J. Schulman. Clustering for edge-cost minimization (extended abstract). In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 547–555 (electronic), New York, 2000. ACM.

[VW04] S. Vempala and G. Wang. A spectral algorithm for learning mixture models. *J. Comp. Sys. Sci.*, 68(2):841–860, 2004.

A Appendix

Theorem 16 *For any $1 \leq c_1 < c_2$, any $\epsilon, \alpha > 0$, there exists a family of metric spaces G and target clusterings that satisfy the (c_1, ϵ) property for the k-median objective (likewise, k-means and min-sum) and yet do not satisfy even the $(c_2, 1/2 - \alpha)$ property for that objective.*

Proof: We focus first on the k-median objective. Consider a set of n points such that the target clustering consists of one cluster C_1 with $n(1-2\alpha)$ points all at the same location ($d(u,v) = 0$ for all $u, v, \in C_1$) and $k-1$ clusters $C_2, \ldots, C_k$ each consisting of $\frac{2\alpha n}{k-1}$ points, all at distance 1. The distance between points in any two distinct clusters C_i, C_j for $i, j \geq 2$ is D, where $D > 1$ will be defined below. Points in C_1 are at distance greater than $c_2 n$ from any of the other clusters.

In this construction, the target clustering *is* the optimal k-median solution, and has a total k-median cost of $2\alpha n - (k-1)$. We now define D so that there (just barely) exists a c_2 approximation that has error $1/2 - \alpha$ with respect to the target clustering. In particular, consider the solution that merges C_2 and C_3 into a single cluster ($C_4, \ldots, C_k$ will each be their own cluster) and uses 2 clusters to evenly split C_1. This clearly has error at least $1/2 - \alpha$, and furthermore this solution has a cost of $\frac{2\alpha n}{k-1} D + (2\alpha n - \frac{2\alpha n}{k-1} - (k-2))$, and we define D to set this equal to $c_2(2\alpha n - (k-1)) = c_2 \mathrm{OPT}$.

We now claim that any c_1 approximation, however, must be ϵ-close to the target for $k > 1 + \alpha/\epsilon$. In particular, say that a point in any given solution is a "high-cost" point if its contribution to the k-median objective is D, and is a "low-cost" point if its contribution is 0 or 1. (There cannot be any points of cost $> D$ in a c_1-approximation in this construction.) By definition of D, since $c_1 < c_2$, any c_1-approximation can have at most $\frac{\alpha n}{k-1} - 1$ high-cost points. This means that in any c_1-approximation, each of the clusters $C_2, \ldots, C_k$ must have at least one low-cost point and therefore must have at least one of the medians. Since we must have at least one median in cluster C_1, this implies that each cluster C_i has exactly one median, and so the overall error of the clustering is at most the fraction of high-cost points, which is at most ϵ by definition of k.

The same construction, with D defined appropriately, applies to k-means and min-sum objectives as well. ■

Theorem 17 *For k-median, k-means, and min-sum objectives, the problem of finding a c-approximation can be reduced to the problem of finding a c-approximation under the (c, ϵ) assumption. Therefore, the problem of finding a c-approximation under the (c, ϵ) assumption is as hard as the problem of finding a c-approximation in general.*

Proof: Given a metric G with n nodes and a value k (a generic instance of the clustering problem) we construct a new instance satisfying the (c, ϵ) assumption. In particular we create a new graph G' by adding an extra n/ϵ nodes that are all very far away from each other and from the nodes in G (call this distance D). We now let $k' = k + n/\epsilon$ and define the target clustering to be the optimal (k-median, k-means, or min-sum) solution on G, together with each of the points in $G' \setminus G$ in its own singleton cluster.

We first claim that G' satisfies the (c, ϵ) property. This is because any solution that does not put each of the new nodes into its own singleton cluster will incur a high cost. So a c-approx can only differ from the target on G (which has only an eps fraction of the nodes). Secondly, a c-approx in G' yields a c-approx in G. This is because the c-approx must have each node of G' in its own cluster. which is what we wanted. ■

The following result shows that obtaining an $O(\epsilon)$-close clustering is unlikely under the $(1, \epsilon)$-property, even though it is possible if we assume the $(1.01, \epsilon)$-property, as we have shown.

Theorem 18 *For any contant $\epsilon < 1/e$, it is NP-hard to find a clustering of error $O(\epsilon)$ for the k-median and k-means problem under the $(1, \epsilon)$-property.*

Proof: We start from the hard instances of k-median instances arising from from max-k-coverage (using edges of cost 1 and 3): the reduction implies that it is hard to distinguish cases when there are k medians that cover all the points at distance 1 (the "yes" case), from the instance where any set of k medians covers at least $(1/e - \delta)$-fraction of the points at distance 3 (the "no" case) for any constant $\delta > 0$. Let us add a small amount of noise to make a unique optimal solution and call this the target; the uniqueness of the optimal solution ensures that we satisfy the $(1, \epsilon)$ assumption.

Now, in the "yes" case, any clustering with error $c\epsilon$ will have cost at most $n[(1 - c\epsilon) + 3c\epsilon]$. This would be less than the cost of the optimal solution in the "no" case (which is still at least $n[(1 - 1/e + \delta) + 3(1/e - \delta)]$) as long as $\epsilon \leq 1/e - \delta$, and would allow us to distingush the "yes" and "no" instances. This completes the proof for the k-median case; this proof can be altered slightly to work for the k-means problem. ■

Lemma 19 *Let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be a k-clustering in which each cluster is nonempty, and let $R = \{(x_1, j_1), (x_2, j_2), \ldots, (x_t, j_t)\}$ be a set of t reassignments of points to clusters (assume that $x_i \notin C_{j_i}$ for all i). Then there must exist a set $R' \subseteq R$ of size at least $t/3$ such that for any $R'' \subseteq R'$, the clustering $\mathcal{C}''$ produced by reassigning points in R'' has distance at least $|R''|$ from $\mathcal{C}$.*

Before proving the lemma, note that we cannot necessarily just choose $R' = R$ because, for instance, it could be that R reassigns all points in C_1 to C_2 and all points in C_2 to C_1: in this case, performing all reassignments in R produces the exact same clustering as we started with (just with different indices). Instead, we need to ensure that each reassignment in R' has an associated certificate ensuring that if implemented, it will increase the resulting distance from $\mathcal{C}$. Note also that if $\mathcal{C}$ consists of 3 singleton clusters: $C_1 = \{x\}, C_2 = \{y\}, C_3 = \{z\}$, and if $R = \{(x, 2), (y, 3), (z, 1)\}$, then any subset of reassignments in R will produce a clustering that differs in at most one element from $\mathcal{C}$; thus, the factor of 3 is tight.

Proof: The proof is based on the following lower-bounding technique. Given two clusterings $\mathcal{C}$ and $\mathcal{C}'$, suppose we can produce a list L of disjoint subsets $S_1, S_2, \ldots$, such that for each i, all points in S_i are in the *same* cluster in one of $\mathcal{C}$ or $\mathcal{C}'$ and they are all in *different* clusters in the other. Then $\mathcal{C}$ and $\mathcal{C}'$ must have pointwise distance at least $\sum_i (|S_i| - 1)$. In particular, any bijection σ on the indices can have agreement between $\mathcal{C}$ and $\mathcal{C}'$ on at most one point from each S_i.

We begin as follows. While there exists a reassignment $(x, j) \in R$ such that x is in a cluster $C(x)$ with at least 3 points: choose an arbitrary point $y \in C(x)$ and add $\{x, y\}$ to L, add (x, j) to R', and remove (x, j) from R as well as any reassignment involving y if one exists. In addition, remove x and y from the point set S. This process guarantees that all pairs added to L are disjoint, and we remove at most twice as many reassignments from R as we add to R'. (So, if R becomes empty, we will have achieved our desired result with $|R'| = t/2$). Moreover, because we only perform this step if $|C(x)| \geq 3$, this process does not produce any empty clusters.

We now have that for all reassignments $(x, j) \in R$, x is in a singleton or doubleton cluster. Let R_{single} be the set of reassignments $(x, j) \in R$ such that x is in a singleton cluster. Viewing these reassignments as directed edges, R_{single} forms a graph on the clusters C_i where each node has outdegree ≤ 1. Therefore, each component of this graph must be an arborescence with possibly one additional edge from the root. We now proceed as follows. While R_{single} contains a source (a node of outdegree 1 and indegree 0), choose an edge (x, j) such that (a) x is a source and (b) for all other edges (y, j), y is either a source or part of a cycle. We then consider two cases:

1. Node j is not a sink in R_{single}: that is, there exists an edge $(z, j_z) \in R_{single}$ for $z \in C_j$. In this case, we add to R' the edge (x, j) and all other edges (y, j) such that y is a source, and we remove from R (and from R_{single}) the edges (z, j_z), (x, j), and all edges (y, j) (including the at most one edge (y, j) such that y is part of a cycle). We then add to L the set $\{x\} \cup \{z\} \cup \{y : (y, j)$ was just added to $R'\}$ and remove these points from S. Note that the number of edges

removed from R is at most the number of edges added to R' plus 2, giving a factor of 3 in the worst case. Note also that we maintain the invariant that no edges in R_{single} point to empty clusters, since we deleted all edges into C_j, and the points x and y added to L were sources in R_{single}.

2. Otherwise, node j is a sink in R_{single}. In this case, we add to R' the edge (x, j) along with all other edges $(y, j) \in R_{single}$ (removing those edges from R and R_{single}). We choose an arbitrary point $z \in C_j$ and add to L the set $\{x\} \cup \{z\} \cup \{y : (y, j)$ was just added to $R'\}$, removing those points from S. In addition, we remove from R all (at most two) edges exiting from C_j (we are forced to remove any edge exiting from z since z was added to L, and there might be up to one more edge if C_j is a doubleton). Again, the number of edges removed from R is at most the number of edges added to R' plus 2, giving a factor of 3 in the worst case.

At this point, if R_{single} is nonempty, its induced graph must be a collection of disjoint cycles. For each such cycle, we choose every other edge (half the edges in an even-length cycle, at least 1/3 of the edges in an odd cycle), and for each edge (x, j) selected, we add (x, j) to R', remove (x, j) and (z, j_z) for $z \in C_j$ from R and R_{single}, and add the pair $\{x, z\}$ to L.

Finally, R_{single} is empty and we finish off any remaining doubleton clusters using the same procedure as in the first part of the argument. Namely, while there exists a reassignment $(x, j) \in R$, choose an arbitrary point $y \in C(x)$ and add $\{x, y\}$ to L, add (x, j) to R', and remove (x, j) from R as well as any reassignment involving y if one exists.

By construction, the set R' has size at least $|R|/3$, and the set L ensures that each reassignment in R' increases the resulting distance from $\mathcal{C}$ as desired. ■