Center-based Clustering under Perturbation Stability[☆]

Pranjal Awasthi

Carnegie Mellon University, Pittsburgh, PA 15213-3891

Avrim Blum

Carnegie Mellon University, Pittsburgh, PA 15213-3891

Or Sheffet

Carnegie Mellon University, Pittsburgh, PA 15213-3891

Abstract

Clustering under most popular objective functions is NP-hard, even to approximate well, and so unlikely to be efficiently solvable in the worst case. Recently, Bilu and Linial [11] suggested an approach aimed at bypassing this computational barrier by using properties of instances one might hope to hold in practice. In particular, they argue that instances in practice should be stable to small perturbations in the metric space and give an efficient algorithm for clustering instances of the Max-Cut problem that are stable to perturbations of size $O(n^{1/2})$. In addition, they conjecture that instances stable to as little as $O(1)$ perturbations should be solvable in polynomial time. In this paper we prove that this conjecture is true for any *center-based* clustering objective (such as k-median, k-means, and k-center). Specifically, we show we can efficiently find the optimal clustering assuming only stability to factor-3 perturbations of the underlying metric in spaces without Steiner points, and stability to factor $2+\sqrt{3}$ perturbations for general metrics. In particular, we show for such instances that the popular Single-Linkage algorithm combined with dynamic programming will find the optimal clustering. We also present NP-hardness results under a weaker but related condition.

Keywords: Clustering, k-median, k-means, Stability Conditions

1. Introduction

Problems of clustering data arise in a wide range of different areas – clustering proteins by function, clustering documents by topic, and clustering images by who or what is in them, just to name a few. In this paper we focus on the popular class of center based clustering objectives, such as k-median, k-center and k-means. Under these objectives we not only partition the data into k subsets, but we also assign k special points, called the *centers*, one in each cluster. The quality of a solution is then measured as a function of the distances between the data points and their centers. For example, in the k-median objective, the goal is to minimize the sum of distances of all points from their nearest center, and in the k-means objective, we minimize the sum of the same distances squared. As these are **NP**-hard problems [17, 18, 13], there has been substantial work on approximation algorithms [2, 3, 8, 12, 19, 14] with both upper and lower bounds on approximability of these and other objective functions. Note that we are especially interested in the case that k is part of the input and *not* a constant.

Recently, Bilu and Linial [11], focusing on the Max-Cut problem [16], proposed considering instances where the optimal clustering is optimal not only under the given metric, *but also under any bounded multiplicative perturbation of the given metric.* This is motivated by the fact that in practice, distances between data points are typically just the result of some heuristic measure (e.g., edit-distance between strings or Euclidean distance in some feature space) rather than true “semantic distance” between objects. Thus, unless the optimal solution on the given distances is correct by pure luck, it likely is correct on small perturbations of the given distances as well. Bilu and Linial [11] analyze Max-Cut instances of this type and show that for instances that are stable to perturbations of multiplicative factor roughly $O(n^{1/2})$, one can retrieve the optimal Max-Cut in polynomial time. However, they conjecture that stability up to only *constant* magnitude perturbations should be enough to solve the problem in polynomial time. In this paper we show that this conjec-

[☆] This work was supported in part by the National Science Foundation under grant CCF-0830540, as well as by CyLab at Carnegie Mellon under grants DAAD19-02-1-0389 and W911NF-09-1-0273 from the Army Research Office.

Email addresses: pawasthi@cs.cmu.edu (Pranjal Awasthi), avrim@cs.cmu.edu (Avrim Blum), osheffet@cs.cmu.edu (Or Sheffet)

ture is indeed true for k-median and k-means objectives and in fact for any well-behaved center-based objective function (see Definition 1.3).

1.1. Main Result

First, let us formally define the notion due to [11] of stability under multiplicative perturbations, stated in this context.

Definition 1.1. *Given a metric (S, d), and $\alpha > 1$, we say a function $d' : S \times S \to \mathbb{R}_{\geq 0}$ is an* α-perturbation *of d, if for any $x, y \in S$ it holds that*

$$d(x, y) \leq d'(x, y) \leq \alpha d(x, y)$$

Note that d' may be any non-negative function, and need not be a metric.

Definition 1.2. *Suppose we have a clustering instance composed of n points residing in a metric (S, d) and an objective function Φ we wish to optimize. We call the clustering instance* α-perturbation resilient *for Φ if for any d' which is an α-perturbation of d, the (only) optimal clustering of (S, d') under Φ is identical, as a partition of points into subsets, to the optimal clustering of (S, d) under Φ.*

We will in particular be concerned with *separable*, *center-based* clustering objectives Φ (which include k-median, k-means, and k-center among others).

Definition 1.3. *A clustering objective is* center-based *if the optimal solution can be defined by k points $c_1^*, \ldots, c_k^*$ in the metric space called* centers *such that every data point is assigned to its nearest center. Such a clustering objective is* separable *if it furthermore satisfies the following two conditions:*

- *The objective function value of a given clustering is either a (weighted) sum or the maximum of the individual cluster scores.*
- *Given a proposed single cluster, its score can be computed in polynomial time.*

Our main result is that we can efficiently find the optimal clustering for perturbation-resilient instances of separable center-based clustering objectives. In particular, we get an efficient algorithm for 3-perturbation-resilient instances when the metric S is defined only over data points, and for $(2 + \sqrt{3})$-perturbation-resiliant instances for general metrics.

Theorem 1.4. *For $\alpha \geq 3$ (in the case of finite metrics defined only over the data) or $\alpha \geq 2 + \sqrt{3}$ (for general metrics), there is a polynomial-time algorithm that finds the optimal clustering of α-perturbation resilient instances for any given separable center-based clustering objective.*

The algorithm, described in Section 2.2, turns out to be quite simple. As a first step, it runs the classic single-linkage algorithm, but unlike the standard approach of halting when k clusters remain, it runs the algorithm until *all* points have been merged into a single cluster and keeps track of the entire tree-on-clusters produced.[1] Then, the algorithm's second step is to apply dynamic programming to this hierarchical clustering to identify the best k-clustering that is present within the tree. Applying a result of Balcan et al. [6] we show that the resulting clustering obtained is indeed the optimal one. Albeit being very different, our approach resembles, in spirit, the work of Bartal [7], Abraham et al. [1] and Räcke [22] in the sense that we reduce the problem of retrieving an optimal solution from a general instance to a tree-like instance (where it is poly-time solvable).

Our algorithms use only a weaker property, which we call center-proximity (see Section 2.1), that is implied by perturbation-resilience. We then complement these results with a lower bound showing that for the problem of k-median on general metrics, for any $\epsilon > 0$, there exist **NP**-hard instances that satisfy $(3 - \epsilon)$-center proximity.[2]

1.2. Related work

There have been a number of investigations of different notions of stability for the problem of clustering. For example, Ostrovsky et al. [21] consider a k-means instance to be stable if the optimal k-clustering is substantially cheaper than the optimal $(k - 1)$-clustering under this objective. They present an efficient algorithm for finding near-optimal k-means clusterings when this gap is large, and these results were subsequently strengthened to apply to smaller gaps in [4]. Balcan et al. [5] consider instead a clustering instance to be stable if good approximations to the given objective are guaranteed to be close, as clusterings, to a desired ground-truth partitioning. This is motivated by the fact that when the true goal is to match some unknown correct answer (e.g., to correctly cluster proteins by their function), this is an implicit assumption already being made when viewing approximation ratio as a good performance measure. Balcan et al. [5] show that in fact this condition can be used to bypass approximation hardness results for a number of clustering objectives including k-median and k-means. In particular, if all $(1+\alpha)$-approximations to the objective are δ-close to the desired clustering in terms of how points are partitioned, they show one can efficiently get $O(\delta/\alpha)$-close to the desired clustering. Ben-David et al. [10, 9] consider a notion of

[1] The example depicted in Figure 3 proves that indeed, halting the Single-Linkage algorithm once k clusters are formed may fail on certain α-perturbation resilient instances.

[2] We note that while our belief was that allowing Steiner points in the lower bound was primarily a technicality, Balcan et al. (M.F. Balcan, personal communication) have recently shown this is not the case, giving a clever algorithm that finds the optimal k-median clustering for metrics without Steiner points when $\alpha = 1 + \sqrt{2}$.

stability of a clustering *algorithm*, which is called stable if it outputs similar clusters for different sets of n input points drawn from the same distribution. For k-means, the work of Meila [20] discusses the opposite direction – classifying instances where an approximated solution for k-means is close to the target clustering.

2. Proof of Main Theorem

2.1. Properties of Perturbation Resilient Instances

We begin by deriving other properties which every α-perturbation resilient clustering instance must satisfy.

Definition 2.1. *Let $p \in S$ be an arbitrary point, let c_i^* be the center p is assigned to in the optimal clustering, and let $c_j^* \neq c_i^*$ be any other center in the optimal clustering. We say a clustering instance satisfies the* α-center proximity *property if for any p it holds that*

$$d(p, c_j^*) > \alpha d(p, c_i^*)$$

Fact 2.2. *If a clustering instance satisfies the α-perturbation resilience property, then it also satisfies the α-center proximity property.*

Proof. Let C_i^* and C_j^* be any two clusters in the optimal clustering and pick any $p \in C_i^*$. Assume we blow up all the pairwise distances within cluster C_i^* by a factor of α. As this is a legitimate perturbation of the metric, it still holds that the optimal clustering under this perturbation is the same as the original optimum. Hence, p is still assigned to the same cluster. Furthermore, since the distances within C_i^* were all changed by the same constant factor, c_i^* will still remain an optimal center of cluster i. The same holds for cluster C_j^*. It follows that even in this perturbed metric, p prefers c_i^* to c_j^*. Hence $\alpha d(p, c_i^*) = d'(p, c_i^*) < d'(p, c_j^*) = d(p, c_j^*)$. □

Corollary 2.3. *For every point p and its center c_i^*, and for every point p' from a different cluster, it follows that $d(p, p') > (\alpha - 1)d(p, c_i^*)$.*

Proof. Denote by c_j^* the center of the cluster that p' belongs to. Now, consider two cases. Case (a): $d(p', c_j^*) \geq d(p, c_i^*)$. In this case, by triangle inequality we get that $d(p, p') \geq d(p', c_i^*) - d(p, c_i^*)$. Since the data instance is stable to α-perturbations, Fact 2.2 gives us that $d(p', c_i^*) > \alpha d(p', c_j^*)$. Hence we get that $d(p, p') > \alpha d(p', c_j^*) - d(p, c_i^*) \geq (\alpha - 1)d(p, c_i^*)$. Case (b): $d(p', c_j^*) < d(p, c_i^*)$. Again by triangle inequality we get that $d(p, p') \geq d(p, c_j^*) - d(p', c_j^*) > \alpha d(p, c_i^*) - d(p', c_j^*) > (\alpha - 1)d(p, c_i^*)$. □

A key ingredient in the proof of Theorem 1.4 is the *tree-clustering* formulation of Balcan et. al [6]. In particular, we prove that if an instance satisfies α-center proximity for $\alpha \geq 3$ (in the case of finite metrics without Steiner points) or for $\alpha \geq 2 + \sqrt{3}$ (for general metrics) then it also satisfies the "min-stability property" (defined below). The min-stability property, as shown in the full version of [6], is sufficient (and necessary) for the Single-Linkage algorithm to produce a tree on clusters such that the optimal clustering forms a pruning of this tree. In order to define the "min-stability" property, we first introduce the following notation. For any two subsets $A, B \subset S$, we denote the minimum distance between A and B as $d_{\min}(A, B) = \min\{d(a, b) \mid a \in A, b \in B\}$.

Definition 2.4. *A clustering instance satisfies the* min-stability property *if for any two clusters C and C' in the optimal clustering, and any subset $A \subsetneq C$, it holds that $d_{\min}(A, C \setminus A) \leq d_{\min}(A, C')$.*

In words, the min-stability property means that for any set A that is a strict subset of some cluster C in the optimal clustering, the closest point to A is a point from $C \setminus A$, and not from some other cluster. The next two lemmas lie at the heart of our algorithm.

Lemma 2.5. *A clustering instance in which centers must be data points that satisfies α-center proximity for $\alpha \geq 3$ (for a center-based clustering objective), also satisfies the min-stability property.*

Proof. Let C_i^*, C_j^* be any two clusters in the target clustering. Let A and A' be any two subsets s.t. $A \subsetneq C_i^*$ and $A' \subseteq C_j^*$. Let $p \in A$ and $p' \in A'$ be the two points which obtain the minimum distance $d_{\min}(A, A')$. Let $q \in C_i^* \setminus A$ be the nearest point to p. Also, denote by c_i^* and c_j^* the centers of clusters C_i^* and C_j^* respectively.

For the sake of contradiction, assume that $d_{\min}(A, C_i^* \setminus A) \geq d_{\min}(A, A')$. Suppose $c_i^* \notin A$. This means that $d(p, p') = d_{\min}(A, A') \leq d_{\min}(A, C_i^* \setminus A) \leq d(p, c_i^*)$. As $\alpha \geq 3$, this contradicts Corollary 2.3.

Thus we may assume $c_i^* \in A$. It follows that $d(q, c_i^*) \geq d(p, p') > (3-1)d(p, c_i^*) = 2d(p, c_i^*)$, so $d(p, c_i^*) < d(q, c_i^*)/2$. We therefore have that $d(p', c_i^*) \leq d(p, p') + d(p, c_i^*) \leq 3d(q, c_i^*)/2$. This implies that $d(p', c_j^*) < d(p', c_i^*)/\alpha < d(q, c_i^*)/2$, and thus $d(q, c_j^*) \leq d(q, c_i^*) + d(c_i^*, p) + d(p, p') + d(p', c_j^*) < 3d(q, c_i^*) \leq \alpha d(q, c_i^*)$. This contradicts Fact 2.2. □

Lemma 2.6. *A clustering instance in which centers need not be data points that satisfies α-center proximity for $\alpha \geq 2 + \sqrt{3}$ (for a center-based clustering objective), also satisfies the min-stability property.*

Proof. As in the proof of Lemma 2.5, let C_i^*, C_j^* be any two clusters in the target clustering and let A and A' be any two subsets s.t. $A \subsetneq C_i^*$ and $A' \subseteq C_j^*$. Let $p \in A$ and $p' \in A'$ be the two points which obtain the minimum distance $d_{\min}(A, A')$ and let $q \in C_i^* \setminus A$ be the nearest point to p. Also, as in the proof of Lemma 2.5, let c_i^* and c_j^* denote the centers of clusters C_i^* and C_j^* respectively (though these need not be datapoints).

By definition of center-proximity, we have the following inequalities:

$$\begin{array}{rcll} d(p,p') + d(p',c_j^*) & > & \alpha d(p,c_i^*) & \text{[c.p. applied to } p\text{]} \\ d(p,p') + d(p,c_i^*) & > & \alpha d(p',c_j^*) & \text{[c.p. applied to } p'\text{]} \\ d(p,p') + d(p',c_j^*) + d(p,q) & > & \alpha(d(q,p) - d(p,c_i^*)) & \\ \multicolumn{4}{r}{\text{[center proximity applied to } q \text{ and triangle ineq.]}} \end{array}$$

Multiplying the first inequality by $1 - \frac{1}{\alpha+1} - \frac{1}{\alpha-1}$, the second by $\frac{1}{\alpha+1}$, the third by $\frac{1}{\alpha-1}$, and summing them together we get

$$d(p,p') > \tfrac{\alpha^2-4\alpha+1}{\alpha-1} d(p,c_i^*) + d(q,p),$$

which for $\alpha \geq 2 + \sqrt{3}$ implies $d(p,p') > d(q,p)$ as desired. □

2.2. The Algorithm

As mentioned, Balcan et al [6] show (proof also given in Appendix A for completeness) that if an instance satisfies min-stability, then the tree on clusters produced by the single-linkage algorithm contains the optimal clustering as some k-pruning of it. The single-linkage algorithm produces a tree by starting with n clusters of size 1 (viewed as leaves), and at each step merging the two clusters C,C' minimizing $d_{\min}(C,C')$ (viewing the merged cluster as their parent) until only one cluster remains. Given the structural results proven above, our algorithm (see Figure 1) simply uses this clustering tree and finds the best k-pruning using dynamic programming.

Proof of Theorem 1.4. By Lemmas 2.5 and 2.6, the data satisfies the min-stability property, which as shown in [6] is sufficient to guarantee that some pruning of the single-linkage hierarchy is the target clustering. We then find the optimal clustering using dynamic programming by examining k-partitions laminar with the single-linkage clustering tree. The optimal k-clustering of a tree-node is either the entire subtree as one cluster (if $k = 1$), or the minimum over all choices of k_1-clusters over its left subtree and k_2-clusters over its right subtree (if $k > 1$). Here k_1, k_2 are positive integers, such that $k_1 + k_2 = k$. Therefore, we just traverse the tree bottom-up, recursively solving the clustering problem for each tree-node. By assumption the clustering objective is separable, so each step including the base-case can be performed in polynomial time. For the case of k-median in a finite metric, for example, one can maintain a $n \times O(n)$ table for all possible centers and all possible clusters in the tree, yielding a running time of $O(n^2 + nk^2)$. For the case of k-means in Euclidean space, one can compute the cost of a single cluster by computing the center as just the average of all its points. In general, the overall running time is $O(n(k^2 + T(n)))$, where $T(n)$ denotes the time it takes to compute the cost of a single cluster. □

2.3. Some Natural Barriers

We complete this section with a discussion of barriers of our approach. First, our algorithm indeed fails on some finite metrics that are $(3-\epsilon)$-perturbation resilient. For example, consider the instance shown in Figure 2. In this instance, the clustering tree produced by single-linkage is not laminar with the optimal k-median clustering. It is easy to check that this instance is resilient to α-perturbations for any $\alpha < 3$ (c and c' should each be viewed as many points at the same location).

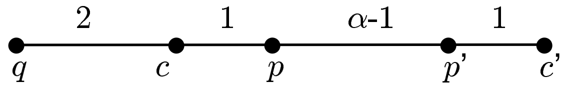

Figure 2: A finite metric k-median instance with $2 < \alpha < 3$ where our algorithm fails. The optimal 2-median clustering is $\{c,p,q\}, \{c',p'\}$. In contrast, when we run our algorithm on this instance, single linkage first connects $\{c,p\}$ with $\{c',p'\}$, and only then merges these 4 points with q.

Second, observe that our analysis, though emanating from perturbation resilience, only uses center proximity. We next show that for general metrics, one cannot hope to solve (in poly-time) k-median instances satisfying α-center proximity for $\alpha < 3$. This is close to our upper bound of $2 + \sqrt{3}$ for general metrics.

Theorem 2.7. *For any $\alpha < 3$, the problem of solving k-median instances over general metrics that satisfy α-center proximity is* **NP***-hard.*

Proof. The proof of Theorem 2.7 follows from the classical reduction of Max-k-Coverage to k-median. In the Max-k-Coverage problem we are given as input a number k and collection of sets $S_1, S_2, \cdots, S_m$, where each set covers a subset of elements from universe $\mathcal{U}$. The goal is to choose k sets such that their union covers the maximum number of elements in $\mathcal{U}$. The reduction to k-median is achieved by creating a bipartite graph where the right-hand side vertices represent the elements in the ground set; the left-hand side vertices represent the given subsets; and the distance between the set-vertex and each element-vertex is 1, if the set contains that element. Using shortest-path distances, it follows that the distance from any element-vertex to a set-vertex to which it does not belong to is at least 3. Using the fact that the **NP**-hardness results for Max-k-Coverage holds for disjoint sets (i.e. the optimal solution of Yes-instances is composed of k disjoint sets, see [15]), the α-center proximity property follows. □

Lastly, we comment that using Single-Linkage in the usual way (namely, stopping when there are k clusters remaining) is *not* sufficient to produce a good clustering. We demonstrate this using the example shown in Figure 3. Observe, in this instance, since C contains significantly less points than A,B, or D, this instance is stable – even

```
1. Run Single-Linkage until only one cluster remains, producing the entire tree on clusters.
2. Find the best $k$-pruning of the tree by dynamic programming using the equality

   $\text{best-}k\text{-pruning}(T) = \min_{0<k'<k} \{\text{best-}k'\text{-pruning}(T\text{'s left child}) + \text{best-}(k-k')\text{-pruning}(T\text{'s right child})\}$
```

Figure 1: Algorithm to find the optimal k-clustering of instances satisfying α-center proximity. The algorithm is described for the case (as in k-median or k-means) that Φ defines the overall score to be a sum over individual cluster scores. If it is a maximum (as in k-center) then replace "+" with "max" above.

if we perturb distances by a factor of 3, the cost of any alternative clustering is higher than the cost of the optimal solution. However, because $d(A, C) > d(B, D)$, it follows that the usual version of Single-Linkage will unite B and D, and only then A and C. Hence, if we stop the Single-Linkage algorithm at $k = 3$ clusters, we will not get the desired clustering.

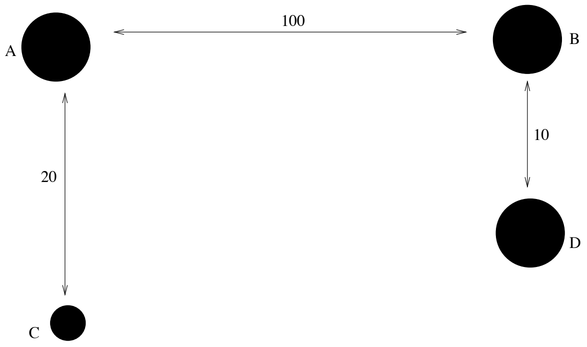

Figure 3: An example showing failure of the usual version of Single-Linkage. The instance is composed of 4 components, each with inner-distance ϵ and outer-distance as described in the figure. However, components A, B and D each contain 100 points, whereas component C has only 10 points. The optimal 3-median clustering consists of 3 clusters: $\{A, C\}, \{B\}, \{D\}$ and has cost $\mathsf{OPT} = 200 + 300\epsilon$.

3. Open Problems

There are several natural open questions left by this work. First, can one reduce the perturbation factor α needed for efficient clustering? As mentioned earlier, recently Balcan et al. (M.F. Balcan, personal communication) have given a very interesting algorithm that reduces the $\alpha = 3$ factor needed by our algorithm for finite metrics to $1 + \sqrt{2}$. Can one go farther, perhaps by using further implications of perturbation-resilience beyond center-proximity? Alternatively, if one cannot find the *optimal* clustering for small values of α, can one still find a near-optimal clustering, of approximation ratio better than what is possible on worst-case instances?

In a different direction, one can also consider relaxations of the perturbation-resilience condition. For example, Balcan et al. (personal communication) also consider instances that are "mostly resilient" to α-perturbations: under any α-perturbation of the underlying metric, no more than a δ-fraction of the points get mislabeled under the optimal solution. For sufficiently large constant α and sufficiently small constant δ, they present algorithms that get good approximations to the objective under this condition. A different kind of relaxation would be to consider a notion of *resilience to perturbations on average*: a clustering instance whose optimal clustering is likely not to change, assuming the perturbation is *random* from some suitable distribution. Can this weaker notion be used to still achieve positive guarantees?

Appendix A. Min-stability and single-linkage

We include here for completeness a proof of the following result from [6].

Theorem A.1 *If a clustering instance with optimal clustering $\mathcal{C}$ satisfies the min-stability property then the single-linkage algorithm will produce a tree such that $\mathcal{C}$ is a pruning of the tree.*

Proof. The single-linkage algorithm starts with n clusters of size 1 as leaves of the tree and at each step creates a new node in the tree by merging two clusters C and C' such that $d_{\min}(C, C')$ is minimized (viewing the new node as the parent of C and C'). In order to prove the theorem it is enough to show that at each step the resulting clustering is laminar with $\mathcal{C}$, i.e., each node is either a subset of a cluster in $\mathcal{C}$, equal to a cluster in $\mathcal{C}$ or a union of clusters in $\mathcal{C}$. This property is clearly maintained at the initial step. By induction assume that the current clustering is laminar and the algorithm decides to merge C and C'. We can assume that one of C or C' (say C) is a strict subset of a cluster C_r in $\mathcal{C}$, otherwise the laminarity property trivially holds. This implies that by the min-stability property, the closest point to C lies in $C_r \setminus C$. Therefore, if the algorithm merges C with C' then $C' \subset C_r$ as well. Hence the resulting clustering is still laminar. □

References

[1] Ittai Abraham, Yair Bartal, T-H. Hubert Chan, Kedar Dhamdhere Dhamdhere, Anupam Gupta, Jon Kleinberg, Ofer Neiman, and Aleksandrs Slivkins. Metric embeddings with relaxed guarantees. In *Proc. 46th Annual IEEE Symp. Foundations of Computer Science (FOCS)*, 2005.

[2] Sanjeev Arora, Prabhakar Raghavan, and Satish Rao. Approximation schemes for euclidean k-medians and related problems. In *Proc. 30th Annual ACM Symp. Theory of Computing (STOC)*, 1998.

[3] Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. Local search heuristic for k-median and facility location problems. In *Proc. 33rd ACM Symp. Theory of Computing (STOC)*, 2001.

[4] Pranjal Awasthi, Avrim Blum, and Or Sheffet. Stability yields a ptas for k-median and k-means clustering. In *Proc. 51st Annual IEEE Symp. Foundations of Computer Science (FOCS)*, 2010.

[5] Maria-Florina Balcan, Avrim Blum, and Anupam Gupta. Approximate clustering without the approximation. In *Proc. 19th Annual ACM-SIAM Symp. Discrete Algorithms (SODA)*, 2009.

[6] Maria-Florina Balcan, Avrim Blum, and Santosh Vempala. A discriminative framework for clustering via similarity functions. In *Proc. 40th Annual ACM Symp. Theory of Computing (STOC)*, 2008.

[7] Yair Bartal. On approximating arbitrary metrices by tree metrics. In *Proc. 30th Annual ACM Symp. Theory of Computing (STOC)*, 1998.

[8] Yair Bartal, Moses Charikar, and Danny Raz. Approximating min-sum k-clustering in metric spaces. In *Proc. 33rd Annual ACM Symp. Theory of Computing (STOC)*, 2001.

[9] Shai Ben-David, Dávid Pál, and Hans-Ulrich Simon. Stability of k-means clustering. In *COLT*, pages 20–34, 2007.

[10] Shai Ben-David, Ulrike von Luxburg, and Dávid Pál. A sober look at clustering stability. In Gábor Lugosi and Hans-Ulrich Simon, editors, *COLT*, volume 4005 of *Lecture Notes in Computer Science*, pages 5–19. Springer, 2006.

[11] Yonatan Bilu and Nati Linial. Are stable instances easy? 1st Symp. Innovations in Computer Science (ICS), 2010.

[12] Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proc. 31st Annual ACM Symp. Theory of Computing (STOC)*, 1999.

[13] Sanjoy Dasgupta. The hardness of k-means clustering, 2008.

[14] W. Fernandez de la Vega, Marek Karpinski, Claire Kenyon, and Yuval Rabani. Approximation schemes for clustering problems. In *Proc. 35th Annual ACM Symp. Theory of Computing (STOC)*, 2003.

[15] Uriel Feige. A threshold of $\ln n$ for approximating set cover. *JACM*, 45:314–318, 1998.

[16] Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1990.

[17] Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. In *Journal of Algorithms*, pages 649–657, 1998.

[18] Kamal Jain, Mohammad Mahdian, and Amin Saberi. A new greedy approach for facility location problems (extended abstract). In *Proc. 34th Annual ACM Symp. Theory of Computing (STOC)*, pages 731–740, 2002.

[19] Amit Kumar, Yogish Sabharwal, and Sandeep Sen. A simple linear time $(1+\epsilon)$ -approximation algorithm for k-means clustering in any dimensions. In *Proc. 45th Annual IEEE Symp. Foundations of Computer Science (FOCS)*, 2004.

[20] Marina Meilă. The uniqueness of a good optimum for k-means. In *Proc. 23rd International Conference on Machine Learning (ICML)*, pages 625–632, 2006.

[21] Rafail Ostrovsky, Yuval Rabani, Leonard J. Schulman, and Chaitanya Swamy. The effectiveness of Lloyd-type methods for the k-means problem. In *Proc. 47th Annual IEEE Symp. Foundations of Computer Science (FOCS)*, pages 165–176, 2006.

[22] Harald Räcke. Optimal hierarchical decompositions for congestion minimization in networks. In *Proc. 40th Annual ACM Symp. Theory of Computing (STOC)*, 2008.