

Composition of nested embeddings with an application to outlier removal*

Shuchi Chawla[†]
shuchi@cs.utexas.edu

Kristin Sheridan[†]
kristin@cs.utexas.edu

Abstract

We study the design of embeddings into Euclidean space with outliers. Given a metric space (X, d) and an integer k , the goal is to embed all but k points in X (called the “outliers”) into ℓ_2 with the smallest possible distortion c . Finding the optimal distortion c for a given outlier set size k , or alternately the smallest k for a given target distortion c are both NP-hard problems. In fact, it is UGC-hard to approximate k to within a factor smaller than 2 even when the metric sans outliers is isometrically embeddable into ℓ_2 . We consider bi-criteria approximations. Our main result is a polynomial time algorithm that approximates the outlier set size to within an $O(\log^2 k)$ factor and the distortion to within a constant factor.

The main technical component in our result is an approach for constructing Lipschitz extensions of embeddings into Banach spaces (such as ℓ_p spaces). We consider a stronger version of Lipschitz extension that we call a *nested composition of embeddings*: given a low distortion embedding of a subset S of the metric space X , our goal is to extend this embedding to all of X such that the distortion over S is preserved, whereas the distortion over the remaining pairs of points in X is bounded by a function of the size of $X \setminus S$. Prior work on Lipschitz extension considers settings where the size of X is potentially much larger than that of S and the expansion bounds depend on $|S|$. In our setting, the set S is nearly all of X and the remaining set $X \setminus S$, a.k.a. the outliers, is small. We achieve an expansion bound that is polylogarithmic in $|X \setminus S|$.

1 Introduction

Low distortion metric embeddings are an important algorithmic tool with a myriad of applications. The goal is to transform a dataset that lies in an unwieldy metric space into one lying in a nicer space, enabling clean and fast algorithms. Embeddings play an important role in the design of approximation algorithms, in finding good low dimensional representations for data in machine learning and data science, in data visualization, in the design of fast algorithms, and more. (See, e.g., [21].)

One of the most enduring concepts from the vast literature on embeddings is that of distortion, which is defined to be the maximum ratio over all pairs of points in the metric by which the distance between the points is expanded or contracted by the embedding. Since this is a worst case notion, it can be particularly sensitive to errors or noise or even intentional corruption of the underlying data. Indeed adversarial or random data corruption is a frequent problem in data science contexts.

Motivated by these applications, Sidiropoulos *et al.* [32] introduced the notion of embeddings with outliers. Given a metric space (X, d_X) of finite size and a target space (Y, d_Y) our goal is to find a low-distortion embedding into Y of all but a few points in X ; these points are called the outliers. Formally, we say that the space (X, d_X) has a (k, c) -outlier embedding into space (Y, d_Y) if there exists an outlier set $K \subset X$ of size at most k and a map α from $X \setminus K$ to Y with distortion at most c . [32] showed that for many host spaces (Y, d_Y) of interest, computing the optimal outlier set size k for a given target distortion c , or the optimal distortion c for a given target outlier set size k , is NP-hard.

In this work, we study the design of approximately optimal outlier embeddings into the Euclidean metric. Given a metric (X, d_X) that admits a (k, c) -outlier embedding into ℓ_2 , we provide a polynomial time algorithm that constructs an $(O(k \text{ polylog } k), O(c))$ -outlier embedding into ℓ_2 . In other words, our algorithm removes

*The full version of the paper can be accessed at <https://arxiv.org/abs/2306.11604>

[†]Department of Computer Science, University of Texas at Austin.

$O(k \text{ polylog } k)$ outliers and returns an embedding of the remaining metric into ℓ_2 that has distortion only a small constant factor worse than the desired one. In fact, our algorithm allows for a tradeoff between the outlier set size and the distortion obtained – allowing us to obtain a distortion of $(1 + \epsilon)c$ for any $\epsilon > 0$ at the cost of blowing up the outlier set size by an additional multiplicative factor of at most $1/\epsilon$.

To our knowledge, the only approximations to multiplicative distortion for outlier embeddings known prior to our work considered the special case of embedding unweighted graphs into a line. For this setting, Chubarian and Sidiropoulos [11] developed an algorithm that constructs an $(O(c^6 k \log^{5/2}(n)), O(c^{13}))$ -outlier embedding when the input metric is an unweighted graph metric. Here n is the size of the given metric X . While our result is incomparable to theirs (as it does not limit the dimension of the embedding), we emphasize that our approximation factors do not depend on the size of the metric X and has a vastly improved dependence on the distortion c , as we discuss below.

Lipschitz extension and composition of nested embeddings. The main technical tool in our work is a stronger version of Lipschitz extensions that we call a composition of nested embeddings. Formally, consider a metric space (X, d_X) and a subspace $S \subset X$. Let $\alpha_S : S \rightarrow Y$ be an embedding from S into Y with distortion c_S . A Lipschitz extension is an embedding α of the entire set X into Y such that $\alpha(s) = \alpha_S(s)$ for all $s \in S$ and the expansion on any pair of points in X is not too much more than c_S . Observe that we only require the *expansion* on points in X to be bounded – distances are allowed to shrink arbitrarily. Results for Lipschitz extensions typically achieve expansion bounds that depend on the size of S and are independent of the size of X . In particular, X can be arbitrarily larger than S , even unbounded.

In our setting, we are interested in the case where S (i.e. the “good” set) is almost all of X and $X \setminus S$ is much smaller. We desire an extension where the expansion bounds depend on the size of $|X \setminus S|$. We call such a Lipschitz extension a *weak* composition of nested embeddings; the terminology is explained below. In Section 4 we design weak nested compositions for embeddings into ℓ_p spaces (or more generally Banach spaces) where expansion on any pair of points is at most $O(H_k)$ times c_S .

Weak nested compositions into the Euclidean metric allow us to identify outliers with the help of a fractional relaxation. We express the problem of finding an outlier embedding into the Euclidean metric as a semi-definite program where all but k fractionally chosen points are required to have small distortion between them. (See Section 3.) The distortion allowed for any pair of points depends on the fractional extent to which either is chosen as an outlier. The existence of a weak $O(H_k)$ -nested composition implies that we can find a feasible fractional solution where the outliers suffer only a small factor larger distortion than the non-outliers. This enables a rounding scheme that approximates the number of outliers to within an $O(H_k^2)$ factor and preserves the distortion over non-outliers to within a constant factor.

Strong nested composition. Although our application to outlier embeddings requires only bounding the *expansion* of the Lipschitz extension from S to $X \setminus S$, more generally we ask whether we can construct an extension which has small *distortion* over all of X , that is, it doesn’t expand or contract by much. We call such an extension a *strong nested composition*, and we believe this concept is of independent interest. Observe that the distortion of such an embedding cannot simply be a function of S alone, as $X \setminus S$ may not admit any good embedding at all.

Formally, we are given two *nested* embeddings $\alpha_X : X \rightarrow Y$ and $\alpha_S : S \rightarrow Y$ with distortions c_X and c_S respectively. In general we would expect that the distortion of α_S is smaller than that of α_X . In fact, the distortion of α_X even restricted to just the subset S may be larger than c_S . Our goal is to find a new embedding from X into Y such that the distortion of this embedding over the good set S is the same as before – c_S – while the distortion over all of X is comparable to c_X . In particular, we want a combination of the two embeddings that inherits their respective distortions, with small multiplicative worsening, over the corresponding sets of points. In Section 5 we show that strong nested compositions exist for the ℓ_1 metric, and leave open the question of extending this result to other ℓ_p metrics.

The concept of nested composition is similar in some ways to unions of embeddings studied previously in [25] and [29]. The goal in these works is to combine two embeddings over *disjoint* subsets A and B of a metric space into a single embedding over their union such that the distortion over the union is comparable to the distortions c_A and

c_B of the given embeddings. When the host metric is ℓ_2 or ℓ_∞ , these works construct unions whose distortion is some constant factor times the *product*, $c_A \cdot c_B$, of the distortions of the given embeddings. Additionally, both the sets inherit the same distortion bound – if, for example, $c_A \ll c_B$, it is not guaranteed that points in A will retain similarly lower distortion in the composition. Extending these results to ℓ_p metrics for $p \notin \{2, \infty\}$ is open. By contrast, our composition guarantees *the same* distortion c_S as the given embedding on the “inner” set of points S , and a bound on the expansion over the remaining points that is linear in c_X (although the latter does not imply a bound the distortion over X as the composed embedding may contract some pairs of points).

Our results are also reminiscent of local versus global guarantees for metric embeddings. In particular, Arora et al. [3] ask: suppose that *every* subset of metric space X of size at most k admits a low distortion embedding into ℓ_1 , does X also admit a low distortion embedding into ℓ_1 ? Charikar et al. [9] show that this is indeed possible but that the distortion blows up by a factor of $\Theta(\log n/k)$. Our setting is slightly different in that we not only need every small set of size k to be embeddable with low distortion, but we also need a good embedding for *one* set of size $n - k + 1$.

A tradeoff between outlier set size and distortion. Armed with a weak nested composition into ℓ_2 space, we develop an SDP-rounding algorithm for obtaining bicriteria approximations for outlier embeddings into ℓ_2 . For a metric (X, d_X) that admits a (k, c) -outlier embedding into ℓ_2 , and any given target distortion $c' = \gamma c$, $\gamma \geq 1$, our algorithm constructs a (k', c') -outlier embedding with at most $k' = O(\frac{\log^2 k}{\gamma^2 - 1} \cdot k)$ outliers. Additionally, we achieve a tradeoff between the outlier set size k' and the target distortion c' . Setting $c' = (1 + \epsilon)c$ for a small $\epsilon > 0$, for example, increases the outlier set size by an $O(1/\epsilon)$ factor.

Hardness of approximation. Finally, we note that Sidiropoulos *et al.* [32] showed that it is NP-hard to determine the size of the smallest outlier set such that the remaining metric is isometrically embeddable into ℓ_2^d for any fixed dimension $d > 1$. Designing an outlier embedding into ℓ_2 with arbitrary dimension is potentially an easier problem. Since we do not limit the dimension of the outlier embeddings we construct, we revisit and strengthen [32]’s result along these lines. We show that NP-hardness continues to hold even for embeddings into ℓ_2 without a specified dimension bound, and also when the given metric is the shortest path metric of an unweighted undirected graph. Our construction of a hard instance is arguably simpler than that of Sidiropoulos *et al.*, and is readily seen to extend to ℓ_p metrics for $p > 1$. We present a separate, more involved, construction for $p = 1$. As with Sidiropoulos *et al.*’s results we show that, under the unique games conjecture, it is also hard to obtain a $2 - \epsilon$ approximation for the minimum outlier set size, for any $\epsilon > 0$.

Further related work

Approximations for distortion. Much of the work on low distortion embeddings focuses on providing uniform bounds for embedding any given finite metric into a structured space. Indyk and Matousek [21] give an excellent overview of many of these results. Bourgain’s Theorem [7, 24] shows that all finite metric spaces of size n have an $O(\log n)$ -distortion embedding into ℓ_p -space for $p \geq 1$ and that a randomized version of such an embedding can be computed quickly. A derandomized such embedding can be computed in polynomial time with $\Theta(n^2)$ dimensions.

More closely related to our work, a number of papers study the objective of approximating the instance-specific minimum distortion for embedding into various host metrics. This includes, e.g., embeddings into constant dimensional Euclidean space [5, 13, 17, 26, 31], the line [26, 19, 4, 28], trees [10, 6], and ultrametrics [2]. These works tend to use combinatorial arguments as they focus on low dimensional embeddings, whereas ours relies on an SDP formulation of the problem.

Lipschitz extension. Much of the prior work on Lipschitz extensions has largely focused on extending embeddings of small subsets of a metric (X, d) into a target space. The Johnson-Lindenstrauss extension theorem shows that for any metric (X, d) and Lipschitz embedding $\alpha_S : S \subseteq X \rightarrow \ell_2$ with $|S| = n$, there exists an embedding $\alpha : X \rightarrow \ell_2$ with Lipschitz constant at most $O(\sqrt{\log n})$ times the Lipschitz constant of the original embedding [22]. (Note that contraction may be arbitrary in this embedding, and X does not have to be a finite space.) More recent work by Naor and Rabani [27] shows that in general, some Lipschitz extensions into Banach spaces require $\Omega(\sqrt{\log n})$ increase in the Lipschitz factor. Other work on Lipschitz extensions has largely focused

on extending embeddings of subsets of X in which all point-wise distances in S are at least an ϵ fraction of the subspace's diameter (i.e. the distances between points in S aren't too different from each other).

In this paper, we focus on extending embeddings of subsets $S \subseteq X$ of relatively large size compared to X into ℓ_p spaces. In particular, we show that if $|X| = n$ and $|X \setminus S| = k$, then there exists a Lipschitz extension of any embedding of S into ℓ_p that increases the Lipschitz constant by at most a factor of $O(\log k)$. Because we are considering outlier sets $X \setminus S$ that we expect to be small, this is generally going to give us a better approximation factor for our purposes.

Outlier embeddings. The notion of outlier embeddings was first introduced by Sidiropoulos *et al.* in [32]. In that paper, they showed that it is NP-hard to find the size of a minimum outlier set for embedding a metric into ultrametrics, tree metrics, or ℓ_2^d for constant d . Under the Unique Games Conjecture, it is also NP-hard to approximate these values to a factor better than 2. On the algorithmic side, they gave polynomial time algorithms to 3, 4, or 2-approximate minimum outlier set size for isometric embeddings into ultrametrics, tree metrics, or ℓ_2^d for fixed constant d , respectively. The algorithm for ℓ_2^d embeddings is exponential in d , so d cannot grow with the size of the input while remaining efficient.

Sidiropoulos *et al.* also gave bi-criteria approximations for ℓ_∞ (i.e. additive) distortion. In particular, they give a polynomial time algorithm to find embeddings with at most $2k$ outliers and $O(\sqrt{\delta})$ ℓ_∞ -distortion when there exists an embedding of the metric with at most k outliers that has ℓ_∞ -distortion at most δ . The algorithm is polynomial in k , δ , and n but exponential in d which is taken to be a constant.

Chubarian *et al.* [11] expanded on the results of Sidiropoulos *et al.* by giving the first bicriteria approximation for minimum outlier sets with multiplicative distortion. In particular, they showed that given an unweighted graph metric and tuple (k, c) , there is a polynomial time algorithm that either correctly decides that there does not exist an embedding of the metric into the real line with at most k outliers and at most c distortion, or outputs an $(O(c^6 k \log^{5/2} n), O(c^{13}))$ -outlier embedding into the real line.

Embeddings with slack. A different notion of distortion that is robust to noise in the data was introduced by Abraham *et al.* [4]. In this *embeddings with slack* model, a budget of slack is applied to pairs of vertices in the metric space (as opposed to individual vertices, as in our model). An embedding of a metric space (X, d_X) into another space (Y, d_Y) has distortion c with ϵ -slack if all but an ϵ fraction of the distances are distorted by at most c . Abraham *et al.* [4] showed that there exists a polynomial time algorithm that finds an $O(\log \frac{1}{\epsilon})^{1/p}$ -distortion embedding with ϵ slack for embeddings into ℓ_p for $p \geq 1, \epsilon > 0$. Chan *et al.* [8] showed that there is a polynomial time algorithm for embedding a metric (V, d) of n points into a spanner graph of at most $O(n)$ edges with ϵ -slack and $O(\log \frac{1}{\epsilon})$ distortion. Lammersen *et al.* [23] extended results in this topic to the streaming setting by giving an algorithm using poly-logarithmic space that computes embeddings with slack into finite metrics. From an algorithmic viewpoint, defining outliers in terms of edges versus nodes leads to very different optimization problems. Furthermore, an important difference between our work and these previous works on embeddings with slack is that we are interested in instance-specific approximations, whereas these latter works aim to find uniform bounds on distortion with slack that hold for all input metric spaces.

2 Definitions and main results

We begin by defining terms used in this paper and discussing our main results.

Outlier embeddings and distortion

DEFINITION 2.1. A metric space is a pair (X, d_X) such that X is a set of elements we call points or nodes and $d_X : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is a function that has the following properties:

1. For all $x, y \in X$, $d_X(x, y) = 0$ if and only if $x = y$
2. For all $x, y \in X$, $d_X(x, y) = d_X(y, x)$
3. For all $x, y, z \in X$, $d_X(x, z) \leq d_X(x, y) + d_X(y, z)$

In this paper we will focus on expanding embeddings from a given finite metric into ℓ_p .

DEFINITION 2.2. An expanding embedding $\alpha : X \rightarrow Y$ of a metric space (X, d_X) into another metric space (Y, d_Y) has distortion $c \geq 1$ if for all $u, v \in X$:

$$d_X(u, v) \leq d_Y(\alpha(u), \alpha(v)) \leq c \cdot d_X(u, v)$$

Following [11] and [32], we consider so-called *outlier embeddings* that embed all but a small set of outliers from the given metric into the host space.

DEFINITION 2.3. An embedding $\alpha : X \rightarrow Y$ of a metric space (X, d_X) into another metric space (Y, d_Y) is a (k, c) -outlier embedding if there exists $K \subseteq X$ such that $|K| \leq k$ and $\alpha|_{X \setminus K}$ (the restriction of α to the domain $X \setminus K$) is an embedding of $(X \setminus K, d|_{X \setminus K})$ with distortion at most c .

Nested compositions and Lipschitz extensions A main component of our approach is showing that a low-distortion embedding of a subset of the given metric space into some ℓ_p space can be extended into an embedding of the entire metric with small expansion.

DEFINITION 2.4. Let (X, d_X) and (Y, d_Y) be two metric spaces and $\alpha_S : S \subseteq X \rightarrow Y$ be an embedding with Lipschitz constant at most L . Then $\alpha : X \rightarrow Y$ is a Lipschitz extension of α_S with extension factor $g(|S|, |X|)$ if for all $x \in S$, $\alpha(x) = \alpha_S(x)$ and for all $x, y \in X$,

$$(2.1) \quad d_Y(\alpha(x), \alpha(y)) \leq g(|S|, |X|) \cdot L \cdot d_X(x, y).$$

We introduce a new variant of Lipschitz extension called nested composition, which focuses on parameters of interest for us, and aims to preserve both the expansion and contraction of the embedding. In particular, we start with *expanding* nested embeddings of S and X into Y with distortions c_S and c_X respectively. Our goal is to produce a single expanding embedding that preserves the smaller distortion c_S over pairs of points in S , and bounds the distortion over X by c_X times some function g of the size of $X \setminus S$. Importantly, the factor g depends only on the size of $X \setminus S$, and not on the size of S , that may be much larger. The weak variation of this notion is similar to Lipschitz extension in that it will still allow arbitrary contraction over X , but it will no longer require that the exact points of the original embedding be preserved.

DEFINITION 2.5. (COMPOSITION OF NESTED EMBEDDINGS) Let (X, d_X) and (Y, d_Y) be two metric spaces and $g : [0, \infty)^2 \times \mathbb{N} \rightarrow [1, \infty)$. A weak g -nested composition is an algorithm that, given a set $S \subseteq X$ with $k := |X \setminus S|$, and two expanding embeddings, $\alpha_S : S \rightarrow Y$ with distortion c_S and $\alpha_X : X \rightarrow Y$ with distortion $c_X \geq c_S$, returns an embedding $\alpha : X \rightarrow Y$ such that,

$$(2.2) \quad \text{for all } u, v \in S, \quad d_X(u, v) \leq d_Y(\alpha(u), \alpha(v)) \leq c_S \cdot d_X(u, v),$$

$$(2.3) \quad \text{and, for all } u, v \in X, \quad d_Y(\alpha(u), \alpha(v)) \leq g(c_S, c_X, k) \cdot d_X(u, v).$$

We say that it is a nested composition if the embedding α is additionally an expanding embedding. That is,

$$(2.4) \quad \text{for all } u, v \in S, \quad d_X(u, v) \leq d_Y(\alpha(u), \alpha(v)) \leq c_S \cdot d_X(u, v),$$

$$(2.5) \quad \text{and, for all } u, v \in X, \quad d_X(u, v) \leq d_Y(\alpha(u), \alpha(v)) \leq g(c_S, c_X, k) \cdot d_X(u, v).$$

We use randomness in our construction of nested compositions. For a randomized Lipschitz extension, we require that $\alpha(x) = \alpha_S(x)$ for all $x \in S$ for any possible α in the distribution, and that expansion is bounded in expectation over the randomness in the construction. For a randomized nested composition, all contraction bounds should be satisfied with probability 1 and expansion bounds in expectation.

In our work, we will show that there is an efficient algorithm that finds a randomized Lipschitz extension (or weak nested composition) with extension factor $O(\log |X \setminus S|)$ where (Y, d_Y) is *any* target metric space. However, the application to outlier embeddings requires the existence of a single deterministic extension/composition, so our deterministic extension requires that the target metric be a Banach space (such as an ℓ_p space).

Main results

Nested compositions. Our main technical result shows that we can efficiently construct randomized weak $O(\log k)$ -nested compositions when the host space Y is an ℓ_p metric. For $p = 2$, the case of interest for us, we can even efficiently construct a *deterministic* weak $O(\log k)$ -nested composition into ℓ_2 space by solving an appropriate semi-definite program. Finally, for $p = 1$, our construction ensures expansion and satisfies both inequalities in constraint (2.5). We leave open the question of constructing (strong) nested compositions satisfying expansion for $p > 1$.

THEOREM 2.6. *Let (X, d_X) be any finite metric and (Y, d_Y) be any Banach space. Let $\alpha_S : X \rightarrow Y$ be any Lipschitz embedding of $S \subseteq X$ into Y with $k := |X \setminus S|$. Then there exists a $125H_k$ -Lipschitz extension (and thus a weak $125H_k c_S$ -nested composition) from X into Y , where H_k is the k th Harmonic number.*

THEOREM 2.7. *Let (X, d_X) be any finite metric. Then there exists a $382H_k c_X$ -nested composition from X into ℓ_1 .*

For metric spaces (X, d_X) where the distortion of the embedding into ℓ_1 depends on the size of the subset being embedded, we can in fact obtain a stronger guarantee – the distortion of the composition depends only on the size of the outlier set $k = |X \setminus S|$ and not on the size of the entire space X . In particular, we can replace the quantity c_X in constraint (2.5) by the worst case distortion from embedding any subset of size $k + 1$ into the host metric.

THEOREM 2.8. *Let (X, d_X) be any finite metric. Suppose that for $k \in \mathbb{Z}^+$ every subset of X of size $k + 1$ can be embedded into ℓ_1 with distortion ζ_k . Then, there exists a $382H_k \zeta_k$ -nested composition from X into ℓ_1 .*

Outlier embeddings. With these results in hand, we obtain the following bicriteria approximation for outlier embeddings from finite metrics into ℓ_2 .

THEOREM 2.9. *Let (X, d_X) be a metric space that admits a (k, c) -outlier embedding. Then there exists a polynomial time algorithm \mathcal{A} that, for any $\gamma > 1$, finds a subset $K \subseteq X$ and an embedding $\alpha : X \setminus K \rightarrow \ell_2$ such that α has distortion at most γc , and*

$$|K| \leq 2 \frac{(125 \cdot H_k)^2 + \gamma^2}{\gamma^2 - 1} k$$

Choosing $\gamma = 1 + \epsilon$ for $\epsilon \in (0, 1]$, in particular, provides an $\left(O\left(\frac{\log^2 k}{\epsilon} k\right), (1 + \epsilon)c\right)$ -outlier embedding from X into ℓ_2 .

Hardness of approximation. Finally, we provide a strengthening of Sidiropoulos *et al.* [32]’s hardness result for outlier embeddings, showing that it is NP-hard to determine the size of the smallest outlier set such that the remaining metric is isometrically embeddable into ℓ_2 even when the dimension of the embedding is unrestricted. As with Sidiropoulos *et al.*’s results, under the unique games conjecture, it is also hard to obtain a $2 - \epsilon$ approximation for the minimum outlier set size, for any $\epsilon > 0$. Furthermore, our construction achieves two other properties that [32]’s doesn’t: (1) Our hardness results apply also to shortest path metrics over unweighted undirected graphs. (2) We show that the hardness result holds for embedding into the ℓ_p metric for any $p \geq 1$.

THEOREM 2.10. *Let (X, d) be the distance metric for an unweighted undirected graph $G = (V, E)$. then, given (X, d, k) it is NP-hard to decide if there exists a subset $K \subseteq X$ with $|K| = k$ such that $(X \setminus K, d|_{X \setminus K})$ is isometrically embeddable into ℓ_p for any finite integer $p \geq 1$.*

Under the unique games conjecture, it is NP-hard to find a $2 - \epsilon$ approximation for the minimum such k , for any $\epsilon > 0$.

3 SDP relaxation and approximation

In this section we will prove Theorem 2.9. We begin with a semi-definite programming formulation for constructing an outlier embedding into ℓ_2 . In the absence of outliers, the optimal embedding of any finite metric into ℓ_2 can be found using an SDP. In particular, for a given such metric (X, d) , let \vec{v}_x for $x \in X$ denote the mapping of x

into ℓ_2 . Then, the constraint $d^2(x, y) \leq \|\vec{v}_x - \vec{v}_y\|^2 \leq c^2 \cdot d^2(x, y)$ ensures that the distance between points x and y is distorted by a factor of at most c . The challenge is to incorporate outliers into this formulation.

Consider the finite metric space (X, d) , and suppose that there exists a (k, c) -outlier embedding from (X, d) into Euclidean space for some integer $k > 0$ and real number $c \geq 1$. We use the vector \vec{v}_x for $x \in X$ to denote the mapping of x into ℓ_2 , and $\delta_x \in [0, 1]$ as an indicator for whether x is an outlier. We then construct the following SDP:

$$\begin{aligned}
 \text{(Outlier SDP)} \quad & \min_{\delta, \vec{v}} \sum_{x \in X} \delta_x \\
 (3.6) \quad & \text{s.t. } \forall x, y \in X : \quad (1 - \delta_x - \delta_y) \cdot d^2(x, y) \leq \|\vec{v}_x - \vec{v}_y\|^2 \leq (c^2 + (\delta_x + \delta_y)f(k)) \cdot d^2(x, y), \\
 & \forall x \in X : \quad \delta_x \in [0, 1].
 \end{aligned}$$

Here $f(k)$ is a function to be determined. We claim that for an appropriate choice of f , this SDP is a relaxation for the problem of minimizing the outlier set size such that all non-outlier elements in X can be embed into ℓ_2 with distortion c . In particular, given a (k, c) -outlier embedding from (X, d) into ℓ_2 , we can find a feasible solution for the SDP with value at most k . For Theorem 2.9, it will be sufficient to set $f(k) = (125 \cdot c \cdot H_k)^2$.

LEMMA 3.1. *Let (X, d) be a finite metric space with expanding embedding $\alpha : S \rightarrow \ell_2$ of distortion at most c for $S \subseteq X$. Then if there exists a $(g(k)/c)$ -Lipschitz extension (or a weak $g(k)$ -nested embedding) of α with $k = |X \setminus S|$, (Outlier SDP) with $f(k) := g(k)^2$ has a feasible solution with value equal to k .*

Proof. Let $K = X \setminus S$. We will construct a feasible solution for (Outlier SDP). Set δ_x to 1 if $x \in K$ and 0 otherwise. Set \vec{v}_x to $\alpha(x)$. Clearly $\sum_{x \in X} \delta_x = k$. We show that this setting of the variables satisfies the given constraints.

Consider a constraint corresponding to $x, y \in X \setminus K$. Then we have that $(1 - \delta_x - \delta_y) \cdot d(x, y)^2 = d(x, y)^2 \leq \|\alpha(x) - \alpha(y)\|_2^2 \leq c^2 \cdot d(x, y)^2 = (c^2 + (\delta_x + \delta_y)f(k)) \cdot d(x, y)^2$ by the facts we have already asserted about α .

Next, consider a constraint corresponding to $x \in X, y \in K$. In this case, $\delta_y = 1$ and $(1 - \delta_x - \delta_y) \leq 0$, so the first inequality in (3.6) is satisfied. On the other hand, by the definition of f and α , $\|\alpha(x) - \alpha(y)\|_2^2 \leq f(k) \cdot d(x, y)^2 \leq (c^2 + (\delta_x + \delta_y)f(k)) \cdot d(x, y)^2$, which gives us the second inequality. Thus, the solution (δ, \vec{v}) is a feasible solution with value k . \square

Observe that f is a function of k , and so in order to set up and solve the SDP, we require knowing the value of the parameter k . We can get around this by setting $k = 1, 2, \dots$, and so on until we find the smallest value of k for which the SDP with parameter $f(k)$ has a feasible solution of value at most k . Rounding this solution then gives the desired theorem.

We are now ready to prove Theorem 2.9.

Proof of Theorem 2.9. Suppose that (X, d) a (k, c) -outlier embedding into ℓ_2 . By Lemma 3.1, there exists a solution to the SDP (Outlier SDP) with $f(k) = (125H_k \cdot c_S)^2$ with value at most k , which we can find efficiently by solving the SDP. Let $\{(\vec{v}_x, \delta_x)\}_{x \in X}$ denote such a solution. Let Δ be a parameter to be defined. Define $\alpha(x) \mapsto \frac{1}{\sqrt{1-2\Delta}}\vec{v}_x$, and $K \mapsto \{x : \delta_x \geq \Delta\}$. We claim that for an appropriate choice of Δ , the solution (K, α) satisfies the requirements of the theorem.

In particular, we note that $|K| \leq (\sum_{x \in X} \delta_x) / \Delta \leq k / \Delta$. To bound the distortion of α restricted to $X \setminus K$ by γc , let us consider some pair of points $x, y \in X \setminus K$, and recall that we have $\delta_x, \delta_y < \Delta$. Then, substituting $\vec{v}_x = \sqrt{1-2\Delta}\alpha(x)$ in (3.6) gives us:

$$(1 - 2\Delta) \cdot d^2(x, y) \leq (1 - 2\Delta)\|\alpha(x) - \alpha(y)\|_2^2 \leq (c^2 + 2\Delta f(k)) \cdot d(x, y)^2$$

The first inequality implies expansion. The second provides an upper bound on the distortion of:

$$\frac{c^2 + 2\Delta f(k)}{1 - 2\Delta}$$

ALGORITHM 4.1. Lipschitz extension

INPUT: Metric space (X, d) , $|X| = n$, subset $S \subseteq X$, embedding $\alpha_S : X \setminus S \rightarrow Y$ for some target metric (Y, d_Y) , and real number $\tau > 0$

OUTPUT: A randomized embedding $\alpha : X \rightarrow Y$ such that for all $x, y \in X$, $E[d_Y(\alpha(x), \alpha(y))] \leq 125H_k \cdot c \cdot d(x, y)$ for $k = |X \setminus S|$, and for all $x \in S$, $\alpha(x) = \alpha_S(x)$

- 1: $K \leftarrow X \setminus S$.
- 2: Define $\gamma : K \rightarrow S$ such that $\gamma(u) \in \arg \min_{v \in S} d(u, v)$. ▷ ie $\gamma(u)$ is one of u 's closest neighbors in S
- 3: Select b uniformly at random from the range $[2, \tau + 2]$
- 4: Select a uniformly random permutation $\pi : K \rightarrow [k]$ of the vertices in K
- 5: $K' \leftarrow K$
- 6: **for** $i = 1$ to k **do**
- 7: $u_i \leftarrow \pi^{-1}(i)$
- 8: $K_i \leftarrow \{v \in K' \mid d(v, u_i) \leq b \cdot d(v, \gamma(v))\}$ ▷ Let the “center” of K_i be u_i
- 9: $K' \leftarrow K' \setminus K_i$

10: Define an embedding $\alpha' : X \rightarrow \ell_p$ such that

$$\alpha'(v) = \begin{cases} \alpha_S(v) & \text{if } v \in S \\ \alpha_S(\gamma(u_i)) & \text{if } v \in K_i \text{ and with } u_i \text{ being the center of } K_i \end{cases}.$$

11: Output α' as α

Figure 1: Algorithm for finding a Lipschitz extension for finite metrics

Setting this quantity equal to $\gamma^2 c^2$ and solving for Δ gives us $\Delta = \frac{c^2(\gamma^2-1)}{2f(k)+2c^2\gamma^2}$ and $|K| \leq \frac{2f(k)+2c^2\gamma^2}{c^2(\gamma^2-1)}k$. □

4 Lipschitz extensions

In this section, we will show existence of Lipschitz extensions of embeddings into $\ell_p, p \geq 1$ space for arbitrary ℓ_p . To do this, we first give a randomized Lipschitz extension of an embedding $\alpha_S : S \subseteq X \rightarrow Y$ for (X, d) and (Y, d_Y) being metric spaces with $|X \setminus S|$ finite.¹ In the case that (Y, d_Y) is an ℓ_p metric, we show that because the output of the algorithm is over a distribution of finite support, averaging over the distribution will result in an embedding in which no distance is stretched too much. In this step, we fundamentally use the fact that the embedding is a Lipschitz extension and not just a weak nested embedding, as the averaging leaves the embeddings of nodes in S the same, whereas averaging in general may cause contraction compared to the expectation itself. Our algorithm for the randomized Lipschitz extension, Algorithm 4.1, is formally specified below.

Let c_S denote the distortion of α_S . We now show that if α is the output of Algorithm 4.1 on input $((X, d), S, \alpha_S, \tau)$ with $\tau = 2$, the distortion between elements in S is at most c_S and all other distortion is at most $125c_S \cdot H_k$ in expectation.

The following lemma states our expansion bounds; we prove it in the following subsection. Throughout these arguments we assume that $\tau = 2$, although it is possible to obtain slightly better distortion bounds by choosing a value for τ carefully. We present general versions of the lemmas, exhibiting the dependence of the bounds on τ in Appendix A. Section 4.2 gives a deterministic construction of a Lipschitz extension and thus proves Theorem 2.6.

LEMMA 4.1. *Let $\alpha \leftarrow$ Algorithm 4.1 $((X, d), S, p, \alpha_S, \alpha_X, \tau)$ with $\tau = 2$. Then we have the following bounds on the expansion for each pair $x, y \in X$:*

¹Note that line 4 of the randomized algorithm given here is inspired by the algorithm given by [18] for randomized embeddings of metrics into trees. In this case however, we want to group together nodes that are close to each other relative to the “good” set of nodes, as this will allow us to place nodes at the same spot as good nodes that they are relatively close to.

membership of x and y	restrictions on $d(x, y)$	upper bound on expected distortion	
$x, y \in S$	none	c_S	(a)
$x \in S, y \in X \setminus S$	none	$10c_S$	(b)
$x, y \in X \setminus S$	$d(x, \gamma(x)) \leq 2 \cdot d(x, y)$	$50c_S$	(c)
$x, y \in X \setminus S$	$d(x, \gamma(x)), d(y, \gamma(y)) > 2 \cdot d(x, y)$	$125c_S \cdot H_k$	(d)

Table 1: Summary of the bounds in Lemma 4.1. Let $\alpha \leftarrow$ Algorithm 4.1 $((X, d), S, \alpha_S, \tau)$ where $\alpha_S : S \rightarrow \ell_p$ is an expanding embedding of distortion at most c_S . Then the third column of the table gives an upper bound on the expected value of $d_Y(\alpha(x), \alpha(y))$ where x and y meet the criteria of the first two columns. Here γ and K_i are as defined in lines (2) and (7)-(8) of the algorithm.

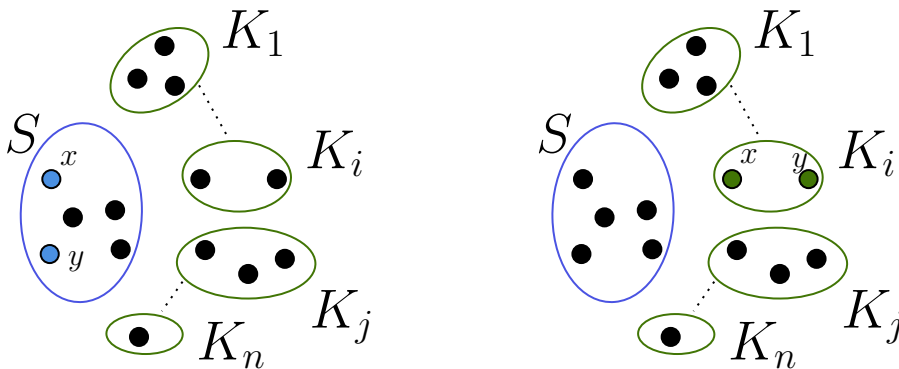


Figure 2: Left: Visualization of nodes referenced in Lemma 4.1 (a) and Lemma 5.2 (a) Right: Visualization of nodes referenced in Lemma 5.2 (b)

- (a) If $x, y \in S$, then $d_Y(\alpha(x), \alpha(y)) \leq c_S \cdot d(x, y)$.
- (b) If $x \in S, y \in X \setminus S$, then $d_Y(\alpha(x), \alpha(y)) \leq 10c_S \cdot d(x, y)$
- (c) If $x, y \in X \setminus S$ and $d(x, \gamma(x)) \leq 2 \cdot d(x, y)$ for γ as defined in line 2 of the algorithm, then $d_Y(\alpha(x), \alpha(y)) \leq 50c_S \cdot d(x, y)$.
- (d) If $x, y \in X \setminus S$ and $d(x, \gamma(x)), d(y, \gamma(y)) > 2 \cdot d(x, y)$ for γ as defined in line 2 of the algorithm, then $E_\alpha[d_Y(\alpha(x), \alpha(y))] \leq 125c_S \cdot d(x, y)$.

4.1 Proofs of expansion bounds The bounds in Lemma 4.1 are summarized in Table 1. We will prove each statement separately.

Lemma 4.1 (a) is automatically true by definition of α .

Now consider the distortion between outliers and non-outliers (i.e. nodes in S compared to nodes in K).

Proof of Lemma 4.1 (b). Let i be the index of x 's cluster, that is, $x \in K_i$. First, suppose that $x = u_i$, that is, x is the center of the cluster it's in. Then we get that $\alpha_S(x) = \alpha_S(\gamma(x))$. Clearly this implies that if $y = \gamma(x)$, the distortion is at most c_S . Otherwise, we have $d(y, x) \geq d(\gamma(x), x)$, so we get

$$\begin{aligned}
 d_Y(\alpha(x), \alpha(y)) &= d_Y(\alpha_S(\gamma(x)), \alpha_S(y)) \\
 &\leq c_S \cdot d(y, \gamma(x)) \\
 &\leq c_S \cdot d(y, x) + c_S \cdot d(x, \gamma(x)) \\
 &\leq 2c_S \cdot d(x, y),
 \end{aligned}$$

where the second line is by the fact that distortion of α_S is at most c_S , the second is by the triangle inequality, and the third is by the fact that $d(x, \gamma(x)) \leq d(x, y)$ by definition of γ .

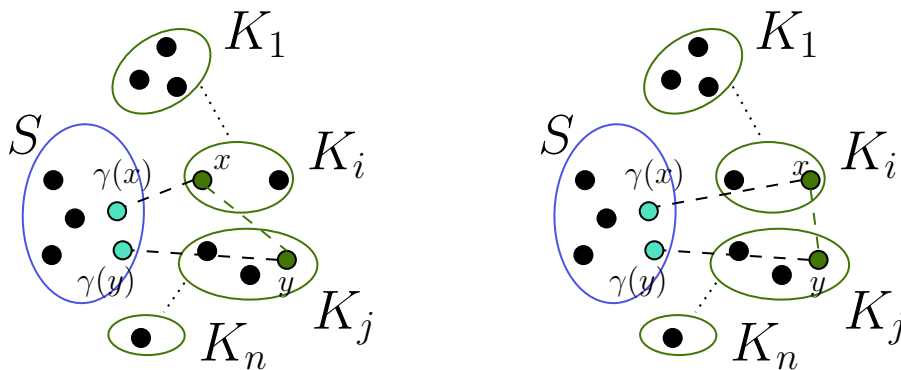


Figure 3: Left: Visualization of nodes referenced in Lemma 4.1 (c) and Lemma 5.2 (d); Right: Visualization of nodes referenced in Lemma 4.1 (d) and Lemma 5.2 (e)

Now consider an arbitrary $x \in K_i$ for some i , and let u_i be the center of K_i (i.e. $\pi(u_i) = i$). We get the following:

$$\begin{aligned} d_Y(\alpha(x), \alpha(y)) &= d_Y(\alpha(u_i), \alpha(y)) \\ &\leq 2c_S \cdot d(u_i, y) \\ &\leq 2c_S \cdot (d(u_i, x) + d(x, y)) \\ &\leq 10c_S \cdot d(x, y), \end{aligned}$$

where the first line is because x is also assigned to the same position as u_i 's closest neighbor, the second line is by the argument we just made for u_i , the third line is by the triangle inequality, and the last line is by the fact that $d(u_i, x) \leq 4d(x, \gamma(x)) \leq 4d(x, y)$ since $b \leq 4$ and $x \in K_i$. \square

Next we consider comparing two the distance of two nodes $x, y \in X \setminus S$. First we consider the case that at least one of the nodes has a relatively short distance to S compared to the distance to the other node.

Proof of Lemma 4.1 (c). Let γ be as defined in line (2) of the algorithm. We get

$$\begin{aligned} d_Y(\alpha(x), \alpha(y)) &\leq d_Y(\alpha(x), \alpha(\gamma(x))) + d_Y(\alpha(\gamma(x)), \alpha(y)) \\ &\leq 10c_S \cdot d(x, \gamma(x)) + 10c_S \cdot d(\gamma(x), y) \\ &\leq 20c_S \cdot d(x, \gamma(x)) + 10c_S \cdot d(x, y) \\ &\leq 40c_S \cdot d(x, y) + 10c_S \cdot d(x, y) \\ &= 50c_S \cdot d(x, y), \end{aligned}$$

where the first line is by the triangle inequality on Y , the second is by Lemma 4.1 (b), the third is by the triangle inequality on X , and the fourth is by the fact that $d(x, \gamma(x)) \leq 2d(x, y)$, a condition of this lemma. \square

Now we consider the final case, where we must consider expected distance.

Proof of Lemma 4.1 (d). Let γ be as defined in line (2) of the algorithm. We will say that x and y are “split” if $x \in K_i, y \in K_j$ for $i \neq j$ and K_i, K_j as defined in line (8) of the algorithm. Let u_i, u_j be as defined in line (7) of the algorithm for the same choice of i, j respectively.

- First consider the worst-case distortion when x and y are *not* split. Then $x, y \in K_i$ for some i , so $d_Y(\alpha(x), \alpha(y)) = 0$.

- Now consider the worst-case distortion when x and y are split.

$$\begin{aligned}
d_Y(\alpha(x), \alpha(y)) &\leq d_Y(\alpha(x), \alpha(\gamma(x))) + d_Y(\alpha(\gamma(x)), \alpha(y)) \\
&\leq 10c_S \cdot d(x, \gamma(x)) + 10c_S \cdot d(\gamma(x), y) \\
&\leq 20c_S \cdot d(x, \gamma(x)) + 10c_S \cdot d(x, y) \\
&\leq 20c_S \cdot d(x, \gamma(x)) + 5c_S \cdot d(x, \gamma(x)) \\
&= 25c_S \cdot d(x, \gamma(x)),
\end{aligned}$$

where the first inequality is by the triangle inequality on Y , the second is by Lemma 4.1 (b), the third is by the triangle inequality on X , and the fourth is by the fact that $d(x, y) < \frac{1}{2}d(x, \gamma(x))$ as a condition of this part of the lemma. Note that an identical analysis shows that $d_Y(\alpha(x), \alpha(y)) \leq 25c_S \cdot d(y, \gamma(y))$.

Now consider the probability of x and y being split. First, let us fix some node u chosen in some iteration of Step (7) of the algorithm and let K_u be the cluster formed by this vertex. Suppose that the placement of x and y is undetermined prior to this point of time. We will bound the probability that x and y are split by u , that is, exactly one of these vertices ends up in the cluster K_u . Without loss of generality, assume that $\frac{d(x, u)}{d(x, \gamma(x))} \leq \frac{d(y, u)}{d(y, \gamma(y))}$. This implies that if x, y are split, then $x \in K_u, y \notin K_u$ and we have

$$\frac{d(x, u)}{d(x, \gamma(x))} \leq b \leq \frac{d(y, u)}{d(y, \gamma(y))}.$$

This implies b must fall in a range of width

$$\begin{aligned}
W &\leq \frac{d(y, u)}{d(y, \gamma(y))} - \frac{d(x, u)}{d(x, \gamma(x))} \\
&\leq \frac{d(x, u) + d(x, y)}{d(x, \gamma(y)) - d(x, y)} - \frac{d(x, u)}{d(x, \gamma(x))} \\
&\leq \frac{d(x, u) + d(x, y)}{d(x, \gamma(x)) - d(x, y)} - \frac{d(x, u)}{d(x, \gamma(x))} \\
&= d(x, y) \cdot \frac{d(x, \gamma(x)) + d(x, u)}{d(x, \gamma(x)) \cdot (d(x, \gamma(x)) - d(x, y))} \\
&\leq d(x, y) \cdot \frac{5}{(d(x, \gamma(x)) - d(x, y))} \\
&\leq 10 \cdot \frac{d(x, y)}{d(x, \gamma(x))},
\end{aligned}$$

where the second inequality is by applying the triangle inequality twice, the third inequality is by the fact that $\gamma(x)$ is a closest node in S to x , and the fourth is by cross-multiplying. The fifth line is by the fact that $x \in K_u$, which implies $d(x, u) \leq \max$ possible value of $b \cdot d(x, \gamma(x))$. Finally, the sixth follows from the fact that $d(x, y) < \frac{1}{2} \cdot d(x, \gamma(x))$, a condition of the lemma.

Next we will bound the overall probability that x and y are split by *some* node u . For a vertex u , define $\beta_u = \min \left\{ \frac{d(x, u)}{d(x, \gamma(x))}, \frac{d(y, u)}{d(y, \gamma(y))} \right\}$. This is the smallest value of b at which the cluster K_u formed by u contains either x or y . Consider ordering vertices u in K in increasing order of β_u , and let $\text{index} : K \rightarrow [k]$ denote this ordering. We say that a node u “decides” the pair (x, y) if at least one of x and y is in K_u . u can decide (x, y) iff $b \geq \beta_u$ (i.e. at least one of x, y meets the criteria to be in the cluster u is the center of). This implies that if u and u' satisfy $\text{index}(u) < \text{index}(u')$ and u appears before u' in the ordering π , then at the time we consider u' in Step (7), either it is the case that $b \geq \beta_u$ and (x, y) has already been decided by u , or it is the case that $b < \beta_u < \beta_{u'}$, in which case u' cannot decide (x, y) . Therefore, in either case, u' does not decide (x, y) , and consequently does not split them.

In other words, in order for a vertex u to be able to split (x, y) , it must be the case that among the $\text{index}(u)$ vertices before u in the index ordering (and including u itself), u is the first vertex to appear in the ordering π .

Let us call this latter event E_u , and observe that this event is independent of the choice of b — it only depends on the choice of the permutations π and \mathbf{index} . We also note that $\Pr_\pi[E_u] = 1/\mathbf{index}(u)$.

We can now write down the probability that x and y are split as follows, using the fact that from our discussion above, $\Pr_b[u \text{ splits } (x, y) | \neg E_u] = 0$ for all u .

$$\begin{aligned} \Pr_{\pi,b}[x, y \text{ are split}] &= \sum_{u \in K} \Pr_b[u \text{ splits } (x, y) | E_u] \cdot \Pr_\pi[E_u] + \Pr_b[u \text{ splits } (x, y) | \neg E_u] \cdot \Pr_\pi[\neg E_u] \\ &\leq \sum_{u \in K} \frac{W}{\tau} \cdot \frac{1}{\mathbf{index}(u)} \\ &= \sum_{u \in K} \frac{10}{2} \frac{d(x, y)}{d(x_u, \gamma(x_u))} \cdot \frac{1}{\mathbf{index}(u)}. \end{aligned}$$

Here we used x_u to denote the node in $\{x, y\}$ that is closer to u , and we substituted expressions from above for W , τ , and $\Pr_\pi[E_u]$.

Finally, let us consider the expected distance between $\alpha(x)$ and $\alpha(y)$. Note that by our earlier analysis, the distance between x and y when they are split by u is at most $25c_S \cdot d(x_u, \gamma(x_u))$. Thus, we can compute the expected distance between x and y as follows:

$$\begin{aligned} E_{\pi,b}[d_Y(\alpha(x), \alpha(y))] &\leq 0 \cdot \Pr[(x, y) \text{ are not split}] \\ &\quad + \sum_{u \in K} \Pr_b[x, y \text{ are split by } u | E_u] \cdot \Pr_\pi[E_u] \cdot 25c_S \cdot d(x_u, \gamma(x_u)) \\ &\leq \sum_{u \in K | \beta_u \leq \tau+2} \frac{5}{\mathbf{index}(u)} \cdot \frac{d(x, y)}{d(x_u, \gamma(x_u))} \cdot 25c_S \cdot d(x_u, \gamma(x_u)) \\ &\leq 125c_S \cdot d(x, y) \cdot \sum_{i=1}^k \frac{1}{i} \\ &= 125c_S \cdot H_k \cdot d(x, y), \end{aligned}$$

where H_k is the k th Harmonic number. □

4.2 Deterministic extensions In this section we prove Theorem 2.6. Note that in the previous part, we were able to allow the target metric to be arbitrary, but now we require that it be some ℓ_p space so that we can “average” the possible outputs of the embedding. Thus, in this section we assume that Y is \mathbb{R}^n with the ℓ_p metric.

Proof of Theorem 2.6. We have shown that for all $x, y \in X \setminus K$, $\|\alpha(x) - \alpha(y)\|_p = d(x, y)$ and for all $x, y \in X$, $E_{\pi,b}[\|\alpha(x) - \alpha(y)\|_2] \leq 125c_S \cdot d(x, y)$. We can now use the convexity of the ℓ_p norm to upper bound the distance between the *expected* points. In particular, note that the point that x is mapped to is completely determined by the choice of subsets K_i . There are at most $2^{2^{|K|}}$ ways to partition K , so the set of points over which $\alpha(x)$ is chosen is finite, and we can assign finite probability p_t to $\Pr[\alpha(x) = \alpha^t(x)]$ such that α^t is a function in the support of α and the sum over this probability for all points in the support is 1. Let $E[\alpha(x)] := E_{\pi,b}[\alpha(x)]$ be $\sum_t p_t \cdot \alpha^t(x)$ where the sum is over all functions α^t points in the support of α . We claim that the embedding $\alpha^* : X \rightarrow \ell_p^n$ defined by $\alpha^*(x) := E[\alpha(x)]$ is a Lipschitz extension of α_S . In particular, we will use the Banach space properties of the ℓ_p norm to obtain our desired bound.

$$\begin{aligned}
\|\alpha^*(x) - \alpha^*(y)\|_p &= \|E[\alpha(x)] - E[\alpha(y)]\|_p \\
&= \left\| \sum_t p_t \cdot \alpha^t(x) - \sum_t p_t \cdot \alpha^t(y) \right\|_p \\
&= \left\| \sum_t (p_t \alpha^t(x) - p_t \alpha^t(y)) \right\|_p \\
&\leq \sum_t \|p_t \cdot (\alpha^t(x) - \alpha^t(y))\|_p \\
&= \sum_t p_t \|\alpha^t(x) - \alpha^t(y)\|_p \\
&= E_{\pi, b} [\|\alpha(x) - \alpha(y)\|_p] \\
&\leq 125c_S \cdot H_k \cdot d(x, y),
\end{aligned}$$

where the first line is by definition of α^* , the second is by definition of $E[\alpha(x)]$, $E[\alpha(y)]$, the third is by regrouping, the fourth is by the triangle inequality of ℓ_p , the fifth is by scalar properties of ℓ_p , the sixth is by the definition of expectation, and the seventh is by our bounds from the previous subsection.

Note that for $x \in S$, $\alpha^t(x) = \alpha^t(x)$ for all α^t in the support of α . Thus, $\alpha^*(x) = \alpha^t(x) = \alpha_S(x)$. \square

In fact, in the proof of the previous theorem, the properties of ℓ_p space that we used were properties of any Banach space, so Theorem 2.6 applies when the destination metric is any Banach space.

5 Nested compositions into ℓ_1

In this section, we will give a modification of Algorithm 4.1 when (Y, d_Y) is ℓ_1 space that will allow us to avoid excessive contraction. We will then use this to argue that compositions of nested embeddings exist for embeddings into ℓ_1 . This will allow us to prove Theorems 2.7 and 2.8.

As in the previous case, we first give Algorithm 5.1, a randomized algorithm that will have good expected distortion for every pair, but now we will work to ensure the distances are not too contracted in addition to not being too expanded.

Note that Algorithm 5.1 appends together a bunch of embeddings, one for S and one for each K_i . The idea is that the main source of our contraction in the previous algorithm comes from the fact that many points may be mapped to the same place. To avoid this, we will use our embedding α_X to add some extra indices that will distinguish points that end up in the same set from each other and from their center's neighbor.

Let c_S and c_X denote the distortion of α_S and α_X respectively. We now show that if α is the output of Algorithm 5.1 on input $((X, d), S, \alpha_S, \alpha_X, \tau)$ with $\tau = 2$, the distortion between elements in S is at most c_S and all other distortion is at most $(\frac{155}{2} \cdot H_k \cdot c_S + (\frac{225}{2} \cdot H_k + 1) \cdot c_X)$ in expectation. Our argument is broken into two parts: Lemma 5.1 shows that the embedding α has low contraction; Lemma 5.2 bounds the amount by which every distance expands.

We state the lemmas first and then prove them in the following subsections. Throughout these arguments we assume that $\tau = 2$, although it is possible to obtain slightly better distortion bounds by choosing a value for τ carefully. We present general versions of the lemmas, exhibiting the dependence of the bounds on τ in Appendix A.

LEMMA 5.1. *Let $\alpha \leftarrow$ Algorithm 5.1 $((X, d), S, \alpha_S, \alpha_X, \tau)$ with $\tau = 2$. Then for all $x, y \in X$, we have $\|\alpha(x) - \alpha(y)\|_1 \geq d(x, y)$.*

LEMMA 5.2. *Let $\alpha \leftarrow$ Algorithm 5.1 $((X, d), S, \alpha_S, \alpha_X, \tau)$ with $\tau = 2$. Then we have the following bounds on the expansion for each pair $x, y \in X$:*

(a) *If $x, y \in S$, then $\|\alpha(x) - \alpha(y)\|_1 \leq c_S \cdot d(x, y)$.*

ALGORITHM 5.1. Nested embeddings

INPUT: Metric space (X, d) , subset $S \subseteq X$, expanding embedding $\alpha_S : X \setminus S \rightarrow \ell_1$, expanding embedding $\alpha_X : X \rightarrow \ell_1$, and real number $\tau > 0$

OUTPUT: A randomized expanding embedding $\alpha : X \rightarrow \ell_1$ such that for all $x, y \in S$, $d(x, y) \leq \|\alpha(x) - \alpha(y)\|_1 \leq c_S \cdot d(x, y)$ and for all $x, y \in X$, $d(x, y) \leq E[\|\alpha(x) - \alpha(y)\|_1] \leq g(c_S, c_X) \cdot d(x, y)$.

- 1: $K \leftarrow X \setminus S$.
- 2: Define a function $\gamma : K \rightarrow S$ such that $\gamma(u) \in \arg \min_{v \in S} d(u, v)$. \triangleright ie $\gamma(u)$ is one of u 's closest neighbors in S
- 3: Select b uniformly at random from the range $[2, \tau + 2]$
- 4: Select a uniformly random permutation $\pi : K \rightarrow [k]$ of the vertices in K
- 5: $K' \leftarrow K, i \leftarrow 1$
- 6: **for** $i = 1$ to k **do**
- 7: $u_i \leftarrow \pi^{-1}(i)$
- 8: $K_i \leftarrow \{v \in K' \mid d(v, u_i) \leq b \cdot d(v, \gamma(v))\}$ \triangleright Let the "center" of K_i be u_i
- 9: Define an embedding $\alpha_i : X \rightarrow \ell_1$ such that

$$\alpha_i(v) = \begin{cases} \alpha_X(v) & v \in K_i \\ \alpha_X(\gamma(u_i)) & v \notin K_i \end{cases}$$

- 10: $K' \leftarrow K' \setminus K_i$
- 11: Define an embedding $\alpha' : X \rightarrow \ell_1$ such that

$$\alpha'(v) = \begin{cases} \alpha_S(v) & \text{if } v \in S \\ \alpha_S(\gamma(u_i)) & \text{if } v \in K_i \text{ and with } u_i \text{ being the center of } K_i \end{cases}$$

- 12: Define an embedding $\alpha : X \rightarrow \ell_1$ such that $\alpha(v) \mapsto (\alpha'(v)|\alpha_1(v)|\cdots|\alpha_i(v))$ \triangleright here $|$ denotes concatenation
 - 13: Output α
-

Figure 4: Algorithm for finding a nested embedding

- (b) If $x, y \in K_i$, then $\|\alpha(x) - \alpha(y)\|_1 \leq c_X \cdot d(x, y)$.
- (c) If $x \in S, y \in X \setminus S$, then $\|\alpha(x) - \alpha(y)\|_1 \leq [7 \cdot c_S + 9 \cdot c_X] \cdot d(x, y)$.
- (d) If $x, y \in X \setminus S$ and $d(x, \gamma(x)) \leq 2 \cdot d(x, y)$ for γ as defined in line 2 of the algorithm, then $\|\alpha(x) - \alpha(y)\|_1 \leq [31 \cdot c_S + 45 \cdot c_X] \cdot d(x, y)$.
- (e) If $x, y \in X \setminus S$ and $d(x, \gamma(x)), d(y, \gamma(y)) > 2 \cdot d(x, y)$ for γ as defined in line 2 of the algorithm, then $E_\alpha[\|\alpha(x) - \alpha(y)\|_1] \leq (\frac{155}{2} \cdot H_k \cdot c_S + (\frac{225}{2} \cdot H_k + 1) \cdot c_X) \cdot d(x, y)$.

5.1 Proof of contraction bounds *Proof of Lemma 5.1.* Let γ be as defined in line (2) of the algorithm and let the u_i and K_i be as defined in lines (7) and (8) of the algorithm. We divide into cases.

1. If $x, y \in S$, then $\|\alpha(x) - \alpha(y)\|_1 = \|\alpha'(x) - \alpha'(y)\|_1 \geq d(x, y)$ because α_S is expanding
2. If $x, y \in K_i$ for some K_i defined in line (8) of the algorithm or if $x \in K_i, y = \gamma(u_i)$, their embeddings differ only on the coordinates associated with α_i , so we have

$$\begin{aligned} \|\alpha(x) - \alpha(y)\|_1 &= \|\alpha_i(x) - \alpha_i(y)\|_1 \\ &= \|\alpha_X(x) - \alpha_X(y)\|_1 \\ &\geq d(x, y), \end{aligned}$$

where the second line is by definition of α_i and the last line is because α_X is expanding.

3. If $x \in S, y \notin S$, then let $y \in K_i$ for some i as defined in line (8) of the algorithm and let u_i be the center for this i defined in line (7). This implies that $\alpha(x)$ and $\alpha(y)$ differ only on coordinates associated with α' and α_i .

$$\begin{aligned} \|\alpha(x) - \alpha(y)\|_1 &= \|\alpha'(x) - \alpha'(y)\|_1 + \|\alpha_i(x) - \alpha_i(y)\|_1 \\ &\geq \|\alpha_S(x) - \alpha_S(\gamma(u_i))\|_1 + \|\alpha_X(\gamma(u_i)) - \alpha_X(y)\|_1 \\ &\geq d(x, \gamma(u_i)) + d(\gamma(u_i), y) \\ &\geq d(x, y), \end{aligned}$$

where the second line is by definition of α' and α_i , the third is by cases 1 and 2 of this lemma, and the fifth is by the triangle inequality.

4. If $x \in K_i, y \in K_j$ for some K_i, K_j defined in line (8) of the algorithm. Let u_i, u_j be the centers of these sets as defined in line (7) of the algorithm. Note that $\alpha(x)$ and $\alpha(y)$ differ only on the indices associated with α', α_i , and α_j . Thus we get

$$\begin{aligned} \|\alpha(x) - \alpha(y)\|_1^p &= \|\alpha'(x) - \alpha'(y)\|_1 + \|\alpha_i(x) - \alpha_i(y)\|_1 + \|\alpha_j(x) - \alpha_j(y)\|_1 \\ &\geq \|\alpha_S(\gamma(u_i)) - \alpha_S(\gamma(u_j))\|_1 + \|\alpha_X(\gamma(u_i)) - \alpha_X(x)\|_1 + \|\alpha_X(\gamma(u_j)) - \alpha_X(y)\|_1 \\ &\geq d(\gamma(u_i), \gamma(u_j)) + d(\gamma(u_i), x) + d(\gamma(u_j), y) \\ &\geq d(x, y), \end{aligned}$$

where the second line is by definition of the three embeddings, the third line is by cases 1 and 2 of this lemma, and the fifth is by the triangle inequality.

□

Note that if we were to use Algorithm 5.1 to nest embeddings into more general ℓ_p spaces, we can use the same analysis as above, but rather than adding the distance between the concatenated vectors, we will need to raise the distances to the power of p , add them, and take the sum to the power $1/p$. Then by applying the power mean inequality we will get that contraction is at most $3^{-1+\frac{1}{p}}$ for all pairs of points, and at most 1 for pairs in S .

membership of x and y	restrictions on $d(x, y)$	upper bound on expected distortion	
$x, y \in S$	none	c_S	(a)
$x, y \in K_i$	none	c_X	(b)
$x \in S, y \in X \setminus S$	none	$[7 \cdot c_S + 9 \cdot c_X]$	(c)
$x, y \in X \setminus S$	$d(x, \gamma(x)) \leq 2 \cdot d(x, y)$	$[31 \cdot c_S + 45 \cdot c_X]$	(d)
$x, y \in X \setminus S$	$d(x, \gamma(x)), d(y, \gamma(y)) > 2 \cdot d(x, y)$	$(\frac{155}{2} \cdot H_k \cdot c_S + (\frac{225}{2} \cdot H_k + 1) \cdot c_X)$	(e)

Table 2: Summary of the bounds in Lemma 5.2. Let $\alpha \leftarrow$ Algorithm 5.1($(X, d), S, p, \alpha_S, \alpha_X, \tau$) where $\alpha_S : S \rightarrow \ell_p$ is an expanding embedding of distortion at most c_S and $\alpha_X : X \rightarrow \ell_p$ is an expanding embedding of distortion at most c_X . Then the third column of the table gives an upper bound on the the expected value of $\|\alpha(x) - \alpha(y)\|_p$ where x and y meet the criteria of the first two columns. Here γ and K_i are as defined in lines (2) and (7)-(8) of the algorithm.

5.2 Proofs of expansion bounds The bounds in Lemma 5.2 are summarized in Table 2. We will prove each statement separately.

Proof of Lemma 5.2 (a). $u, v \notin K_i$ for all i , so $\alpha_i(u) = \alpha_i(v)$ for all i . Thus,

$$\begin{aligned} \|\alpha(u) - \alpha(v)\|_1 &= \|\alpha'(u) - \alpha'(v)\|_1 \\ &= \|\alpha_S(u) - \alpha_S(v)\|_1 \\ &\leq c_S \cdot d(u, v), \end{aligned}$$

where the last equality is by definition of α_S . \square

Proof of Lemma 5.2 (b). Since $x, y \in K_i$ for a fixed i , we have that $\alpha(x), \alpha(y)$ differ only on coordinates associated with α_i . Thus, we get

$$\begin{aligned} \|\alpha(x) - \alpha(y)\|_1 &= \|\alpha_i(x) - \alpha_i(y)\|_1 \\ &= \|\alpha_X(x) - \alpha_X(y)\|_1 \\ &\leq c_X \cdot d(x, y), \end{aligned}$$

where we have used the fact that $\alpha_X(u) = \alpha_i(u)$ for $x \in K_i$. \square

Now we consider the distortion between outliers and non-outliers. To prove these bounds, we will need to define extra points in the ℓ_p space that X is mapped to. Consider the points u_i defined in Step (7) of the algorithm. We will now define a point $\beta(u_i)$ for each such $u_i \in K$. If $u_i \in K_i$ we set $\beta(u_i) = \alpha(u_i)$. Otherwise if $u_i \in K_j$ for $j < i$, we set all of the coordinates of $\beta(u_i)$ to be the same as $\alpha(u_i)$, except for the coordinates associated with $\alpha_j(u_i)$, that are replaced with those of $\alpha_X(\gamma(u_j))$; and the coordinates associated with $\alpha_i(u_i)$, that are replaced with those for $\alpha_X(u_i)$. In other words, $\beta(u_i)$ is the point that u_i would be mapped to by α if it had so happened that u_i belonged to K_i .

Now we are ready to prove Lemma 5.2 (c). We break this proof up into two parts: Lemmas 5.3 and 5.4.

LEMMA 5.3. *Let $\alpha \leftarrow$ Algorithm 5.1($(X, d), S, \alpha_S, \alpha_X, \tau$) with $\tau = 2$. Consider $x \in S$; u_i as defined in Step (7) of the algorithm for some i ; and $\beta(u_i)$ as defined above. If $x = \gamma(u_i)$ where $\gamma : X \setminus S \rightarrow S$ is as defined in Step (2) of the algorithm, then $\|\alpha(x) - \beta(u_i)\|_1 \leq c_X \cdot d(x, u_i)$. For all other x , $\|\alpha(x) - \beta(u_i)\|_1 \leq (c_S + c_X) \cdot d(x, u_i)$.*

Proof. First, note that the point $\beta(u_i)$ is exactly the point that would have been assigned to $\alpha(u_i)$ if it was not placed in some K_j for $j < i$. Additionally, this means that $\beta(u_i)$ and $\alpha(x)$ are the same on the indices associated with α_j for all $j \neq i$ (namely they are assigned $\alpha_X(\gamma(u_j))$).

Now we have $\alpha_j(x) = \alpha_j(u_i)$ for all $j \neq i$. Thus, we have

$$\begin{aligned} \|\alpha(x) - \beta(u_i)\|_1 &= \|\alpha'(x)|\alpha_i(x) - \alpha'(u_i)|\alpha_X(u_i)\|_1 \\ &\leq \|\alpha'(x) - \alpha'(u_i)\|_1 + \|\alpha_i(x) - \alpha_X(u_i)\|_1. \end{aligned}$$

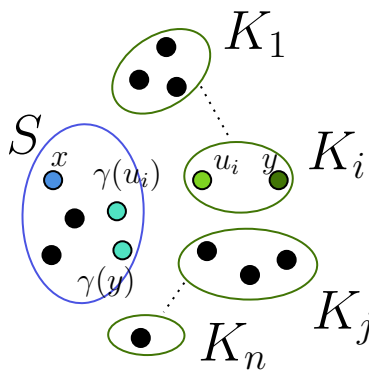


Figure 5: Visualization of nodes referenced in Lemmas 5.3 and 5.4 when $u_i \in K_i$.

We have two cases to consider.

1. If $x = \gamma(u_i)$, then we have $\alpha'(u_i) = \alpha'(x)$ by definition, so we have

$$\begin{aligned} \|\alpha(x) - \beta(u_i)\|_1 &\leq \|\alpha_i(x) - \alpha_X(u_i)\|_1 \\ &= \|\alpha_X(x) - \alpha_X(u_i)\|_1 \\ &\leq c_X \cdot d(x, u_i), \end{aligned}$$

where the last line is by the fact that α_X is an embedding with at most c_X distortion.

2. If $x \neq \gamma(u_i)$, we have $d(\gamma(u_i), u_i) \leq d(x, u_i)$ by definition of $\gamma(u_i)$. Then we use the triangle inequality as follows.

$$\begin{aligned} \|\alpha(x) - \beta(u_i)\|_1 &\leq \|\alpha(x) - \alpha(\gamma(u_i))\|_1 + \|\alpha(\gamma(u_i)) - \beta(u_i)\|_1 \\ &\leq c_S \cdot d(x, u_i) + c_X \cdot d(\gamma(u_i), u_i) \\ &\leq c_S \cdot d(x, u_i) + c_X \cdot d(x, u_i) \\ &= (c_S + c_X)d(x, u_i) \end{aligned}$$

where the second line comes from Lemma 5.2 (a) and the first case of this lemma.

□

Next we want to consider the distortion between more general members of a set K_i and members of $X \setminus K_i$. For the coming proofs, to show that x and y are not too distorted, we will generally focus on showing that the distance between x and y is at least a constant factor larger than the distance from x or y to some other point whose distance to x or y is already known not to be too distorted.

LEMMA 5.4. Let $\alpha \leftarrow$ Algorithm 5.1($(X, d), S, \alpha_S, \alpha_X, \tau$) with $\tau = 2$. Let K_i be as defined in Step (8) of the algorithm for some i . Consider $x \in S$ and $y \in K_i$ for some i . If $x = \gamma(y)$ where γ is as defined in line (2) of the algorithm, then we have $\|\alpha(x) - \alpha(y)\|_1 \leq [5 \cdot c_S + 9 \cdot c_X] \cdot d(x, y)$. For all other x , $\|\alpha(x) - \alpha(y)\|_1 \leq [7 \cdot c_S + 9 \cdot c_X] \cdot d(x, y)$.

Proof. Let $\gamma : X \setminus S \rightarrow S$ be as defined in Step (2) of the algorithm, u_i be as defined in Step (7) for i consistent with the lemma statement, and $\beta(u_i)$ be as defined above. We divide analysis into two cases.

1. If $x = \gamma(y)$, the lemma essentially comes from the triangle inequality and y 's relative closeness to u_i that it

must have if it is assigned to K_i .

$$\begin{aligned}
\|\alpha(x) - \alpha(y)\|_1 &= \\
\|\alpha(\gamma(y)) - \alpha(y)\|_1 &\leq \|\alpha(x) - \beta(u_i)\|_1 + \|\alpha(y) - \beta(u_i)\|_1 \\
&\leq [(c_S + c_X) \cdot d(x = \gamma(y), u_i) + c_X d(y, u_i)] \\
&\leq (c_S + c_X) \cdot [d(\gamma(y), y) + d(y, u_i)] + c_X \cdot d(y, u_i) \\
&= (c_S + c_X) \cdot d(\gamma(y), y) + (c_S + 2c_X) \cdot d(y, u_i) \\
&\leq (c_S + c_X) \cdot d(\gamma(y), y) + 4(c_S + 2c_X) \cdot d(y, \gamma(y)) \\
&= [5c_S + 9c_X] \cdot d(y, \gamma(y) = x).
\end{aligned}$$

Here the first inequality is by the fact that $\alpha(x)$ and $\alpha(y)$ may only differ on the indices associated with α' and the indices associated with α_i . The second inequality is by Lemma 5.3 and Lemma 5.2 (b), the third is by the triangle inequality, and the fourth line is by rearranging terms. The fifth line is by the fact that $d(y, u_i) \leq b \cdot d(y, \gamma(y))$ since y is in K_i and u_i is the center, and the fact that $b \leq 4$ no matter the result of the random choice in the algorithm.

2. If $x \neq \gamma(y)$, we have $d(\gamma(y), y) \leq d(x, y)$ by definition of $\gamma(y)$. Thus, we get

$$\begin{aligned}
\|\alpha(x) - \alpha(y)\|_1 &\leq \|\alpha(x) - \alpha(\gamma(y))\|_1 + \|\alpha(\gamma(y)) - \alpha(y)\|_1 \\
&\leq [c_S \cdot d(x, \gamma(y)) + [5c_S + 9c_X] \cdot d(\gamma(y), y)] \\
&\leq [c_S \cdot [d(x, y) + d(y, \gamma(y))] + [5c_S + 9c_X] \cdot d(\gamma(y), y)] \\
&\leq [7c_S + 9c_X] \cdot d(x, y),
\end{aligned}$$

where the second inequality is by Lemma 5.2 (a) and by the previous case of this lemma, the third inequality is by the triangle inequality, and the last inequality is because $\gamma(y)$ is a closest node in S to y .

□

Finally, we consider comparing two the distance of two nodes $x, y \in X \setminus S$. First we consider the case that at least one of the nodes has a relatively short distance to S compared to the distance to the other node.

Proof of Lemma 5.2 (d). Let γ be as defined in line (2) of the algorithm. We get

$$\begin{aligned}
\|\alpha(x) - \alpha(y)\|_1 &\leq \|\alpha(x) - \alpha(\gamma(x))\|_1 + \|\alpha(\gamma(x)) - \alpha(y)\|_1 \\
&\leq [5c_S + 9c_X] \cdot d(x, \gamma(x)) + [7c_S + 9c_X] \cdot d(\gamma(x), y) \\
&\leq [12c_S + 18c_X] \cdot d(x, \gamma(x)) + [7c_S + 9c_X] \cdot d(x, y) \\
&\leq [31c_S + 45c_X] \cdot d(x, y),
\end{aligned}$$

where the first inequality is by the triangle inequality, the second is by Lemma 5.3, the third is by the triangle inequality applied to $d(\gamma(x), y)$ and rearranging terms, and the fourth is by the fact that $d(x, \gamma(x)) \leq 2 \cdot d(x, y)$ and rearranging terms. □

Now we consider the final case, where we must consider expected distance.

Proof of Lemma 5.2 (e). Let γ be as defined in line (2) of the algorithm. We will say that x and y are “split” if $x \in K_i, y \in K_j$ for $i \neq j$ and K_i, K_j as defined in line (8) of the algorithm. Let u_i, u_j be as defined in line (7) of the algorithm for the same choice of i, j respectively.

- First consider the worst-case distortion when x and y are *not* split. Then $x, y \in K_i$ for some i . We have $\|\alpha(x) - \alpha(y)\|_1 \leq c_X \cdot d(x, y)$ by Lemma 5.2 (b), no matter the choice of b .

- Now consider the worst-case distortion when x and y are split such that x is placed in K_i and y is placed in K_j for $i < j$. We have

$$\begin{aligned} \|\alpha(x) - \alpha(y)\|_1 &\leq \|\alpha(x) - \alpha(\gamma(x))\|_1 + \|\alpha(\gamma(x)) - \alpha(y)\|_1 \\ &\leq [5c_S + 9c_X] \cdot d(x, \gamma(x)) + [7c_S + 9c_X] \cdot d(\gamma(x), y) \\ &\leq [5c_S + 9c_X] \cdot d(x, \gamma(x)) + [7c_S + 9c_X] \cdot [d(x, y) + d(x, \gamma(x))] \\ &\leq [5c_S + 9c_X] \cdot d(x, \gamma(x)) + [7c_S + 9c_X] \cdot \frac{3}{2}d(x, \gamma(x)) \\ &= \left[\frac{31}{2}c_S + \frac{45}{2}c_X\right] \cdot d(x, \gamma(x)), \end{aligned}$$

where the first inequality is by the triangle inequality, the second is by Lemma 5.4, the third inequality is by the triangle inequality on $d(\gamma(x), y)$, the fourth inequality is by the fact that $d(x, y) < \frac{1}{2}d(x, \gamma(x))$ by the condition of this lemma, and the final equality is by rearranging terms.

Note that the probability of x and y being split is no different in this case than in the proof of Lemma 4.1 (e). The only changes are to the amount of distortion incurred when they are split versus when they are not split. Using the same definitions as in the proof of Lemma 4.1 (e) and our earlier analysis that the distance between x and y when they are split by u is at most $\left[\frac{31}{2}c_S + \frac{45}{2}c_X\right] \cdot d(x_u, \gamma(x_u))$, we can compute the expected distance between x and y as follows:

$$\begin{aligned} E_{\pi, b}[\|\alpha(x) - \alpha(y)\|_1] &\leq c_X \cdot d(x, y) \cdot \Pr[(x, y) \text{ are not split}] \\ &\quad + \sum_{u \in K} \Pr_b[x, y \text{ are split by } u | E_u] \cdot \Pr_\pi[E_u] \cdot \left(\frac{31}{2}c_S + \frac{45}{2}c_X\right) \cdot d(x_u, \gamma(x_u)) \\ &\leq c_X \cdot d(x, y) + \sum_{u \in K | \beta_u \leq \tau+2} \frac{5}{\text{index}(u)} \cdot \frac{d(x, y)}{d(x_u, \gamma(x_u))} \cdot \left(\frac{31}{2}c_S + \frac{45}{2}c_X\right) \cdot d(x_u, \gamma(x_u)) \\ &\leq c_X \cdot d(x, y) + 5 \cdot \left(\frac{31}{2}c_S + \frac{45}{2}c_X\right) \cdot d(x, y) \cdot \sum_{i=1}^k \frac{1}{i} \\ &= \left(\frac{155}{2} \cdot H_k \cdot c_S + \left(\frac{225}{2} \cdot H_k + 1\right) \cdot c_X\right) \cdot d(x, y), \end{aligned}$$

where H_k is the k th Harmonic number. □

5.3 Deterministic nested embeddings In this section, we show that we can obtain $O(H_k)(c_S + c_X)$ -nested embeddings when the target metric is ℓ_1 . As in the proof of Lipschitz extensions for general ℓ_p spaces, this method is not efficient and only serves as a proof of existence.

Proof of Theorem 2.7. Consider the embedding α^{**} that is $\alpha^{**}(x) := \|\|_t p_t \alpha^t(x)$, where $\|\|_t$ implies concatenation over all choices of t and p_t, α^t are as defined in the proof of Theorem 2.6. Now we show that $\alpha^{**}(x) = E_{\pi, b}[x]$ for all $x \in X$.

$$\begin{aligned} \|\alpha^{**}(x) - \alpha^{**}(y)\|_1 &= \sum_t \|p_t \alpha^t(x) - p_t \alpha^t(y)\|_1 \\ &= \sum_t p_t \|\alpha^t(x) - \alpha^t(y)\|_1 \\ &= E_{\pi, b}[\|\alpha(x) - \alpha(y)\|_1], \end{aligned}$$

where the first line is by definition of α^{**} and the fact that for ℓ_1 , the norm of two concatenated vectors equals the sum of their individual norms. The second line is by the scalar properties of ℓ_p norms, and the third line is by

how we defined the p_t and α^t . Thus, we get that our bounds on distortion for individual pairs in our randomized embedding holds for all pairs simultaneously in this choice of embedding and we get existence as desired. Note that the embedding presented here may have a large (but finite) number of coordinates, but any ℓ_p^d metric on n points is isometrically embeddable into ℓ_p^n , so we obtain existence of an embedding with low distortion and at most n dimensions. \square

Note that trying to expand the above proof to work for other ℓ_p bounds we have the problem that the norm of vectors u concatenated with v is $(\|u\|_p^p + \|v\|_p^p)^{1/p}$, which is smaller than $\|u\|_p + \|v\|_p$ for $p > 1$. However, the randomized analysis goes through for any $p \geq 1$.

Proof of Theorem 2.8. Notice that Algorithm 5.1 can replace its use of α_X in line 8 with any embedding of $K_i \cup \{\gamma(u_i)\}$, and if we have an upper bound on such an embedding's distortion, we can replace c_X in all of the theorems and lemmas in this section with that bound. \square

6 Conclusion

In this paper, we give a bi-criteria approximation algorithm that given a constant c and metric (X, d) finds an $(O(k \log^2 k), O(c))$ -outlier embedding into ℓ_2 if the metric has a (k, c) -outlier embedding into ℓ_2 . In doing so, we show that given a metric space (X, d) , a c_S -distortion embedding of a subset $S \subseteq X$ into ℓ_1 , there exists a Lipschitz extension with Lipschitz factor at most $O(H_k \cdot c_S)$ on every pair of points. Additionally, when the target metric is ℓ_1 and we have an embedding of the entire space with distortion at most c_X , there exists a single (composition) embedding of X into ℓ_1 such that distortion between pairs of points in S is at most c_S and distortion between all pairs of points is at most $O(c_S + H_k \cdot c_X)$. We also leave several open questions on this topic. Among them, we ask:

- Is there a polynomial time algorithm that given constant c , finds an $(O(k), O(c))$ -embedding into ℓ_2 ?
- What bicriteria approximations can be obtained for outlier embeddings into ℓ_p for other values of p ?
- Do there exist nested compositions into ℓ_p for $p \neq 1$ that do not incur the contraction presented in this paper (i.e. “strong” nested embeddings)?
- Can the parameters for our nested embedding or Lipschitz extension algorithms be improved?
- Can our hardness of approximation result be extended to non-isometric outlier embeddings for large constants or slow-growing functions in n ?
- Can we obtain any larger concrete lower bounds on approximation factors for an optimal outlier set?

Acknowledgements

We are grateful to Ola Svensson for comments on a previous version of this paper that allowed us to simplify Algorithm 4.1 and improve our approximation factor for the number of outliers from $O(\log^4 k)$ to $O(\log^2 k)$.

This work was supported in part by NSF awards CCF-2217069 and CCF-2225259.

References

- [1] I. Abraham, Y. Bartal, J. Kleinberg, T.-H.H. Chan, O. Neiman, Kedar Dhamdhere, A. Slivkins, and Anupam Gupta. Metric embeddings with relaxed guarantees. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 83–100, 2005.
- [2] Noga Alon, Mihai Badoiu, Erik D. Demaine, Martin Farach-Colton, Mohammad Taghi Hajiaghayi, and Anastasios Sidiropoulos. Ordinal embeddings of minimum relaxation: General properties, trees, and ultrametrics. *ACM Trans. Algorithms*, 4(4):46:1–46:21, 2008.
- [3] Sanjeev Arora, László Lovász, Ilan Newman, Yuval Rabani, Yuri Rabinovich, and Santosh Vempala. Local versus global properties of metric spaces. *SIAM Journal on Computing*, 41(1):250–271, 2012.

- [4] Mihai Badoiu, Julia Chuzhoy, Piotr Indyk, and Anastasios Sidiropoulos. Low-distortion embeddings of general metrics into the line. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 225–233. ACM, 2005.
- [5] Mihai Badoiu, Julia Chuzhoy, Piotr Indyk, and Anastasios Sidiropoulos. Embedding ultrametrics into low-dimensional spaces. In Nina Amenta and Otfried Cheong, editors, *Proceedings of the 22nd ACM Symposium on Computational Geometry, Sedona, Arizona, USA, June 5-7, 2006*, pages 187–196. ACM, 2006.
- [6] Mihai Badoiu, Piotr Indyk, and Anastasios Sidiropoulos. Approximation algorithms for embedding general metrics into trees. In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 512–521. SIAM, 2007.
- [7] Jean Bourgain. On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52:46–52, 1985.
- [8] T. H. Hubert Chan, Michael Dinitz, and Anupam Gupta. Spanners with slack. In Yossi Azar and Thomas Erlebach, editors, *Algorithms – ESA 2006*, pages 196–207, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [9] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Local global tradeoffs in metric embeddings. *SIAM Journal on Computing*, 39(6):2487–2512, 2010.
- [10] Victor Chepoi, Feodor F. Dragan, Ilan Newman, Yuri Rabinovich, and Yann Vaxès. Constant approximation algorithms for embedding graph metrics into trees and outerplanar graphs. *Discret. Comput. Geom.*, 47(1):187–214, 2012.
- [11] Karine Chubarian and Anastasios Sidiropoulos. Computing bi-lipschitz outlier embeddings into the line. In *International Workshop and International Workshop on Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2020.
- [12] James A. Clarkson. Uniformly convex spaces. *Transactions of the American Mathematical Society*, 40:396–414, 1936.
- [13] Mark de Berg, Krzysztof Onak, and Anastasios Sidiropoulos. Fat polygonal partitions with applications to visualization and embeddings. *J. Comput. Geom.*, 4:212–239, 2010.
- [14] M. Deza and S. Shpectorov. Recognition of the ℓ_1 -graphs with complexity $O(nm)$, or football in a hypercube. *European Journal of Combinatorics*, 17(2-3):279–289, 1996.
- [15] Michel Deza and Monique Laurent. The cut cone and ℓ_1 metrics. In R. L. Graham, B. Korte, Bonn L. Lovasz, A. Wigderson, and G.M. Ziegler, editors, *Geometry of Cuts and Metrics*, chapter 4, pages 37–52. Springer, 1997.
- [16] D. Ž. Djoković. Distance-preserving subgraphs of hypercubes. *Journal of Combinatorial Theory, Series B*, 14(3):263–267, 1973.
- [17] Jeff Edmonds, Anastasios Sidiropoulos, and Anastasios Zouzias. Inapproximability for planar embedding problems. In Moses Charikar, editor, *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*, pages 222–235. SIAM, 2010.
- [18] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, STOC '03*, page 448–455, New York, NY, USA, 2003. Association for Computing Machinery.
- [19] Michael R. Fellows, Fedor V. Fomin, Daniel Lokshtanov, Elena Losievskaja, Frances A. Rosamond, and Saket Saurabh. Distortion is fixed parameter tractable. *ACM Trans. Comput. Theory*, 5(4):16:1–16:20, 2013.

- [20] Ronald L Graham and Peter M Winkler. On isometric embeddings of graphs. *Transactions of the American Mathematical Society*, 288(2):527–536, 1985.
- [21] Piotr Indyk and Jirí Matousek. Low-distortion embeddings of finite metric spaces. In Jacob E. Goodman and Joseph O’Rourke, editors, *Handbook of Discrete and Computational Geometry, Second Edition*, pages 177–196. Chapman and Hall/CRC, 2004.
- [22] William B. Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into hilbert space. *Contemporary mathematics*, 26:189–206, 1984.
- [23] Christiane Lammersen, Anastasios Sidiropoulos, and Christian Sohler. Streaming embeddings with slack. In Frank Dehne, Marina Gavrilova, Jörg-Rüdiger Sack, and Csaba D. Tóth, editors, *Algorithms and Data Structures*, pages 483–494, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
- [24] Nathan Linial, Eran London, and Yuri Rabinovich. The geometry of graphs and some of its algorithmic applications. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 577–591. IEEE Computer Society, 1994.
- [25] Konstantin Makarychev and Yury Makarychev. A union of euclidean metric spaces is euclidean. *Discrete Analysis*, 14, 02 2016.
- [26] Jirí Matousek and Anastasios Sidiropoulos. Inapproximability for metric embeddings into \mathbb{R}^d . In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 405–413. IEEE Computer Society, 2008.
- [27] Assaf Naor and Yuval Rabani. On lipschitz extension from finite subsets. *Israel Journal of Mathematics*, 219, 06 2015.
- [28] Amir Nayyeri and Benjamin Raichel. Reality distortion: Exact and approximate algorithms for embedding into the line. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 729–747. IEEE Computer Society, 2015.
- [29] Mikhail I. Ostrovskii and Beata Randrianantoanina. On ℓ_1 -embeddability of unions of ℓ_1 -embeddable metric spaces and of twisted unions of hypercubes. *Analysis and Geometry in Metric Spaces*, 10(1):313–329, 2022.
- [30] S. V. Shpectorov. On scale embeddings of graphs into hypercubes. *Eur. J. Comb.*, 14(2):117–130, March 1993.
- [31] Anastasios Sidiropoulos, Mihai Badoiu, Kedar Dhamdhere, Anupam Gupta, Piotr Indyk, Yuri Rabinovich, Harald Räcke, and R. Ravi. Approximation algorithms for low-distortion embeddings into low-dimensional spaces. *SIAM J. Discret. Math.*, 33(1):454–473, 2019.
- [32] Anastasios Sidiropoulos, Dingkang Wang, and Yusu Wang. Metric embeddings with outliers. In *ACM-SIAM Symposium on Discrete Algorithms*, 2015.

A Full constants for Sections 4 and 5

Lemma A.1 gives a version of Lemma 4.1 more generally in terms of τ and other flexible parameters. The proof would be identical to that given in Section 4.

LEMMA A.1. *Let $\alpha \leftarrow$ Algorithm 4.1($(X, d), S, \alpha_S, \tau$). Then we have the following bounds on the expansion for each pair $x, y \in X$:*

- (a) *If $x, y \in S$, then $d_Y(\alpha(x), \alpha(y)) \leq c_S \cdot d(x, y)$.*
- (b) *If $x \in S, y \in X \setminus S$, then $d_Y(\alpha(x), \alpha(y)) \leq (2\tau + 6)c_S \cdot d(x, y)$.*
- (c) *If $x, y \in X \setminus S$ and $d(x, \gamma(x)) \leq \kappa \cdot d(x, y)$ for positive κ and γ as defined in line 2 of the algorithm, then $d_Y(\alpha(x), \alpha(y)) \leq (2\kappa + 1)(2\tau + 6)c_S \cdot d(x, y)$.*

(d) If $x, y \in X \setminus S$ and $d(x, \gamma(x)), d(y, \gamma(y)) > \kappa \cdot d(x, y)$ for γ as defined in line 2 of the algorithm and real number $\kappa > 1$, then

$$E_{\alpha}[d_Y(\alpha(x), \alpha(y))] \leq \left(\frac{(\tau + 3)(2\tau + 6)(2\kappa + 1)}{\tau(\kappa - 1)} \right) \cdot c_S \cdot d(x, y).$$

Lemma [A.2](#) gives a version of Lemma [5.2](#) more generally in terms of τ and other flexible parameters. The proof would be identical to that given in Section [5](#).

LEMMA A.2. Let $\alpha \leftarrow$ Algorithm [5.1](#) $(X, d, S, \alpha_S, \alpha_X, \tau)$. Then we have the following bounds on the expansion for each pair $x, y \in X$:

(a) If $x, y \in S$, then $\|\alpha(x) - \alpha(y)\|_1 \leq c_S \cdot d(x, y)$.

(b) If $x, y \in K_i$, then $\|\alpha(x) - \alpha(y)\|_1 \leq c_X \cdot d(x, y)$.

(c) If $x \in S, y \in X \setminus S$, then $\|\alpha(x) - \alpha(y)\|_1 \leq [(\tau + 5)c_S + (2\tau + 5)c_X] \cdot d(x, y)$.

(d) If $x, y \in X \setminus S$ and $d(x, \gamma(x)) \leq \kappa \cdot d(x, y)$ for positive κ and γ as defined in line 2 of the algorithm, then

$$\|\alpha(x) - \alpha(y)\|_1 \leq [((2\tau + 8)\kappa + \tau + 5) \cdot c_S + ((4\tau + 10)\kappa + 2\tau + 5) \cdot c_X] \cdot d(x, y).$$

(e) If $x, y \in X \setminus S$ and $d(x, \gamma(x)), d(y, \gamma(y)) > \kappa \cdot d(x, y)$ for γ as defined in line 2 of the algorithm and real number $\kappa > 1$, then

$$\begin{aligned} E_{\alpha}[\|\alpha(x) - \alpha(y)\|_1] \\ \leq \left[\left[\frac{\tau + 3}{(\kappa - 1)\tau} \cdot H_k \cdot (\kappa(\tau + 3) + (\kappa + 1)(\tau + 5)) \right] c_S + \left[1 + \frac{\tau + 3}{(\kappa - 1)\tau} \cdot H_k \cdot (2\tau + 5)(2\kappa + 1) \right] c_X \right] \cdot d(x, y). \end{aligned}$$

B Hardness of finding outlier sets

Sidiropoulos *et al.* [\[32\]](#) showed that for any $t \geq 2$, it is NP-hard to determine the size of the smallest outlier set for a finite metric (X, d) such that the metric without the outlier set is isometrically embeddable into ℓ_2^t . Additionally, because they reduce from Vertex Cover, they show that under the Unique Games Conjecture it is NP-hard to approximate the size of such a set to a factor better than $2 - \epsilon$.

In this appendix, we will give an alternate proof of a similar conclusion, but we extend their result to show that it holds even if the input metric is an unweighted graph metric. We note that unlike the Sidiropoulos *et al.* proof, our proof does not apply for arbitrary choice of dimension d .

First, we claim the following Lemma [B.1](#) which we will prove later. Note that it is NP-hard to decide if a general metric is isometrically ℓ_1 -embeddable and thus it is hard to decide if the minimum outlier set for such an embedding has size 0 or size larger than 0, implying hardness of any approximation for this value. However, ℓ_1 -embeddability can be decided in polynomial time for unweighted graph metrics [\[30, 14, 15\]](#), and we show that even with this restriction on the input, it is hard to determine minimum outlier set size.

LEMMA B.1. Let (X, d) be the distance metric for an unweighted graph $G = (V, E)$. then, given (X, d, k) it is NP-hard to decide if there exists a subset $K \subseteq X$ with $|K| = k$ such that $(X \setminus K, d|_{X \setminus K})$ is isometrically embeddable into ℓ_1 , even when the input metric is an unweighted graph metric.

Under the unique games conjecture, it is NP-hard to find a $2 - \epsilon$ approximation for the minimum such k , for any $\epsilon > 0$.

We can now use this result to prove Theorem [2.10](#).

Proof of Theorem [2.10](#). We appeal to Lemma [B.1](#) to show that the theorem holds for $p = 1$ and here we show that it holds for $1 < p < \infty$.

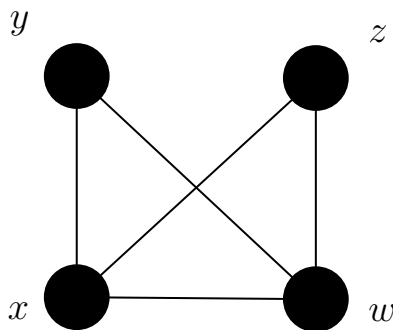


Figure 6: Example of a subgraph formed by nodes $\{u_1 = x, u_2 = y, v_1 = w, v_2 = z\}$ as defined in the proof of Theorem 2.10.

Consider a graph $G = (V, E)$. Given G , we give a polynomial time construction of an unweighted graph metric $(V', d_{G'})$ such that the size of the minimum outlier set for embedding the metric into ℓ_p is the same as the size of the minimum vertex cover on G . In this proof, minimum outlier set refers to the minimum outlier set for distortion 1 and we consider embeddings into ℓ_p for finite integer $p > 1$.

Construction: We will construct an unweighted graph $G' = (V', E')$ and let $d_{G'}$ be the distance metric on this graph. In particular, let $V' = \{u_1 | u \in V\} \cup \{u_2 | u \in V\}$. Add edges between every pair of nodes in the graph, but omit edges $u_2 v_2$ for $uv \in E$.

Correctness: Let K be a minimum outlier set on $(V', d_{G'})$ and let \hat{V} be a minimum vertex cover on G . We claim $|K| = |\hat{V}|$.

$|K| \leq |\hat{V}|$: We construct an outlier set of size at most $|\hat{V}|$. Define $K' := \{u_2 | u \in \hat{V}\}$. We claim that $(V' \setminus K', d|_{V' \setminus K'})$ is the equidistant metric with distance 1 between all points, which is always embeddable in ℓ_p for any p . Assume this is not the case. Note that all distances in G' are 1 or 2, so there exists a pair of nodes with distance 2 between them. The only such pairs are of the form $\{u_2, v_2\}$ for $uv \in E$. However, $u_2, v_2 \in V' \setminus K'$ implies that \hat{V} does not cover edge uv so it is not a vertex cover and we reach a contradiction. Thus since K is minimum, we get $|K| \leq |K'| \leq |\hat{V}|$.

$|\hat{V}| \leq |K|$: We construct a vertex cover of size at most $|K|$. Define $\hat{V}' := \{u | u_1 \in K \text{ or } u_2 \in K\}$. We see $|\hat{V}'| \leq |K|$ and we claim it is a vertex cover. Assume otherwise. Then there exists an edge uv that is not covered by \hat{V}' and there is a subgraph of the form in Figure 6 in the induced subgraph of G' on $V' \setminus K', G'[V' \setminus K']$. Note that distances in this subgraph are exactly distances in the entire graph by our construction. Thus we need only show that this subgraph is not isometrically embeddable in ℓ_p .

Note that ℓ_p is a strictly convex space for $1 < p < \infty$ [12]. Thus, for $a, b \in \mathbb{R}^t$ for any fixed t , if $\|a\|_p = \|b\|_p = 1$ and $a \neq b$, then $\|\frac{a+b}{2}\|_p < 1$, which implies $\|a+b\|_p < 2^{1/p}$. Let w, x, y, z be four points in ℓ_p such that the distance between all pairs of points is 1, except between y and z which are a distance 2 apart. Then let $a = y - x$ and $b = x - z$. We have $\|y - x\|_p = \|x - z\|_p = 1$, so $\|y - z\|_p < 2^{1/p}$ unless $y - x = x - z$. Since $2^{1/p} < 2$ for $p > 1$, we get that $y - x = x - z \implies x = \frac{y+z}{2}$. However, equivalent analysis on w implies $w = \frac{y+z}{2}$. This means that x and w are the same point and their distance is 0, not 1. Thus, this set of four points cannot exist in ℓ_p for $1 < p < \infty$ and the subgraph in Figure 6 is not isometrically embeddable into this space.

□

Proof of Lemma B.1. As in the previous lemma, we reduce the vertex cover problem to this problem. Consider a graph $G = (V, E)$. We give a polynomial time construction of an unweighted graph metric $(V', d_{G'})$ such that that size of the minimum outlier set for embedding the metric into ℓ_1 is the same as the size of the minimum vertex cover on G . In this proof, minimum outlier set refers to the minimum outlier set for distortion 1 and we consider embeddings into ℓ_1 .

Construction: We will construct an unweighted graph $G' = (V', E')$ and let $d_{G'}$ be the distance metric on this graph. In particular, let $V' = \{x_i, y_i, z_i, w_i | i \in V\}$. Add edges between every pair of nodes in the graph, but omit edges $x_i y_i$ for all i . Additionally, omit edges $x_i x_j$ if $ij \in E$.

Correctness: Let K be a minimum outlier set on $(V', d_{G'})$ and let \hat{V} be a minimum vertex cover on G . We claim $|K| = |\hat{V}|$.

$|K| \leq |\hat{V}|$: Define $K' := \{x_i | i \in \hat{V}\}$. We claim that $(V' \setminus K', d_{G'}|_{V' \setminus K'})$ is ℓ_1 embeddable. In particular, [30, 14], and [15] show that if an unweighted graph is such that each node has at most one other node it does not have an edge to (i.e. if the graph is a subgraph of a cocktail party graph), then the graph is ℓ_1 -embeddable. Note that $d_{G'}$ restricted to the nodes in $V' \setminus K'$ is in fact the distance metric on the induced subgraph of G' on those same nodes, $G'[V' \setminus K']$. This is because we only removed some of the x_i , which cannot affect distances between any remaining pairs.

Thus, we are left with showing that $G'[V' \setminus K']$ is such that each node has at most one other node to which it does not have an edge. Assume otherwise. Then there exists $a, b, c \in V' \setminus K'$ such that a does not have an edge to b or to c . The only nodes in G' that are missing an edge to more than one other node in G' are some of the x_i , so a must be x_i for some i . Additionally, the nodes that x_i does not have an edge to are y_i and all x_j such that $ij \in E$. Thus, at least one of b and c must be x_j for some j such that $ij \in E$. However, this implies $x_i, x_j \notin K' \implies i, j \notin \hat{V}$, so \hat{V} is not a vertex cover and we reach a contradiction. Thus K' is an outlier set and we get $|K| \leq |K'| = |\hat{V}|$

$|\hat{V}| \leq |K|$: Define $\hat{V}' := \{i | x_i, y_i, z_i, \text{ or } w_i \in K\}$. We have $|\hat{V}'| \leq |K|$, so if \hat{V}' is a valid vertex cover then we obtain the desired bound. Assume \hat{V}' is not a vertex cover. Then there exists $u_i u_j \in E$ such that $\{x_i, y_i, z_i, w_i, x_j, y_j, z_j, w_j\} \subseteq V' \setminus K$. These nodes form the subgraph pictured in Figure 7, and $d_{G'}$ on this subset of nodes has the same value as the graph metric on this subgraph. Thus, we need only show that the subgraph in Figure 7, which we call G'' , is not ℓ_1 -embeddable.

By [15], an unweighted graph is ℓ_1 -embeddable if and only if there exists an integer $t \in \mathbb{Z}^+$ such that the same graph with all edge weights set to t is hypercube embeddable. Deza and Shpectorov [30, 14] show that if an unweighted graph is ℓ_1 -embeddable and it is not “reducible” as defined by [20], then it must be an isometric subgraph of a cocktail party graph or a half-cube (a type of graph that is hypercube embeddable at scale 2). In Lemma B.2 we use Graham and Winkler’s [20] techniques to show that G'' is not reducible.

Additionally, x_i lacks neighbors y_i and x_j , so this is not a subgraph of a cocktail party graph, which is a graph in which each node is a neighbor of all but one node. This leaves us with showing that G'' is not a subgraph of a half-cube, which we prove by showing that it is not hypercube embeddable at scale 2 in Lemma B.3. Thus, we conclude G'' cannot be a subgraph of $G'[V' \setminus K]$ and \hat{V}' must be a vertex cover.

□

LEMMA B.2. *The graph appearing in Figure 7 is not reducible, where reducible is as defined by [20]*

Proof. Graham and Winkler show that a graph is not reducible if all of its edges are in the same equivalence class of the equivalence relation $\hat{\theta}$. $\hat{\theta}$ is defined to be the transitive closure of θ , which is defined on the edges of a graph as follows:

For a graph $G'' = (V'', E'')$, edges $ab, cd \in E''$ are related by θ if and only if

$$[d_{G''}(a, c) - d_{G''}(a, d)] - [d_{G''}(b, c) - d_{G''}(b, d)] \neq 0.$$

We will show that all edges of G'' are in the same equivalence class of $\hat{\theta}$.

First, notice that in the big/main clique of $y_i, y_j, z_i, z_j, w_i, w_j$ in G'' , all edges must be related by $\hat{\theta}$. Take two adjacent edges ab, bc . Since all distances in the clique are 1, we get $[d_{G''}(a, b) - d_{G''}(a, c)] - [d_{G''}(b, b) - d_{G''}(b, c)] = 1 \neq 0$. Thus all adjacent edges in the clique are related by θ and thus all edges in the clique are related by $\hat{\theta}$. This just leaves us to consider the edges to x_i and x_j . We see that $z_i x_i \theta z_i w_i$ because again we have adjacent

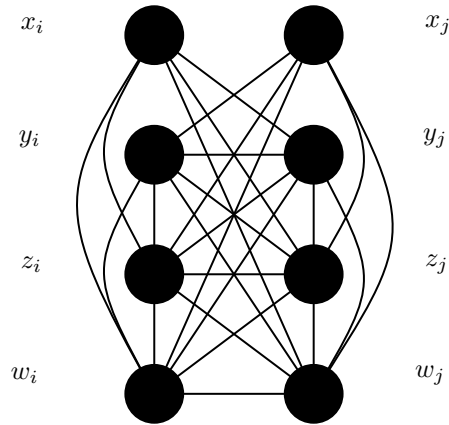


Figure 7: Example of a subgraph formed by nodes $\{x_i, y_i, z_i, w_i, x_j, y_j, z_j, w_j\}$ as defined in the proof of Lemma [B.1](#) where $ij \in E$.

edges with distances between all three vertices being 1. The same analysis goes for all the other edges to x_i and to x_j since they are part of forming a smaller clique with x_i or x_j and two members of the main clique. This means they are all in the same equivalence class as the edges in the main clique and thus all edges are in the same equivalence class, meaning it is irreducible. \square

LEMMA B.3. *The graph appearing in Figure [7](#) is not hypercube embeddable, and the same graph with all edge weights scaled to 2 is also not hypercube embeddable.*

Proof. The graph is not bipartite, so it is not hypercube embeddable at scale 1 [\[16\]](#).

For the second part of this proof, we need only verify that the graph is not an isometric subgraph of a halved cube. To do this, we could run the algorithm of Deza and Shpectorov [\[14\]](#) for doing so. However, we will present the proof in a different way because their algorithm is more generalized than what is needed for our specific purposes.

To begin the second part of this part of the proof, we first consider an alternative view of a hypercube embedding. In particular, if we have a hypercube embedding in dimension t for a metric space on n nodes, we can write an $n \times t$ matrix in which each row of the matrix is the binary string associated with a particular node. The number of columns where two rows differ is then the distance between the corresponding nodes in the graph. Notice that this matrix defines a hypercube embedding and a hypercube embedding defines this matrix. Additionally, we can remove any columns in which all rows have the same value without affecting the “distance” between rows. Additionally, if we swap the order of the columns or if we pick a column of this matrix and flip all bits in that column, it has no effect on the distance between the rows. Thus, we can always assume that the top row of the matrix for a hypercube embedding is made up of all 0s, as such a choice of embedding must exist if some hypercube embedding does.

Let’s begin constructing a hypercube embedding for the graph G'' . In particular, we define a matrix in which the top row corresponds to the node $x_i \in V(G'')$ and the row consists of all 0s (which we’ve already argued is a fine assumption). Then we know that all rows except those corresponding to y_i and x_j must have exactly two 1s in their rows, and y_i and x_j have exactly four 1s in their rows (in order to make their distance from x_i correct). We will assume that all the columns in the matrix we construct have at least two distinct values in each column, as we have argued such an embedding must exist if any embedding exists since we can delete columns where all values are equal. Thus, we can assume that M has at most 18 columns since there are at most eighteen 1s in the entire matrix. This leads us to the following partial embedding where most values are unassigned so far.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
x_i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_i																		
y_i																		
x_j																		
w_i																		
w_j																		
z_j																		
y_j																		

We will next assume that the two 1s in z_i 's row are in the first two columns. (As we mentioned, we can always rearrange the columns of a hypercube embedding matrix to make one that looks this way since those columns at this point are indistinguishable.) This gives us the following matrix:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
x_i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_i	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y_i																		
x_j																		
w_i																		
w_j																		
z_j																		
y_j																		

Next, consider node y_i that is a distance 2 closer to z_i than to x_i (in the scaled graph, as it is a distance 1 closer in the original graph). Notice that in all columns after the first two columns, if y_i doesn't match x_i , then it also doesn't match z_i . Thus, the only place where it can have these two differences with x_i that it does not have those differences with z_i is the first two columns. Thus, y_i must have 1s in the first two columns. The same goes for x_j because it is also a distance 2 closer to z_i than to x_i . For all the other nodes, they are equidistant from x_i and z_i . Thus, because they must have the same number of differences with x_i 's and z_i 's rows after the first two columns, they must also have the same number of differences in the first two columns. This means they either have 10 or 01 in the first two columns. Because we don't know which of these to put in those columns, for now we don't fill those values in yet. Because y_i and x_j are a distance 4 from x_i , they must have four 1s total. We will assume that the first four 1s for y_i are in the first four columns, because we can rearrange any matrix in which the other two 1s are in later columns such that they are in the second two columns. This leaves us with the following matrix constructed so far.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
x_i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_i	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y_i	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_j	1	1																
w_i																		
w_j																		
z_j																		
y_j																		

Now we can consider the distance between x_j and y_i . They are a distance 2 apart in the scaled graph, and we only have two more 1s that we can place in x_j 's row. If we put both of these 1s in positions 3 and 4, then there are no differences between x_j and y_i , and if we put none of these 1s in positions 3 or 4, the two have a distance 4 apart. Thus, we must have exactly a single 1 in columns 3 and 4. We will assume that this is in column 3, as at the moment these columns are indistinguishable so if a good embedding has the opposite assignment, we can swap the columns to get something consistent with this embedding. We will also assume that the last 1 is in column 5 since the columns after 4 are indistinguishable at this point.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
x_i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_i	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y_i	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_j	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
w_i																		
w_j																		
z_j																		
y_j																		

Now we can consider labeling w_i, w_j, z_j . We already established that due to the fact they are equidistant from x_i and z_i , they must all have either 10 or 01 in the first two columns. Then they have a single 1 left for the rest of the columns since they are a distance 2 from x_i . We notice that if we put this second 1 in column 4, then the distance between the node and x_j is 4 (one difference in the first two columns and one each in each of columns 3 through 5), when it should be 2. Analogously, if the 1 is in column 5, the distance to y_i is 4 instead of 2. If the 1 is in column 6 or larger, the distance to y_i and x_j is 4 instead of 2. Thus, this leaves us with putting the 1 in column 3 for all three of these nodes. Everything after this must be 0s since we used up our only other 1 in the first two columns. This gives us the following partial embedding.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
x_i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_i	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y_i	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_j	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
w_i			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w_j			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z_j			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y_j																		

Now we notice that w_i, w_j, z_j have no differences after column 3. Thus, all of their differences must be in the first two columns. This means that we must come up with three length 2 binary strings that are all Hamming distance 2 from each other, which is impossible. Because all of the decisions we made in constructing this embedding were necessary, this partial construction is required which means it's impossible to construct a hypercube embedding for this graph at scale 2. \square