Randomized Dimensionality Reduction for Facility Location and Single-Linkage Clustering

Shyam Narayanan [*1] **Sandeep Silwal** [*1] **Piotr Indyk** [1] **Or Zamir** [2]

Abstract

Random dimensionality reduction is a versatile tool for speeding up algorithms for high-dimensional problems. We study its application to two clustering problems: the facility location problem, and the single-linkage hierarchical clustering problem, which is equivalent to computing the minimum spanning tree. We show that if we project the input pointset X onto a random $d = O(d_X)$-dimensional subspace (where d_X is the doubling dimension of X), then the optimum facility location cost in the projected space approximates the original cost up to a constant factor. We show an analogous statement for minimum spanning tree, but with the dimension d having an extra $\log \log n$ term and the approximation factor being arbitrarily close to 1. Furthermore, we extend these results to approximating *solutions* instead of just their *costs*. Lastly, we provide experimental results to validate the quality of solutions and the speedup due to the dimensionality reduction. Unlike several previous papers studying this approach in the context of k-means and k-medians, our dimension bound does not depend on the number of clusters but only on the intrinsic dimensionality of X.

1. Introduction

Clustering is a fundamental problem with many applications in machine learning, statistics, and data analysis. Although many formulations of clustering are NP-hard in the worst case, many heuristics and approximation algorithms exist and are widely deployed in practice. Unfortunately, many of those algorithms suffer from large running times, especially if the input data sets are high-dimensional.

In order to improve the performance of clustering algorithms in high-dimensional spaces, a popular approach is to project the input point set into a lower-dimensional space and perform the clustering in the projected space. Reducing the dimension (say, from m to $d \ll m$) has multiple practical and theoretical advantages, including (i) lower storage space, which is linear in d as opposed to m; (ii) lower running time of the clustering procedure - the running times are often dominated by distance computations, which take time linear in the dimension; and (iii) versatility: one can use *any* algorithm or its implementation to cluster the data in the reduced dimension. Because of its numerous benefits, dimensionality reduction as a tool for improving algorithm performance has been studied extensively, leading to many theoretical tradeoffs between the projected dimension and the solution quality. A classic result in this area is the Johnson-Lindenstrauss (JL) lemma (1984) which (roughly) states that a random projection of a dataset $X \subseteq \mathbb{R}^m$ of size n onto a dimension of size $O(\log n)$ approximately preserves all pairwise distances. This tool has been subsequently applied to many clustering and other problems (see (Naor, 2018) and references therein).

Although the JL lemma is known to be tight (Larsen & Nelson, 2017) in general, better tradeoffs are possible for *specific* clustering problems. Over the last few years, several works (Boutsidis et al., 2010; Cohen et al., 2015; Becchetti et al., 2019; Makarychev et al., 2019) have shown that combining random dimensionality reduction with k-means leads to better guarantees than implied by the JL lemma. In particular, a recent paper by Makarychev, Makarychev, and Razenshteyn (2019) shows that to preserve the k-means cost up to an arbitrary accuracy, it suffices to project the input set X onto a dimension of size $O(\log k)$, as opposed to $O(\log n)$ guaranteed by the JL lemma. Since k can be much smaller than n, the improvement to the dimension bound can be substantial. However, when k is comparable to n, the improvement is limited. This issue is particularly salient for clustering problems with a variable number of clusters, where no a priori bound on the number of clusters exists.

In this paper we study randomized dimensionality reduction

[*]Equal contribution [1]Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, Cambridge, MA, USA [2]Institute of Advanced Study, Princeton, NJ, USA. Correspondence to: Shyam Narayanan <shyamsn@mit.edu>, Sandeep Silwal <silwal@mit.edu>, Piotr Indyk <indyk@mit.edu>, Or Zamir <orzamir@ias.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

over Euclidean space $\mathbb{R}^m$ in the context of two fundamental clustering problems with a variable number of clusters. In particular:

- Facility location (FL): given a set of points $X \subset \mathbb{R}^m$ and a facility opening cost, the goal is to open a subset $\mathcal{F} \subseteq X$ of facilities in order to minimize the total cost of opening the facilities plus the sum of distances from points in X to their nearest facilities (see Section 2 for a formal definition). Such cost functions are often used when the "true" number of clusters k is not known, see e.g., (Manning et al., 2009), section 16.4.1.
- Single-linkage clustering, or (equivalently) Minimum Spanning Tree (MST): given a set of points $X \subset \mathbb{R}^m$, the goal is to connect them into a tree in order to minimize the total cost of the tree edges. This is a popular variant of Hierarchical Agglomerative Clustering (HAC) that creates a hierarchy of clusters, see e.g., (Manning et al., 2009), section 17.2.

We remark that some papers, e.g., (Abboud et al., 2019) define approximate HAC operationally, by postulating that each step of the clustering algorithm must be approximately correct. However, there are other theoretical formulations of approximate HAC as well, e.g., (Dasgupta, 2016; Moseley & Wang, 2017). Since single-linkage clustering has a natural objective function induced by MST, defining approximate single-linkage clustering as approximate MST is a natural, even if not unique, choice.

Our Results Our main results show that, for both FL and MST, it is possible to project input point sets into low (sometimes even constant) dimension while provably preserving the quality of the solutions. Specifically, our theorems incorporate the *doubling dimension* d_X of the input datasets X. This parameter[1] measures the "intrinsic dimensionality" of X and can be much lower than its ambient dimension m. If X has size n, the doubling dimension d_X is always at most $\log n$, and is often much smaller. We show that random projections into dimension roughly proportional to d_X suffice in order to approximately preserve the solution quality. The specific bounds are listed in Table 1.

We distinguish between two types of guarantees. The first type states that the minimum *cost* of FL or MST is preserved by a random projection (with high probability) up to the specified factor. This guarantee is useful if the goal is to quickly estimate the optimal value. The second type states that a *solution* computed in the projected space induces a solution in the original space which approximates the best solution (in the original space) up to the specified approximation factor. This guarantee implies that one can find an approximately optimal clustering by mapping the data into low dimensions and clustering the projected data. To obtain the second guarantee, we need to assume that the solution in the projected space is either globally optimal (for MST) or locally optimal[2] (for FL). We note that these two types of guarantees are incomparable. In fact, for FL, our proofs of the cost and of the solution guarantees are substantially different. We also prove analogous theorems for the "squared" version of FL, where the distance between points is defined as the *square* of the Euclidean distance between them.

We complement the above results by showing that the conditions and assumptions in our theorem cannot be substantially reduced or eliminated. Specifically, for both FL and MST, we show that:

- The bounds on the projected dimension d in the theorems specified in the table must be at least $\Omega(d_X)$, as otherwise the approximation factors for both the cost and the solution become super-constant (Theorems 6.1, 6.2, 6.3)
- The assumptions that the solution in the projected space is (locally) optimal cannot be relaxed to "approximately optimal" (Lemmas 6.4, 6.5).

Also, we show that, in contrast to facility location and MST, one must project to $\Omega(\log k)$ dimensions for preserving both the cost and solution for k-means and k-medians clustering, even if the doubling dimension d_X is $O(1)$.

Finally, we present an experimental evaluation of the algorithms suggested by our results. Specifically, we show that both FL and MST, solving these problems in reduced dimension can reduce the running time by 1-2 orders of magnitude while increasing the solution cost only slightly. We also give empirical evidence that the doubling dimension of the input point set affects the quality of the approximate solutions. Specifically, we study two simple point sets of size n that have similar structure but very different doubling dimension values ($O(1)$ and $O(\log n)$, respectively). We empirically show that a good approximation of the MST can be found for the former point set by projecting it into much fewer dimensions than the latter point set.

Related Work There is a long line of existing work on approximating the solution of various clustering problems in metric spaces with small doubling dimensions (see (Frigstad et al., 2019; Gottlieb, 2015; Chan et al., 2018; Talwar, 2004)). The state of the art result is given in (Saulpic et al.,

[1] We formally define it in Section 2.

[2] Informally, a solution is locally optimal if opening any new facility does not decrease its cost. The formal definition is slightly more general, and is given in Section 3. Note that any solution found by local search algorithms such as that in (Mettu & Plaxton, 2000) satisfies this condition.

Figure 1. Number of dimensions d required for a random projection to provide a good clustering approximation

Problem	Proj. dimension d	Approx.	Cost/Solution	Reference
FL	$O(d_X)$	$O(1)$	Cost	Theorem 4.1
FL	$O(d_X)$	$O(1)$	Locally optimal solution	Theorem 4.2
MST	$O(1/\epsilon^2 \cdot (d_X \log(1/\epsilon) + \log\log n))$	$1+\epsilon$	Cost	Theorem 5.1
MST	$O(1/\epsilon^2 \cdot (d_X \log(1/\epsilon) + \log\log n))$	$1+\epsilon$	Optimal solution	Theorem 5.1

2019) where a near linear $(1+\epsilon)$-approximation algorithm is given for a variety of clustering problems. However, these runtimes have a doubly-exponential dependence on d which is proven to be unavoidable unless P = NP (Saulpic et al., 2019). For MST in spaces of doubling dimension d_X, it is known that an $(1+\epsilon)$-approximate solution can be computed in time $2^{O(d_X)} n \log n + \epsilon^{-O(d_X)} n$ (Gottlieb & Krauthgamer, 2013). To the best of our knowledge, none of these algorithms have been implemented.

In addition, the notion of doubling dimension has also been previously used to study algorithms for high dimensional problems such as the nearest neighbor search, see e.g., (Indyk & Naor, 2007; Clarkson, 2012; Har-Peled & Kumar, 2013). The paper (Indyk & Naor, 2007) is closest in spirit to our work, as it shows that, for a fixed point q and a data set X, a random projection into $O(d_X)$ dimensions approximately preserves the distance from q to its nearest neighbor in X with a "good" probability. If the probability of success was of the form $1 - 1/2n$, we could apply this statement to all (up to n) facilities in the solution simultaneously, which would prove our results. Unfortunately, the probability of failure is much higher than $1/n$, and therefore this approach fails. Nevertheless, our proofs use some of the lemmas developed in that work, as discussed in Section 2.

2. Preliminaries

Problem Definitions The *Euclidean Facility Location* problem is defined as follows: We are given a dataset $X \subset \mathbb{R}^m$ of n points and a nonnegative function $c : X \to \mathbb{R}$ that represents the cost of *opening* a facility at a particular point. The goal is to find a subset $\mathcal{F} \subseteq X$ that minimizes the objective $\text{cost}(\mathcal{F}) = \sum_{f\in\mathcal{F}} c(f) + \sum_{x\in X} D(x,\mathcal{F})$, where $D(x,\mathcal{F}) = \min_{f\in\mathcal{F}} \|x-f\|$. In this work we restrict our attention to the case that $\|\cdot\|$ is the Euclidean (ℓ_2) metric. The first term $\sum_{f\in\mathcal{F}} c(f)$ is referred to as the opening costs and the second term $\sum_{x\in X} D(x,\mathcal{F})$ is referred to as the connection costs. In this work, we also focus on the *uniform* version of facility location where all opening costs are the same. By re-scaling the points, we can further assume that $f(x) = 1$ for all $x \in X$. Therefore, throughout the paper, we focus on minimizing the following objective function:

$$\text{cost}(\mathcal{F}) = |\mathcal{F}| + \sum_{x\in X} \min_{f\in\mathcal{F}} \|x-f\|. \tag{1}$$

A set $\mathcal{F}$ of facilities is also referred to as a *solution* to the facility location problem.

The *Euclidean Minimum Spanning Tree* problem is defined as follows. Given a dataset $X \subset \mathbb{R}^m$ of n points, we wish to find a set $\mathcal{M}$ of edges (x,y) that forms a spanning tree of X and minimizes the following objective function:

$$\text{cost}(\mathcal{M}) = \sum_{(x,y)\in\mathcal{M}} \|x-y\|. \tag{2}$$

Properties of Doubling Dimension We parameterize our dimensionality reduction using *doubling dimension*, a measure of the intrinsic dimensionality of the dataset. The notion of doubling dimension holds for a general metric space X and is defined as follows. Let $B(x,r)$ denote the ball of radius r centered at $x \in X$, intersected with the points in X. Then the doubling *constant* λ_X is the smallest constant λ such that for all $x \in X$ and for all $r > 0$, there exists $S \subseteq X$ with $|S| \le \lambda$ such that $B(x,r) \subseteq \bigcup_{s\in S} B(s, r/2)$. The doubling *dimension* of X is is defined as $d_X := \log \lambda_X$. One can see that $\lambda_X \le |X|$, so $d_X \le \log |X|$. In this paper, we focus on the case that X is a subset of Euclidean space $\mathbb{R}^m$.

Dimension Reduction In this paper we define a random projection as follows.

Definition 2.1. *A random projection from $\mathbb{R}^m$ to $\mathbb{R}^d$ is a linear map G with i.i.d. entries drawn from $\mathcal{N}(0, 1/d)$.*

The following dimensionality reduction result related to doubling dimension was proven in (Indyk & Naor, 2007). Informally, the lemma below states that a random projection of X onto a dimension $O(d_X)$ subspace does not 'expand' X very much.

Lemma 2.2 (Lemma 4.2 in (Indyk & Naor, 2007))**.** *Let $X \subseteq B(0,1)$ be a subset of the m-dimensional Euclidean unit ball, and let G be a random projection from $\mathbb{R}^m$ to $\mathbb{R}^d$. Then there exist universal constants $c, C > 0$ such that for $d \ge C \cdot d_X + 1$ and $t > 2$, $\Pr(\exists x \in X,\ \|Gx\| \ge t) \le \exp(-cdt^2)$.*

For our proofs, we will need some additional preliminary results on random projections, which are deferred to Supplementary Section A.

3. Local Optimality for Facility Location

We now define the notion of a locally optimal solution for facility location. As stated in the introduction, this notion plays a key role in our approximation guarantees. Before we present our criterion for local optimality, we begin by discussing the Mettu Plaxton (MP) algorithm, an approximation algorithm for the facility location problem. The MP approximation algorithm gives a useful geometric quantity to understand the facility location problem.

3.1. Approximating the Cost of Facility Location

For each $p \in X$, we associate with it a radius $r_p > 0$ which satisfies the relation

$$\sum_{q \in B(p,r_p) \cap X} (r_p - \|p - q\|) = 1. \tag{3}$$

It can be checked that a unique value r_p satisfying $1/n \leq r_p \leq 1$ exists for every p. The geometric interpretation of r_p is shown in Figure 2. This quantity was first defined by Mettu and Plaxton (2000), who proved that a simple greedy procedure of iteratively selecting facilities that lie in balls of radii $2r_p$ gives a 3 factor approximation algorithm for the facility location problem. For completeness, their algorithm is given in Supplementary Section B.

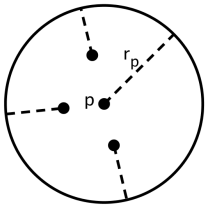


Figure 2. r_p is defined such that the dotted lines add to 1.

One of the main insights from Mettu and Plaxton's algorithm is that the sum of the radii r_p is a constant factor approximation to the cost of the optimal solution. This insight was first stated in (Badoiu et al., 2005) where it was used to design a sublinear time algorithm to approximate the cost of the facility location problem. In particular, we have the following result from (Badoiu et al., 2005) about the approximation property of the radii values.

Lemma 3.1 (Lemma 2 in (Badoiu et al., 2005))**.** *Let C_{OPT} denote the cost of the optimal facility location solution. Then $\frac{1}{4} \cdot C_{OPT} \leq \sum_{p \in X} r_p \leq 6 \cdot C_{OPT}$.*

For our purposes, we use the radii values to define a local optimality criterion for a solution to the facility location problem. Our local optimality criterion states that each point p must have a facility that is within distance $3r_p$.

Definition 3.2. *A solution $\mathcal{F}$ to the facility location problem is* locally optimal *if for all $p \in X$, $B(p, 3r_p) \cap \mathcal{F} \neq \emptyset$.*

We show in Lemma 3.3 that a solution that is not locally optimal can be improved, i.e. the objective function given in Eq. (1) can be improved, by adding p to the set of facilities. This implies that any global optimal solution must also be locally optimal, so requiring a solution of the facility location problem to be locally optimal is a *less restrictive* condition than requiring a solution to be globally optimal.

Lemma 3.3. *Let $\mathcal{F}$ be any collection of facilities. If there exists a $p \in X$ such that $B(p, 3r_p) \cap \mathcal{F} = \emptyset$ then* $\text{cost}(\mathcal{F} \cup \{p\}) < \text{cost}(\mathcal{F})$*, i.e., we can improve the solution.*

The proof of Lemma 3.3 is deferred to Supplementary Section B.

4. Dimension Reduction for Facility Location

4.1. Approximating the Optimal Facility Location Cost

In this subsection we show that we can *estimate* the cost of the global optimal solution for a point set X by computing the value of the radii after a random projection onto dimension $d = O(d_X)$. We do this by showing that for each p, the value of r_p can be approximated up to a constant multiplicative factor in $\mathbb{R}^d$, the lower dimension.

For each $p \in X$, let r_p and $\tilde{r}_p$ be the radius of p and Gp in $\mathbb{R}^m$ and $\mathbb{R}^d$, respectively, computed according to Eq. (3). Then we prove that $\mathbb{E}[\tilde{r}_p] = \Theta(r_p)$, where the expectation is over the randomness of the projection G.

This proof can be divided into showing $\mathbb{E}[\tilde{r}_p] = O(r_p)$ and $\mathbb{E}[\tilde{r}_p] = \Omega(r_p)$. Our proof strategy for the former is to use the concentration inequality in Lemma 2.2 to roughly say that points in $B(p, r_p) \cap X$ cannot get 'very far' away from p after a random projection. In particular, they must all still be at a distance $O(r_p)$ of p after the random projection. Then using the geometric definition of r_p given in (3) and Figure 2, we can say that the corresponding radii of Gp in $\mathbb{R}^d$ denoted as $\tilde{r}_p$ must then be upper bounded by $O(r_p)$. Our proof strategy for the latter is different in that our challenge is to show that points do not 'collapse' closer together. In more detail, for a fixed point p, we need to show that after a dimension reduction, many *new* points do not come inside a ball of radius $O(r_p)$ around the point Gp. An application of Theorem A.4 in Supplementary Section A, due to Indyk and Naor (2007), deals with this event.

By adding these expectations over each point p and applying Lemma 3.1, we can prove that the facility location cost is preserved under a random projection. Formally, we obtain the following theorem:

Theorem 4.1. *Let $X \subseteq \mathbb{R}^m$ and let G be a random projection from $\mathbb{R}^m$ to $\mathbb{R}^d$ for $d = O(d_X)$. Let $\mathcal{F}_m$ be the optimal solution in $\mathbb{R}^m$ and let $\mathcal{F}_d$ be the optimal solution for the dataset $GX \subseteq \mathbb{R}^d$. Then there exist constants $c, C > 0$ such that $c \cdot \text{cost}(\mathcal{F}_m) \leq \mathbb{E}[\text{cost}(\mathcal{F}_d)] \leq C \cdot \text{cost}(\mathcal{F}_m)$.*

The full proof of Theorem 4.1 and the lemmas bounding $\mathbb{E}[\tilde{r}_p]$ are deferred to Supplementary Section C.

4.2. Obtaining Facility Location Solution in Larger Dimension

As discussed in the introduction, for many applications, it is not enough to be able to approximate the *cost* of the optimal solution, but rather *obtain* a good solution.

In particular, we would like to perform dimensionality reduction on a dataset X, use some algorithm to solve facility location, and then have the guarantee that the quality of the solution we found is a good indicator of the quality of the solution in the original dimension. Furthermore, since optimally solving facility location in the smaller dimension might still be a challenging task, it is desirable to have a guarantee that a good solution (not necessarily the global optimum) will be a good solution in the larger dimension. We show in this section that this is indeed the case for *locally optimal* solutions.

Specifically, we show that the cost of a locally optimal solution found in $\mathbb{R}^d$ does not increase substantially when evaluated in the larger dimension. More formally, we prove the following theorem:

Theorem 4.2. *Let $X \subset \mathbb{R}^m$ and G be a random projection from $\mathbb{R}^m$ to $\mathbb{R}^d$ for $d = O(d_X \cdot \log(1/\epsilon)/\epsilon^2)$. Let $\mathcal{F}_d$ be a locally optimal solution for the dataset GX. Then, the cost of $\mathcal{F}_d$ evaluated in $\mathbb{R}^m$, denoted as $\text{cost}_m(\mathcal{F}_d)$, satisfies*

$$\mathbb{E}[\text{cost}_m(\mathcal{F}_d)] \leq |\mathcal{F}_d| + O\left(\sum_{p \in X} r_p\right) \leq \text{cost}_d(\mathcal{F}_d) + O(F),$$

where F is the optimal facility location cost of X in $\mathbb{R}^m$.

To describe the proof intuition, first note that the cost function defined in Eq. (1) has two components. One is the number of facilities opened, and the other is the connection cost. The first term is automatically preserved in the larger dimension since the number of facilities stays the same. Therefore, the main technical challenge is to show that if a facility is within distance $O(\tilde{r}_p)$ of a fixed point p in $\mathbb{R}^d$ (note that $\tilde{r}_p$ is calculated according to Eq. (3) in $\mathbb{R}^d$), then the facility must be within distance $O(r_p)$ in $\mathbb{R}^m$, the larger dimension. From here, one can use Lemma 3.1 to bound $\sum_{p \in X} r_p$ by $O(F)$, and the simple fact that $|\mathcal{F}_d| \leq \text{cost}_d(\mathcal{F}_d)$.

The proof of our main technical challenge relies on the careful balancing of the following two events. First, we control the value of the radius $\tilde{r}_p$ and show that $\tilde{r}_p \approx r_p$. In particular, we show that the probability of $\tilde{r}_p \geq k r_p$ for any constant k is exponentially decreasing in k. Next, we need to bound the probability that a 'far' point comes 'close' to p after the dimensionality reduction. While there exists a known result on this (e.g., Theorem A.4 in Supplementary Section A), we need a novel, more detailed result to quantify how close far points can come after the dimension reduction.

To study this in a more refined manner, we bucket the points in $X \setminus \{p\}$ according to their distance from p, with the ith level representing distance approximately i from p. We show that points in $X \setminus \{p\}$ that are in 'level' i do not shrink to a 'level' smaller than $O(\sqrt{i})$. Note that we need to control this even across all levels. To do this requires a chaining type argument which crucially depends on the doubling dimension of X. Finally, a careful combination of probabilities gives us our result.

The proof of Theorem 4.2 is deferred to Supplementary Section C.

Remark 4.3. *Our proof of Theorem 4.2 generalizes to the case of* arbitrary *opening costs c_p by changing the definition of r_p to be $\sum_{q \in B(p, r_p)} (r_p - \|x - q\|) = c_p$.*

4.3. Facility Location with Squared Costs

Facility location problem with squared costs is the following variant of facility location. Given a dataset $X \subset \mathbb{R}^m$, our goal is to find a subset $\mathcal{F} \subseteq X$ that minimizes the objective

$$\text{cost}(\mathcal{F}) = |\mathcal{F}| + \sum_{x \in X} \min_{f \in \mathcal{F}} \|x - f\|^2. \tag{4}$$

In contrast to (1), we are adding the *squared* distance from each point to its nearest facility in $\mathcal{F}$, rather than just the distance. This is comparable to k-means, whereas standard facility location is comparable to k-medians.

For the facility location problem with squared costs, we are again able to show that a random projection of X into $O(d_X)$ dimensions preserves the optimal cost up to an $O(1)$ factor, and that any locally optimal solution in the reduced dimension has its cost preserved in the original dimension. The formal statements and proofs are very similar to those of the standard facility location problem, and are deferred to Supplementary Section F.

5. Dimension Reduction for MST

In this section we demonstrate the effectiveness of dimensionality reduction for the minimum spanning tree (MST) problem. As in the case of facility location, we show that we can *estimate* the cost of the optimum MST solution by computing the MST in a lower dimension, and that the minimum spanning tree in the lower dimension is an *approximate* solution to the high-dimensional MST problem.

This time our approximations, both to the optimum cost and the optimum solution, can be $(1+\epsilon)$-approximations for any $\epsilon > 0$, as opposed to the constant factor approximations that we could guarantee for facility location. To formally state our theorem, for some spanning tree T of X, let $\text{cost}_X(T)$

be the sum of the lengths of the edges in T. Likewise, let $\text{cost}_{GX}(T)$ be the sum of the lengths of the edges in T, where distances are measured in the projected tree GX. Our main result is the following theorem:

Theorem 5.1. *For some positive integers m, d, let $X \subset \mathbb{R}^m$ be a point set of size n and let $G : \mathbb{R}^m \to \mathbb{R}^d$ be a random projection. Let $\mathcal{M}$ represent the minimum spanning tree of X, with $M = \text{cost}_X(\mathcal{M})$ and $\widetilde{\mathcal{M}}$ represent the minimum spanning tree of GX, with $\widetilde{M} = \text{cost}_{GX}(\widetilde{\mathcal{M}})$. Then, for some sufficiently large constant C_6, if $d \geq C_6 \cdot \epsilon^{-2} \cdot (\log \epsilon^{-1} \cdot d_X + \log\log n)$, the following are true:*

1. *The cost of the MST is preserved under projection with probability at least $\frac{9}{10}$. In other words, $\widetilde{M} \in [1-\epsilon, 1+\epsilon] \cdot M$.*
2. *The optimal projected MST $\widetilde{\mathcal{M}}$ is still an approximate MST in the original dimension with probability at least $\frac{9}{10}$. In other words, $\text{cost}_X(\widetilde{\mathcal{M}}) \in [1, 1+\epsilon] \cdot M$.*

Hence, we obtain a significantly stronger theoretical guarantee for preserving the MST than $d = \Theta(\epsilon^{-2} \log n)$, which is promised by the Johnson-Lindenstrauss Lemma, assuming that d_X and ϵ^{-1} are constant or very small.

Our main technical result in establishing Theorem 5.1 is the following crucial lemma, which will in fact allow us to prove both parts of the above theorem simultaneously.

Lemma 5.2. *For all notation as in Theorem 5.1, $\mathbb{E}[\text{cost}_X(\widetilde{\mathcal{M}}) - \text{cost}_{GX}(\widetilde{\mathcal{M}})] \leq O(\epsilon) \cdot M$.*

The proof strategy for Lemma 5.2 involves first dividing the edges of $\widetilde{\mathcal{M}}$ into levels based on their lengths, and bounding the difference between edge lengths (pre- and post- projection) in each level separately. To analyze a level consisting of the edges of length approximately t, we first partition the point set X (in the original dimension $\mathbb{R}^m$) into balls $B_1, \ldots, B_r$ of radius $c \cdot t$ for a small constant c, and show via chaining-type arguments that not too many pairs of balls that were originally far apart come close together after the random projection. Moreover, using Lemma 2.2, we show that almost all of the balls do not expand by much.

Therefore, there are not many *bad* pairs of balls (B_i, B_j), where (B_i, B_j) is bad if there exists $p \in B_i, q \in B_j$ where $\|p - q\|$ is much bigger than t but $\|Gp - Gq\|$ is approximately t. Now, assuming that none of the balls expand by much in the random projection, for any bad pair (B_i, B_j) and edges (p, q) and (p', q') with $p, p' \in B_i$ and $q, q' \in B_j$, we cannot have both edges in the minimum spanning tree of GX. This is because $\|Gp - Gq\|, \|Gp' - Gq'\| \approx t$, but since B_i and B_j have radius $c \cdot t$ and do not expand by much, we can improve the spanning tree by replacing (Gp, Gq) with either (Gp, Gp') or (Gq, Gq'). So, each bad pair can have at most 1 edge in $\widetilde{\mathcal{M}}$, the MST of GX. Overall, in each level, not too many edges in $\widetilde{\mathcal{M}}$ can shrink by much after the projection.

The full proofs of Lemma 5.2 and Theorem 5.1 are given in Supplementary Section D.

6. Lower Bounds for Projection Dimension

In this section, we state various lower bounds for the projection dimension d for both facility location clustering and minimum spanning tree. We also show that, in contrast to facility location, low doubling dimension does not actually help with dimensionality reduction for k-means or k-medians clustering. All proofs are deferred to Supplementary Section E.

In all results of this section, we think of X as a point set of size n in Euclidean space $\mathbb{R}^m$, and G as a random projection sending X to $GX \subset \mathbb{R}^d$. In this section, for FL, we always let $\mathcal{F}$ be the optimal set of facilities in X, with cost F, and $\widetilde{\mathcal{F}}$ be the optimal set of facilities in GX, with cost $\widetilde{F}$. We define $\mathcal{M}, M, \widetilde{\mathcal{M}}, \widetilde{M}$ analogously for MST. We use $o(1)$ to denote functions going to 0 as $n \to \infty$, and $\omega(1)$ to denote functions going to ∞ as $n \to \infty$, where $n = |X|$.

First, we show that the dependence of the projected dimension d on the doubling dimension d_X in Theorems 4.1, 4.2, and 5.1 are all required to obtain constant factor approximations for either the cost or the pullback solution. Namely, we show the following three theorems:

Theorem 6.1 (FL)**.** *Let $d = o(\log n)$. There exists X with doubling dimension $\Theta(\log n)$, such that with at least $2/3$ probability over $G : \mathbb{R}^m \to \mathbb{R}^d$, $\widetilde{F} = o(1) \cdot F$. Moreover, with probability at least $2/3$, $\widetilde{\mathcal{F}}$, when pulled back to X, has cost $\omega(1) \cdot F$ in the original dimension.*

Theorem 6.2 (MST)**.** *Let $d = o(\log n)$. There exists X with doubling dimension $\Theta(\log n)$, such that with probability at least $2/3$, $\widetilde{M} = o(1) \cdot M$.*

Theorem 6.3 (MST)**.** *Let $d = o(\log n)$. There exists X with doubling dimension $\Theta(\log n)$, such that with probability at least $2/3$, $\widetilde{\mathcal{M}}$, when pulled back to X, will have cost $\omega(1) \cdot M$.*

Next, we show that (local) optimality is required for Theorems 4.2 and 5.1, and cannot be replaced with *approximate* optimality. In other words, random projections to $o(\log n)$ dimensions do not necessarily preserve the set of approximate solutions for either facility location or MST, even for point sets of low doubling dimension. Namely, we show the following two lemmas:

Lemma 6.4 (FL)**.** *Let $d = o(\log n)$. There exists X with constant doubling dimension, such that with at least $2/3$ probability, there exists a $(1 + O(m^{-1/2d})) = (1 + o(1))$-*

approximate solution $\mathcal{F}'$ for GX whose total cost when pulled back to X is at least $\Omega(m^{1/2d}) \cdot F = \omega(1) \cdot F$.

Lemma 6.5 (MST)**.** *Let $d = o(\log n)$ but $d = \omega(\log \log n)$. There exists X with constant doubling dimension, such that with at least $2/3$ probability, there exists a $(1 + o(1))$-approximate MST $\mathcal{M}'$ for GX whose total cost whose total cost when pulled back to X is at least $\omega(1) \cdot M$.*

Finally, we show that the guarantees of facility location are in fact not maintained for k-means and k-medians clustering. In other words, the bound of $O(\log k)$ by (Makarychev et al., 2019) is optimal even for sets of doubling dimension $O(1)$.

Theorem 6.6 (k-means/k-medians)**.** *Let $k < n$ and $d = o(\log k)$. Then, there exists X with constant doubling dimension, such that with probability at least $2/3$, the k-means (resp., medians) cost of GX is $o(1)$ times the k-means (resp., medians) cost of X. Moreover, the optimal choice of k centers in GX, when pulled back to X, will be an $\omega(1)$-approximate solution in the original dimension $\mathbb{R}^m$.*

At a first glance, Theorem 6.6 may appear to contradict our upper bounds for facility location. However, in our counterexamples for k-means and k-medians, the cost (both initially and after projection) is substantially smaller than k. Facility location adds a cost of k for the k facilities that are created, and since these facilities now make up the bulk of the cost, the facility location cost is still approximately preserved under random projection.

7. Experiments

We use the following datasets in our experiments for Subsections 7.1 and 7.2.

- **Faces Dataset**: This dataset is used in the influential ISOMAP paper and consists of 698 images of faces in dimension 4096 (Tenenbaum et al., 2000). From (Kégl, 2002), we can estimate that the doubling dimension of this dataset is a small constant.
- **MNIST '2' Dataset**: 1000 randomly chosen images from the MNIST dataset (dimension 784) restricted to the digit 2. We picked 2 since it is considered in the original ISOMAP paper (Tenenbaum et al., 2000).

All of our experimental results are averaged over 20 independent trials and the projection dimension d ranges from 5 to 20 inclusive.All of our experiments were done on a CPU with i5 2.7 GHz dual core and 8 GB RAM.

7.1. Facility Location: Cost versus Accuracy Analysis

In this section we compare the accuracy of the MP algorithm with/without dimensionality reduction for various number of centers opened.

Experimental Setup We project our datasets and compute a facility location clustering with the opening costs scaled so that $n/2, n/5$, and $n/10$ facilities are opened respectively. We then take this solution and evaluate its cost in the original dimension. We also perform a clustering in the original dimension with the same prescribed number of facilities opened and plot the ratio of the cost of the solution found in the lower dimension (but evaluated in the larger dimension) to the solution found in the larger dimension. We also plot the time taken for the clustering algorithm in the projected dimension. We use the MP algorithm to perform our clustering due to the intractability of finding the exact optimum and also because the MP algorithm is fast and quite practical to use.

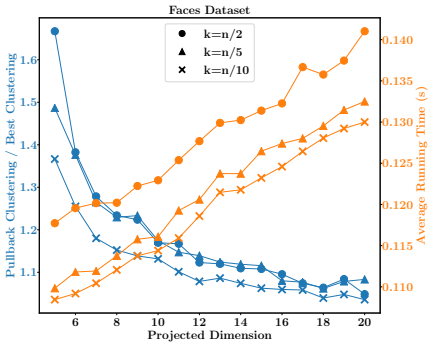
Results Our results are plotted in Figures 3a-3b. Our experiments empirically demonstrate that the dimensionality reduction step does not significantly decrease the accuracy of the solution. Furthermore, we get a substantial reduction in the runtime since the average runtime was at least 20 seconds for Faces and around 6.5 seconds for MNIST '2' in the original dimension for all the values of k tested, which is **1-2** orders of magnitude higher than the runtime when random projections are used. Note that the runtime includes the time taken to perform the random projection. Overall, our experiments demonstrate that the method of performing dimensionality reduction to perform facility location clustering is well-founded.

7.2. MST: Cost versus Accuracy Analysis

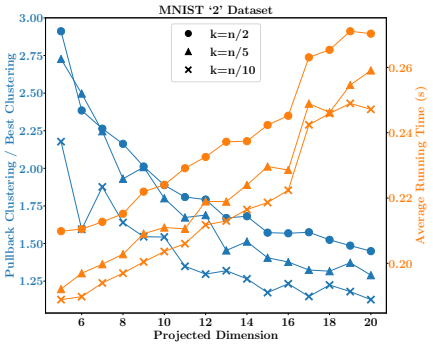
We empirically show the benefits of using dimensionality reduction for minimum spanning tree computation.

Experimental Setup We project our datasets and compute a MST. We then take the tree found in the lower dimension and compare its cost in the higher dimension against the actual MST. Our MST algorithm is a variant of the Boruvka algorithm from (March et al., 2010) that is suitable for point sets in large dimensions and is implemented in the popular 'mlpack' machine learning library (Curtin et al., 2018).

Results Our results are plotted in Figures 4a-4b. In the blue plots of these figures, the ratio of the cost of the tree found in the projected dimension, but evaluated in the original dimension, to the cost of the actual MST is shown. We see that indeed as projection dimension increases, the ratio approaches 1. However even for very low values of d, such as 10, the tree found in the projected space serves as a good approximate for the actual MST. Conversely, we see that as d increases, the cost of computing the MST also increases as shown in the orange plots of the Figures 4a and 4b. Note that the time taken to perform the projection is also included. The time taken to compute the MST in the

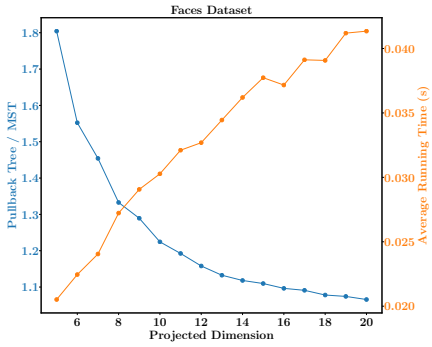


(a) Faces Dataset

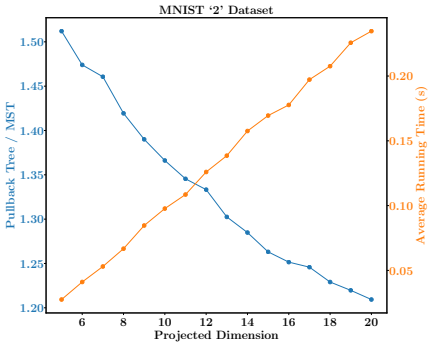


(b) MNIST '2' Dataset

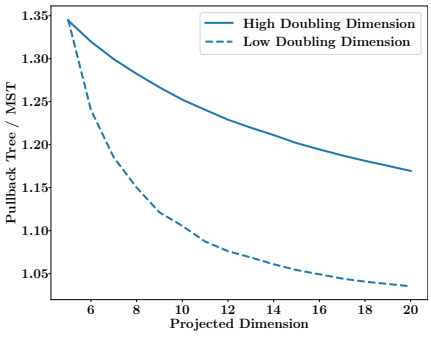
Figure 3. Facility Location Experiments. (a) **Blue:** Ratio of solution costs with/without dimensionality reduction, as a function of d. **Orange:** Running time (in secs) as a function of d. (b) Same plot as (a) but for MNIST '2' dataset.



(a) Faces Dataset



(b) MNIST '2' Dataset



(c) Low vs High Doubling Dimension Comparison

Figure 4. Minimum Spanning Tree Experiments. (a) **Blue:** Ratio of solution costs with/without dimensionality reduction, as a function of d. **Orange:** Running time (in secs) as a function of d. (b) Same plots as (a) but for MNIST '2' dataset. (c) Dataset 1 (low doubling dimension) can be projected into a much smaller dimension than Dataset 2 for MST computation.

original dimension was approximately 3.2 seconds for the Faces dataset and 7.1 seconds for the MNIST '2' dataset. Therefore, projection to dimension $d = 20$ gives us approximately **80x** improvement in speed for the Faces dataset and **30x** improvement in speed for the MNIST '2' dataset while having a low cost distortion.

7.3. Large versus Small Doubling Dimension

In this section we present two datasets in $\mathbb{R}^n$ where one dataset has doubling dimension $O(1)$ and the other has doubling dimension at least $\Omega(\log n)$ which is asymptotically the largest doubling dimension of any set of size n. We empirically show that the second dataset requires larger projection dimension than the first to guarantee that the MST found in the projected space induces a good solution in the original space. Our two datasets are the following. Let e_i denote the standard basis vectors in $\mathbb{R}^n$. We first draw n standard Gaussians $g_1, \cdots, g_n \in \mathbb{R}$. Our datasets are:

Dataset 1: $\{g_1 \cdot e_1, g_1 \cdot e_1 + g_2 \cdot e_2, \ldots, g_1 \cdot e_1 + \cdots + g_n \cdot e_n\}$.
Dataset 2: $\{g_1 \cdot e_1, g_1 \cdot e_2, \ldots, g_n \cdot e_n\}$.

Note that we use the same g_i's for both datasets. The above datasets appear to be similar, but it can be shown that their respective doubling dimensions are $O(1)$ and $\Omega(\log n)$.

Experimental Setup We let $n = 1000$ and construct the two datasets. We project our datasets and find the MST for each dataset in the projected space. Then we evaluate the cost of this tree in the larger dimension and compare this cost to the cost of the actual MST for each dataset.

Results Figure 4c demonstrates that we can find a high quality approximation of the MST by finding the MST in a much smaller dimension for Dataset 1 compared to Dataset 2. For example, Dataset 1 required only $d = 10$ dimensions to approximate the true MST within 10% relative error while Dataset 2 needed $d = 38$ to get within 10% relative error of the true MST.

Acknowledgments

This research was supported in part by the NSF TRIPODS program (awards CCF-1740751 and DMS-2022448); NSF award CCF-2006798; MIT-IBM Watson collaboration; Simons Investigator Award; and NSF Graduate Research Fellowship Program.

References

Abboud, A., Cohen-Addad, V., and Houdrouge, H. Subquadratic high-dimensional hierarchical clustering. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 11576–11586, 2019.

Badoiu, M., Czumaj, A., Indyk, P., and Sohler, C. Facility location in sublinear time. In Caires, L., Italiano, G. F., Monteiro, L., Palamidessi, C., and Yung, M. (eds.), *Automata, Languages and Programming*, pp. 866–877, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31691-6.

Becchetti, L., Bury, M., Cohen-Addad, V., Grandoni, F., and Schwiegelshohn, C. Oblivious dimension reduction for k-means: beyond subspaces and the johnson-lindenstrauss lemma. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1039–1050, 2019.

Boutsidis, C., Zouzias, A., and Drineas, P. Random projections for k-means clustering. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1*, NIPS'10, pp. 298–306, Red Hook, NY, USA, 2010. Curran Associates Inc.

Chan, T.-H. H., Hu, S., and Jiang, S. H.-C. A ptas for the steiner forest problem in doubling metrics. *SIAM Journal on Computing*, 47(4):1705–1734, 2018. doi: 10.1137/16M1107206. URL https://doi.org/10.1137/16M1107206.

Clarkson, K. *Nearest-Neighbor Searching and Metric Space Dimensions*, pp. 15–59. 04 2012.

Cohen, M. B., Elder, S., Musco, C., Musco, C., and Persu, M. Dimensionality reduction for k-means clustering and low rank approximation. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pp. 163–172, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450335362. doi: 10.1145/2746539.2746569. URL https://doi.org/10.1145/2746539.2746569.

Curtin, R. R., Edel, M., Lozhnikov, M., Mentekidis, Y., Ghaisas, S., and Zhang, S. mlpack 3: a fast, flexible machine learning library. *Journal of Open Source Software*, 3:726, 2018. doi: 10.21105/joss.00726. URL https://doi.org/10.21105/joss.00726.

Dasgupta, S. A cost function for similarity-based hierarchical clustering. In Wichs, D. and Mansour, Y. (eds.), *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pp. 118–127. ACM, 2016. doi: 10.1145/2897518.2897527.

Friggstad, Z., Rezapour, M., and Salavatipour, M. R. Local search yields a ptas for k-means in doubling metrics. *SIAM Journal on Computing*, 48(2):452–480, 2019. doi: 10.1137/17M1127181. URL https://doi.org/10.1137/17M1127181.

Gottlieb, L. A light metric spanner. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pp. 759–772, Oct 2015. doi: 10.1109/FOCS.2015.52.

Gottlieb, L.-A. and Krauthgamer, R. Proximity algorithms for nearly doubling spaces. *SIAM Journal on Discrete Mathematics*, 27(4):1759–1769, 2013.

Har-Peled, S. and Kumar, N. Approximate nearest neighbor search for low-dimensional queries. *SIAM Journal on Computing*, 42(1):138–159, 2013. doi: 10.1137/110852711. URL https://doi.org/10.1137/110852711.

Indyk, P. and Naor, A. Nearest-neighbor-preserving embeddings. *ACM Trans. Algorithms*, 3(3):31–es, August 2007. ISSN 1549-6325. doi: 10.1145/1273340.1273347. URL https://doi.org/10.1145/1273340.1273347.

Johnson, W. and Lindenstrauss, J. Extensions of lipschitz maps into a hilbert space. *Contemporary Mathematics*, 26:189–206, 01 1984. doi: 10.1090/conm/026/737400.

Kégl, B. Intrinsic dimension estimation using packing numbers. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, NIPS'02, pp. 697–704, Cambridge, MA, USA, 2002. MIT Press.

Larsen, K. G. and Nelson, J. Optimality of the johnson-lindenstrauss lemma. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 633–638. IEEE, 2017.

Makarychev, K., Makarychev, Y., and Razenshteyn, I. Performance of johnson-lindenstrauss transform for k-means and k-medians clustering. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pp. 1027–1038, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367059. doi: 10.1145/

3313276.3316350. URL https://doi.org/10.1145/3313276.3316350.

Manning, C., Raghavan, P., and Schütze, H. *An Introduction to Information Retrieval Christopher D*. Cambridge University Press, 2009.

March, W. B., Ram, P., and Gray, A. G. Fast euclidean minimum spanning tree: Algorithm, analysis, and applications. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pp. 603–612, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781450300551. doi: 10.1145/1835804.1835882. URL https://doi.org/10.1145/1835804.1835882.

Mettu, R. R. and Plaxton, C. G. The online median problem. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pp. 339–348, Nov 2000. doi: 10.1109/SFCS.2000.892122.

Moseley, B. and Wang, J. Approximation bounds for hierarchical clustering: Average linkage, bisecting k-means, and local search. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, 2017.

Naor, A. Metric dimension reduction: A snapshot of the ribe program. *ArXiv*, abs/1809.02376, 2018.

Saulpic, D., Cohen-Addad, V., and Feldmann, A. Near-linear time approximations schemes for clustering in doubling metrics. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 540–559, Nov 2019. doi: 10.1109/FOCS.2019.00041.

Talwar, K. Bypassing the embedding: Algorithms for low dimensional metrics. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pp. 281–290, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138520. doi: 10.1145/1007352.1007399. URL https://doi.org/10.1145/1007352.1007399.

Tenenbaum, J. B., Silva, V. d., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. ISSN 0036-8075. doi: 10.1126/science.290.5500.2319. URL https://science.sciencemag.org/content/290/5500/2319.