MaxSketch: Robust Distinct Counting in Streams via Random Projections

Nikos Tsikouras[1,2] **Constantine Caramanis**[2,3] **Christos Tzamos**[1,2]
[1] National and Kapodistrian University of Athens, Greece
[2] Archimedes, Athena Research Center, Greece
[3] The University of Texas at Austin, USA

Abstract

Estimating the number of distinct elements in a data stream is well understood when repeated elements are identical. In modern settings, however, observations are high-dimensional and noisy, so repeated instances of the same object are only approximately similar – for example, different images of the same individual may vary significantly at the pixel level. Classical sketches such as HyperLogLog rely on consistent hash values for identical elements and break down in this regime. Recent work on robust distinct counting in general metric spaces gives algorithms that use $\tilde{O}(\sqrt{n}/\varepsilon)$ words of memory and proves an $\Omega(\sqrt{n})$-bit lower bound for general noisy datasets. We show that substantially improved memory guarantees are possible under geometric structure common in learned representations. We introduce MaxSketch, a simple max-linear sketch built from random Gaussian projections, and prove that it succeeds in estimating the number of distinct latent objects. Concretely, we show that under this geometric structure $m = O(\varepsilon^{-2}\log(2/\delta))$ independent Gaussian projections suffice to estimate the true distinct count within a factor of $1+\varepsilon$ with probability at least $1-\delta$. Experiments on image streams confirm that MaxSketch accurately estimates distinct counts and generalizes beyond the training regime. Our results bridge classical streaming algorithms and modern representation learning, showing how geometric structure can fundamentally reduce the complexity of distinct counting.

1 Introduction

Counting the number of distinct elements in a data stream is a classical algorithmic problem with deep connections to randomized algorithms, information theory, and large-scale data analysis. In its canonical form, the task is to estimate the number of unique keys in a stream using sublinear memory and a single pass over the data. A long line of work has shown that this is possible with striking efficiency: algorithms such as [5, 6, 7] achieve accurate estimates using only logarithmic memory by maintaining carefully designed randomized sketches of the stream.

These classical results, however, rely on a crucial assumption: the same element is represented identically every time it appears. In modern data modalities, this assumption is fundamentally violated. To see this, consider the task of counting the number of distinct people passing through a busy intersection observed by a camera. Suppose Alice and Bob each pass through once, and later Alice passes through again. Although it is the same underlying individual, her second appearance may differ substantially at the pixel level, due to changes in lighting, viewpoint, or pose. As a result, the two observations of Alice may be nearly as different from each other as they are from observations of entirely different individuals.

A natural approach is to embed each photo into a continuous feature space and apply distinct counting in that representation. However, this does not resolve the issue: repeated observations of the same underlying object are no longer identical, causing sketching methods to overcount, while clustering-

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

based approaches require maintaining a number of representatives that grows with the number of distinct elements, undermining the sublinear memory guarantees of streaming algorithms.

A recent line of work [2, 3, 29] has studied this *robust* distinct counting problem, where elements are considered equal if they are sufficiently close in some metric. While these results give algorithms with non-trivial guarantees, they also reveal that the problem is fundamentally harder than its classical counterpart: [29] establishes that, on general noisy datasets, any streaming algorithm that approximates the distinct count within a factor of 1.1 requires $\Omega(\sqrt{n})$ bits of memory, exponentially far from the rates achievable under exact key equality [15].

Modern representation learning, however, often induces a more useful geometric structure than the distance-based separation studied above: observations of the same object form tight clusters in embedding space, while different objects correspond to well-separated directions.

This raises a central question:

Closeness of representations in an arbitrary metric space is not strong enough to allow robust distinct counting with memory independent of n ($\Omega(\sqrt{n})$ lower bound in [29]). Does clustered geometry with intra-class closeness and inter-class separation bypass this barrier?

This question sits at the intersection of classical streaming algorithms and modern representation learning. On the one hand, the theory of distinct counting demonstrates that extreme-value statistics of random hash functions contain sufficient information about the number of unique elements [6, 15]. On the other hand, permutation-invariant architectures such as DeepSets [27] can in principle aggregate set-structured data, but provide no theoretical guarantees on recovering global combinatorial properties like distinct counts. In this work, we bridge this gap with a deliberately minimal max-linear sketch that is both analytically tractable and provably correct.

1.1 A geometric model for distinct objects

We assume that each distinct object corresponds to an unknown latent direction in a high-dimensional space, and that each observation is a noisy but well-aligned version of its object's direction. Different objects occupy well-separated directions, with low inner products between distinct centers. This picture is motivated by neural collapse: at the end of training, deep classifiers map the training examples of each class close to their class mean, and the class means become nearly orthogonal [19, 30]. Face-recognition embeddings are trained to give the same structure for identities not seen during training [21].

Under this model, the number of distinct objects corresponds to the number of well-separated directions generating the stream. The challenge is to recover this number without explicitly clustering and without storing the embeddings.

The $\Omega(\sqrt{n})$ lower bound of [29] uses adversarial datasets that are not "well-shaped": points of different objects can be as close as near-duplicates of the same object. Our clusterability assumption (Definition 2.1) is not chosen to sidestep this bound; it is motivated by the geometry of learned representations. Clusterable streams fall in the well-shaped class, for which [29] explicitly leaves open whether $\Omega(\sqrt{n})$ memory is necessary. We show that it is not necessary under the stronger inner-product structure of clusterability. This structure makes Gaussian extreme-value statistics separate different cardinalities, and $O(\varepsilon^{-2}\log(2/\delta))$ Gaussian projections suffice.

1.2 MaxSketch: a max-linear analogue of classical sketches

Our key observation is that max aggregation discards multiplicity by construction: once a cluster has contributed a large projection value, additional repetitions of the same object leave the sketch unchanged. This contrasts sharply with sum or mean aggregation, which scale with frequency and therefore confuse distinctness with multiplicity. Given random Gaussian vectors $w_1, \ldots, w_m \in \mathbb{R}^d$, we compute

$$M_j(X) = \max_{1 \le i \le n} \langle w_j, x_i \rangle$$

where $x_1, \ldots, x_n \in \mathbb{R}^d$ are observations generated from k underlying objects, and average these maxima across projections. The expectation of this statistic grows monotonically with the number

of distinct latent directions, which is the signal we want to recover. This operation is permutation-invariant, streaming-compatible, and stores only m scalars per stream.

We call this representation MaxSketch. Unlike classical sketches such as HyperLogLog, MaxSketch operates directly on continuous embeddings and tolerates noise and approximate equality. Unlike general-purpose neural set models, MaxSketch is deliberately minimal and analytically tractable.

1.3 Theoretical guarantees

Our main result shows that MaxSketch provably recovers the number of distinct objects up to multiplicative error.

Informal Theorem 4.4 (Robust Distinct Count Estimation). Under a natural clusterability assumption, $m = O(\varepsilon^{-2} \log(2/\delta))$ random projections suffice to estimate the number of distinct objects in a stream of length n within a $(1 + \varepsilon)$-multiplicative error, with probability at least $1 - \delta$.

Compared to the $\Omega(\sqrt{n})$ memory required in general metric spaces [29], the improvement is enabled by separation between cluster centers, a property missing from the worst-case instances underlying that bound.

Our analysis shows that, despite correlations induced by approximate cluster structure, the expected projection maximum is close to the expected maximum of independent Gaussians. We prove that this expectation rises by a detectable amount when the distinct count grows by a factor of $1 + \varepsilon$, and use superconcentration to control fluctuations around that expectation. These ingredients yield the stated projection bound.

This is a key distinction with other methods. We view MaxSketch as a statistics-preserving embedding: rather than preserving distances or geometry, it maps a set of observations to a single number, $S(X)$, whose expectation grows with the number of distinct objects. This has two consequences. When the assumptions hold only approximately, $S(X)$ remains informative, as illustrated empirically in the face counting experiments of Section 5.3. And once the projections are fixed, learning to count reduces to fitting a one-dimensional monotone function of $S(X)$, which needs only polynomially many labelled streams (Theorem 4.8).

1.4 End-to-end learning from count-only supervision

The theoretical guarantees are stated for fixed Gaussian projections, but MaxSketch slots naturally into an end-to-end pipeline: an encoder maps raw inputs to embeddings, MaxSketch aggregates them, and a learned readout produces the count. Our theory does not assume that projections or embeddings are learned optimally; it establishes only that a compact sketch *exists* whenever the embeddings satisfy the geometric assumption. We examine this empirically in Section 5: on MNIST, end-to-end training under count-only supervision induces well-clustered embeddings; on CIFAR-10, where intra-class variability is much higher, MaxSketch counts almost exactly on top of a frozen self-supervised encoder.

1.5 Contributions

- **MaxSketch**, a simple max-linear, permutation-invariant sketch inspired by classical extreme-value methods. We prove that under a clusterability condition, $m = O(\varepsilon^{-2} \log(2/\delta))$ Gaussian projections give a $(1 + \varepsilon)$-multiplicative approximation with probability at least $1 - \delta$. By contrast, the best known algorithm for well-shaped data uses memory growing roughly like $\sqrt{n}$, and on general noisy datasets $\Omega(\sqrt{n})$ bits are necessary [29].

- **A learning reduction** showing that, once projections are fixed, count prediction reduces to fitting a one-dimensional monotone function of a fixed scalar statistic.

- **Empirical validation** on MNIST, CIFAR-10, and CelebA face streams, demonstrating extrapolation to sequences up to ten times longer than those seen in training, and accurate counting of thousands of unseen identities on faces.

1.6 Related Work

Distinct counting and sketching. Estimating the number of distinct elements in a data stream is a classical problem in algorithms, with celebrated solutions such as Flajolet–Martin, LogLog, and HyperLogLog achieving near-optimal accuracy using logarithmic memory via randomized hashing and extreme-value statistics [5, 6, 7, 15]. These methods crucially rely on exact key equality: repeated occurrences of the same element must hash to the same value. A natural idea in continuous settings is to first apply locality-sensitive hashing (LSH) [13, 14] and then run HyperLogLog on the resulting hash codes. However, this approach fundamentally fails for distinct counting: LSH guarantees collision for some near neighbors, not all repetitions of the same object, and its collision probability depends on distance in a way that entangles object identity with multiplicity. As a result, LSH-based preprocessing introduces bias that cannot be corrected by classical sketches.

Robust distinct counting under near-duplicates. A line of work initiated by [2, 3] studies robust distinct counting under near-duplicates, including well-shaped datasets in low-dimensional Euclidean spaces and, more recently, general metric spaces. Zhang [29] gives algorithms for general metric spaces that use $O(\varepsilon^{-1}\sqrt{n\log(1/\varepsilon)})$ words on well-shaped datasets and $\tilde{O}(\sqrt{n}/\varepsilon)$ words on general noisy datasets. He also proves that $\Omega(\sqrt{n})$ bits are necessary for a 1.1-approximation on general noisy datasets, and leaves open whether this holds for well-shaped data. We compare with these bounds in the word model of [29], in which one word stores one element of the space, here a vector in $\mathbb{R}^d$; we likewise count a real number as one word. In this model MaxSketch uses $O(m)$ words: the m real numbers and the m projection vectors, which are drawn once, before the stream, and do not depend on it.

MaxSketch addresses a different, more structured regime. The stream is embedded in a Euclidean space, observations are tightly aligned with latent centers, and distinct centers have small inner products. This inner-product structure is stronger than a metric distance-separation condition: it allows Gaussian projection maxima to behave as extreme-value statistics whose expectations separate different cardinalities. Under this clusterability assumption, MaxSketch uses $O(\varepsilon^{-2}\log(2/\delta))$ Gaussian projections, showing that substantially smaller sketches are possible in structured learned representations than in the worst-case general-metric setting.

2 Problem Setting

We observe a stream of unit vectors $X = (\tilde{x}_1, \ldots, \tilde{x}_n) \subset \mathbb{S}^{d-1}$, arriving in arbitrary order. Each $\tilde{x}_i$ is a noisy observation of one of $k^\star$ unknown latent objects; the same object may appear multiple times, with each appearance producing a different vector. Our goal is to estimate $k^\star$ in a single pass using memory sublinear in n, with supervision (when available) limited to the total count.

This setting differs from classical distinct-elements counting in that exact equality between repeated occurrences is replaced by approximate geometric similarity. To make the problem well-posed, we assume the stream has the following structure.

Definition 2.1 ((η,ρ)-clusterable sequence)**.** Fix $\eta \in (0,1)$ and $\rho \in [0,1)$. A sequence $X = (\tilde{x}_1, \ldots, \tilde{x}_n)$ of unit vectors in $\mathbb{R}^d$ is *(η,ρ)-clusterable* with $k^\star$ centers if there exist distinct unit vectors $x^{(1)}, \ldots, x^{(k^\star)} \in \mathbb{R}^d$ and a surjective assignment $r : [n] \to [k^\star]$ such that:

1. (**Within-cluster alignment**): $\langle \tilde{x}_i, x^{(r(i))} \rangle \geq 1 - \eta/2 \quad$ for all i.

2. (**Cross-cluster separation**): $|\langle x^{(r)}, x^{(s)} \rangle| \leq \rho \quad$ for all $r \neq s$.

Throughout, $\tilde{x}_i$ denotes an observed point and $x^{(r)}$ a cluster center; when no clusterability is assumed (e.g. in Definition 3.1) we write a generic stream as $x_1, \ldots, x_n$.

The first condition says each observation is directionally close to its center (capturing noise, augmentation, or pose variation); the second says distinct centers are well-separated, with ρ controlling approximate independence under Gaussian projection. In particular, when η is small, clusterable streams are well-shaped in the sense of [29].[1] This assumption formalizes a structure widely observed

[1] Specifically, when $\eta < (1-\rho)/18$, a condition much weaker than the one our main result assumes. Take $\alpha = 2\sqrt{\eta}$ in the definition of [29]. Since $\|\tilde{x}_i - x^{(r(i))}\|^2 = 2 - 2\langle \tilde{x}_i, x^{(r(i))} \rangle \leq \eta$, observations of the same object are within distance α of each other, while observations of different objects are at least $\sqrt{2-2\rho} - 2\sqrt{\eta} > 2\alpha$ apart.

in learned representations. When training embeddings for classification, or other self-supervised objectives, the learned space typically exhibits clustering by semantic identity: embeddings of the same object (or class) cluster tightly, while distinct objects occupy well-separated directions. Recent work on neural collapse [10, 19, 30, 31] formalizes this phenomenon in deep networks. Our results do not require learning this geometry, only that it exists.

3 MaxSketch: A Max-Linear Streaming Representation

We now describe a simple sketch that exploits this geometry.

Definition 3.1 (**MaxSketch**). Fix an integer m and let $w_1, \dots, w_m \sim N(0, I_d)$ be independent random vectors, fixed once and for all. Given a stream $X = (x_1, \dots x_n)$, define

$$M_j(X) = \max_{1 \le i \le n} \langle w_j, x_i \rangle, \quad S(X) = \frac{1}{m} \sum_{j=1}^{m} M_j(X).$$

Sketch size. As is common in the sketching literature, we measure the size of MaxSketch by its dimension m, the number of Gaussian projections. The sketch maps a stream X to the vector $(M_1(X), \dots, M_m(X)) \in \mathbb{R}^m$ and updates it in a single pass. The projection vectors are drawn once and reused for every stream; only the m values M_j depend on the stream.

Why max aggregation? Max aggregation discards multiplicity information by construction: once a cluster has contributed a large projection value, additional repetitions do not change the sketch. This sharply contrasts with sum or mean aggregation, which scale with frequency and therefore confuse distinctness with multiplicity. As we show, under clusterability, maxima of random projections isolate extreme-value statistics that depend primarily on the number of distinct latent directions.

4 Theoretical Guarantees

We show that MaxSketch estimates the number of distinct clusters under (η, ρ)-clusterability. The analysis has three steps: (i) characterize the extreme-value behavior of Gaussian projections onto well-separated centers, (ii) bound the perturbation introduced by within-cluster noise, and (iii) combine an expectation gap with concentration to distinguish different cluster counts using $O\left(\varepsilon^{-2} \log(2/\delta)\right)$ projections.

4.1 Gaussian maxima over cluster centers

Fix unit vectors $x^{(1)}, \dots, x^{(k)} \in \mathbb{R}^d$ and $w \sim \mathcal{N}(0, I_d)$, and let $G_r := \langle w, x^{(r)} \rangle$. Then $G = (G_1, \dots, G_k)$ is jointly Gaussian with mean zero, unit variances, and $\mathrm{Cov}(G_r, G_s) = \langle x^{(r)}, x^{(s)} \rangle$. We control $\mathbb{E}[\max_r G_r]$ when the centers are well-separated by comparing to the i.i.d. case, where extreme-value asymptotics are well understood.

Lemma 4.1 (Gaussian maxima comparison). *Let $(G_1, \dots, G_k)$ be a centered Gaussian vector with $\mathrm{Var}(G_r) = 1$ and $|\mathrm{Cov}(G_r, G_s)| \le \rho$ for $r \ne s$, where $\rho \in [0, 1)$. Let $Z_1, \dots, Z_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$. Then*

$$\sqrt{1-\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r\Big] \;\le\; \mathbb{E}\Big[\max_{r \le k} G_r\Big] \;\le\; \sqrt{1+\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r\Big].$$

4.2 Robustness to within-cluster perturbations

Maxima over observed points differ from maxima over centers by an amount controlled by the within-cluster slack η.

Lemma 4.2 (Perturbation bound). *Let $\widetilde{X} = (\tilde{x}_1, \dots, \tilde{x}_n) \subset \mathbb{S}^{d-1}$ satisfy the within-cluster alignment condition of Definition 2.1 with centers $x^{(1)}, \dots, x^{(k)}$ and assignment r. Let $w \sim \mathcal{N}(0, I_d)$. Then*

$$\Big| \mathbb{E}\Big[\max_{i \le n} \langle w, \tilde{x}_i \rangle\Big] - \mathbb{E}\Big[\max_{r \le k} \langle w, x^{(r)} \rangle\Big] \Big| \;\le\; \sqrt{2\eta \log(2n)}.$$

4.3 Separating different numbers of clusters

To estimate the number of clusters, we must distinguish between k and $(1+\varepsilon)k$ centers via their expected maxima.

Lemma 4.3 (Gap in Gaussian maxima)**.** *Let $k \geq 2$, $\varepsilon \in (0,1]$, and $k' = \lceil (1+\varepsilon)k \rceil$. Let $x^{(1)}, \ldots, x^{(k')} \in \mathbb{R}^d$ be unit vectors with $|\langle x^{(i)}, x^{(j)} \rangle| \leq \rho$ for $i \neq j$, where $\rho \leq c_\rho \varepsilon / \log k$ for a sufficiently small absolute constant $c_\rho > 0$, and let $w \sim \mathcal{N}(0, I_d)$. Define*

$$\Delta_{\text{centers}}(k) := \mathbb{E}\left[\max_{1 \leq i \leq k'} \langle w, x^{(i)} \rangle\right] - \mathbb{E}\left[\max_{1 \leq i \leq k} \langle w, x^{(i)} \rangle\right].$$

Then $\Delta_{\text{centers}}(k) = \Omega(\varepsilon / \sqrt{\log k})$ for all $k \geq 2$.

Each lemma isolates one ingredient: Lemma 4.1 controls the effect of correlation between centers through ρ, Lemma 4.2 controls the within-cluster noise through η, and Lemma 4.3 gives us the signal that distinguishes k from $(1+\varepsilon)k$ centers. Theorem 4.4 combines them with a concentration bound for $S(\widetilde{X})$. All three lemmas are proved in Appendix B.1.

4.4 Main counting theorem

Theorem 4.4 (Estimating the number of clusters)**.** *Fix $n \geq 2$, $\varepsilon \in (0, 1/2]$, and $\delta \in (0,1)$. Let $\widetilde{X} = (\widetilde{x}_1, \ldots, \widetilde{x}_n) \subseteq \mathbb{S}^{d-1}$ be (η, ρ)-clusterable with $2 \leq k^\star \leq n$ centers, where $\rho \leq c_\rho \frac{\varepsilon}{\log n}, \eta \leq c_\eta \frac{\varepsilon^2}{(\log n)^2}$, for sufficiently small absolute constants $c_\rho, c_\eta > 0$.*

Draw $w_1, \ldots, w_m \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d), m \geq C \frac{\log(2/\delta)}{\varepsilon^2}$, for a sufficiently large absolute constant C.

Then there exists an explicit nondecreasing estimator, that uses only the MaxSketch statistic of Definition 3.1,

$$S(\widetilde{X}) = \frac{1}{m} \sum_{j=1}^{m} \max_{1 \leq i \leq n} \langle w_j, \widetilde{x}_i \rangle,$$

that outputs $\widehat{k}$ satisfying

$$k^\star \leq \widehat{k} \leq (1+\varepsilon)k^\star$$

with probability at least $1 - \delta$ over the Gaussian projections.

Remark 4.5 (Estimator). The estimator uses a geometric grid of $O(\log n / \varepsilon)$ candidate output values. Associated with each grid point is a deterministic threshold $\theta_0 \leq \theta_1 \leq \cdots$. Given the observed value $S(\widetilde{X})$, the estimator returns the output associated with the first threshold satisfying $S(\widetilde{X}) \leq \theta_r$. If no threshold satisfies this inequality, it returns n. Thus the estimator is a deterministic nondecreasing function of the scalar statistic $S(\widetilde{X})$. The thresholds depend only on n and ε.

Two features of Theorem 4.4 are worth noting. First, the number of projections depends only on ε and δ: it does not grow with the stream length n, the number of objects $k^\star$, or the dimension d. The stream length enters only through the geometric conditions, which become stricter as n grows. Second, the estimator reads a single number, $S(\widetilde{X})$, and is a fixed nondecreasing function of it (Remark 4.5). The proof is in Appendix B.2.

Can a better use of the registers reduce the number of projections? The next proposition shows that, as far as the dependence on ε and δ is concerned, it cannot.

Proposition 4.6 (Gaussian-max lower bound)**.** *Fix $\varepsilon \in (0, 1/4]$ and $\delta \in (0, 1/8]$. Let $k^\star$ be an unknown integer. Suppose an estimator observes only m independent samples from the distribution of $M_{k^\star} = \max_{1 \leq r \leq k^\star} Z_r$, where $Z_r \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$, and must output $\widehat{k}$ with $k^\star \leq \widehat{k} \leq (1+\varepsilon)k^\star$ with probability at least $1 - \delta$ for every $k^\star \geq \lceil 1/\varepsilon \rceil$. Then $m = \Omega(\varepsilon^{-2} \log(1/\delta))$.*

Consider a noiseless stream whose $k^\star$ centers are orthogonal. It satisfies the conditions of Theorem 4.4, and each MaxSketch register is the maximum of $k^\star$ independent standard Gaussians. Proposition 4.6 therefore says that no estimator working from the register values alone, ours included, can do better than Theorem 4.4 in its dependence on ε and δ, except by constant factors. The proposition says nothing about estimators that also see the projection vectors w_j, or about other streaming algorithms. The proof is in Appendix B.5.

4.5 From random to fixed weights, and learning the readout

Theorem 4.4 provides guarantees over a fresh draw of projections for every clusterable input. In practice, however, the sketch is built once and deployed across many sequences from some unknown distribution $\mathcal{D}$. Corollary 4.7 shows that a single fixed sketch suffices for most inputs from $\mathcal{D}$.

Corollary 4.7 (Fixed sketch for a distribution)**.** *Under the conditions of Theorem 4.4, let $\mathcal{D}$ be any distribution over (η, ρ)-clusterable sequences of length at most n with at least two centers. Then there exist fixed $w_1, \ldots, w_m$ such that MaxSketch achieves a $(1+\varepsilon)$-multiplicative approximation with probability at least $1-\delta$ over a fresh draw $X \sim \mathcal{D}$.*

Once the projections $W = (w_1, \ldots, w_m)$ are fixed, each sequence is represented by the scalar statistic $S_W(X)$. Theorem 4.4 constructs a deterministic threshold readout whose thresholds do not depend on X or on the realized W. For each fixed clusterable sequence X of length n, the readout returns a $(1+\varepsilon)$-approximation with probability at least $1-\delta$ over the initial draw of W. Corollary B.10 (proved in Appendix B.3) extends this pointwise guarantee to sequences of length at most n with at least two centers, and states explicitly that the readout has only $O(\log n/\varepsilon)$ possible outputs. We next show that, for sequences drawn from an unknown distribution, a monotone readout can be learned efficiently from count-labeled examples.

Theorem 4.8 (Learnability)**.** *Under the conditions of Theorem 4.4, let $\mathcal{D}$ be any distribution over (η, ρ)-clusterable sequences of length at most n with at least two centers. There exists a polynomial-time learning algorithm which, given $N = \mathrm{poly}(\log n, 1/\varepsilon, 1/\delta)$ i.i.d. training samples $(X_1, k_1), \ldots, (X_N, k_N) \sim \mathcal{D}$, outputs a monotone readout function $f : \mathbb{R} \to \mathbb{R}$ such that, for an independent test sample $(X, k) \sim \mathcal{D}$, $k \le f(S_W(X)) \le (1+\varepsilon)k$ with probability at least $1-\delta$ over the sketch W, the training sample and the test sample.*

Proofs of Corollary 4.7 and Theorem 4.8 are in Appendices B.3–B.4.

Together, Theorem 4.4 and Theorem 4.8 give a complete pipeline: a fixed Gaussian sketch (good with high probability per Theorem 4.4) followed by a monotone readout (efficiently fittable per Theorem 4.8). The next section examines whether this signal is useful on real embeddings, whether learned end-to-end from count supervision or supplied by a pretrained encoder.

5 Experiments

Our experiments have three goals. First, on MNIST we ask whether an encoder and a MaxSketch readout can be trained jointly from count-only supervision, and whether the resulting model extrapolates to sequence lengths far longer than those seen during training – consistent with the theory once embeddings are clusterable.

Second, on CIFAR-10, where intra-class variability is substantially higher, we ask whether MaxSketch still counts exactly when the embedding comes from a frozen, label-free self-supervised encoder, and whether a single readout works across stream lengths, including lengths never seen in training.

Third, we evaluate MaxSketch on real face streams using fixed pretrained embeddings, where the assumptions of our theory are only approximately satisfied, and we compare against the streaming algorithm of Zhang [29].

5.1 End-to-end learning to count on MNIST

We study the problem of learning to count the number of distinct latent objects in a sequence under aggregate supervision only. Each data point is a sequence $X = (x_1, \ldots, x_n)$ of MNIST images, and the only supervision is the number of distinct latent classes $k(X) = |\{z_1, \ldots, z_n\}|$, appearing in X; the image labels $z_i \in \{1, \ldots, 10\}$ are never observed. The model consists of an encoder $\phi_\theta : \mathcal{X} \to \mathbb{R}^d$, applied independently to each image, and an aggregator $f_\psi : (\mathbb{R}^d)^n \to \mathbb{R}$ applied to the resulting set of embeddings, both trained jointly to predict $k(X)$.

Beyond MaxSketch, we include two natural baselines that share the same encoder but differ in f_ψ: a DeepSets aggregator [27], which sum-pools embeddings, and an LSTM aggregator [12] that processes them sequentially. DeepSets is permutation-invariant but conflates frequency with distinctness; the LSTM is included as a non-set baseline.

All three models reach near-perfect accuracy across sequence lengths $n \in \{2, 5, 10, 20, 50, 100\}$ (Figure 2), and ± 1 accuracy (within one of the correct value) is exactly 100% throughout. This is consistent with the theory: MNIST embeddings have low intra-class variability and strong inter-class separation, so learning an encoder to satisfy the (η, ρ)-clusterability assumption becomes substantially easier. In this regime, the aggregation mechanism plays a relatively minor role, and even sum- or sequence-based models can recover the correct count. These results are a sanity check rather than a comparison: when the embeddings cluster this well, any reasonable aggregator can count.

The more informative test is whether the learned model extrapolates beyond the count and length range seen at training. Table 1 reports MaxSketch trained at $n_{\text{train}} \in \{10, 20, 50\}$ and evaluated at $n_{\text{test}} \in \{150, 250, 500\}$. Models trained on shorter sequences exhibit some underestimation bias for very long sequences, but keep high ± 1 accuracy, indicating that the learned statistic still carries meaningful count information.

The learned encoder, evaluated independently by 1-NN classification on test images, attains 1.00 accuracy regardless of n_{train}, so the failure at long sequences is not a representation collapse. Rather, it is a calibration failure of the regression head. This behaviour matches the theoretical picture: under clusterability the MaxSketch statistic tracks k, so what remains is calibrating the readout. A readout fitted on short streams is miscalibrated on much longer ones. In a stream of 10 images, not all of the chosen classes appear, so the readout rarely sees large counts during training.

Table 1: Generalization of MaxSketch models to unseen, longer sequence lengths on MNIST. Each model is trained on a specific sequence length n_{train} and evaluated on sequences of lengths n_{test} not seen during training. Numbers indicate exact count accuracy, with ± 1 accuracy shown in parentheses.

n_{train}	$n_{\text{test}} = 150$	$n_{\text{test}} = 250$	$n_{\text{test}} = 500$
10	0.37 (0.97)	0.25 (0.90)	0.06 (0.78)
20	0.70 (0.99)	0.58 (0.98)	0.37 (0.96)
50	0.99 (1.0)	0.98 (1.0)	0.96 (1.0)

5.2 Counting on CIFAR-10 with a frozen encoder

Self-supervised encoders are trained without labels to map views of the same object close together and different objects far apart, which is close to the geometry our analysis assumes. We therefore test MaxSketch on CIFAR-10 with a frozen DINOv2 encoder [18]. CIFAR-10 class labels are used only to build streams and their count targets, exactly as in the MNIST experiment. Full details can be found in Appendix C.2.

We use two readouts, both fitted with count supervision only: an MLP on the m registers, trained once on streams of lengths $n \leq 100$ and evaluated without retraining on streams of length up to $n = 500$, and the one-dimensional isotonic map of $S(X)$ analysed in Section 4, calibrated separately for each n.

Table 2 reports the results over three random seeds. With DINOv2-L features, the MLP readout recovers the exact count on 91–100% of test streams, and every prediction is within one of the truth at every length. Its accuracy does not degrade at $n = 250$ and $n = 500$, which are longer than any training stream; the smaller DINOv2-B encoder is only slightly behind. The isotonic readout of the scalar $S(X)$ is within one on 94–100% of streams and exact on 53–71%. This is expected: with up to ten classes, exact counting means telling k apart from $k + 1$, a relative difference of as small as 10%, and Theorem 4.4 requires more projections and tighter clusters the finer the distinction. The MLP is not covered by our analysis, but it reads all m registers instead of their average, and so has more information to work with.

As in the MNIST experiment, the readout is trained and tested on the same ten classes, with disjoint training and test images.

5.3 Counting Unique Faces in Real-World Image Sequences

We now turn to the setting of counting people in face streams. Here the number of distinct objects runs into the thousands, and the embedding comes from a face-recognition encoder that is never adapted

Table 2: Counting on CIFAR-10 with a frozen self-supervised encoder (DINOv2, trained without labels). Embeddings are centered and projected onto their top principal directions, fitted on unlabeled training images. Each readout is trained with count supervision on streams of training images and evaluated on streams of test images, with $k \in \{1, \ldots, 10\}$ and $m = 4{,}096$. The MLP readout is trained once on lengths $n \leq 100$, so $n = 250$ and $n = 500$ are lengths not seen during training; the isotonic readout of $S(X)$ is calibrated separately for each n. Numbers indicate exact count accuracy (mean over 3 seeds; standard deviation ≤ 0.03), with ± 1 accuracy shown in parentheses.

Encoder, readout	$n = 10$	$n = 20$	$n = 50$	$n = 100$	$n = 250$	$n = 500$
DINOv2-L, MLP on registers	**0.91** (1.00)	**0.91** (1.00)	**0.97** (1.00)	**1.00** (1.00)	**1.00** (1.00)	**1.00** (1.00)
DINOv2-B, MLP on registers	0.88 (1.00)	0.89 (1.00)	0.96 (1.00)	0.98 (1.00)	0.98 (1.00)	0.96 (1.00)
DINOv2-L, isotonic on $S(X)$	0.54 (0.96)	0.53 (0.94)	0.65 (0.98)	0.71 (1.00)	0.70 (1.00)	0.66 (0.99)

to counting. We use the CelebA face dataset [17], restricted to identities with at least two images so that each identity can appear multiple times in a stream. We also remove identities whose mean embeddings are nearly identical (cosine similarity above 0.85), which likely denote the same person under two labels. The remaining $9{,}868$ identities are split uniformly at random into a calibration half and an evaluation half of $4{,}934$ identities each. For each image we detect and crop the face with MTCNN [28] and compute a 512-dimensional embedding with a pretrained FaceNet encoder [21], trained on VGGFace2 [1], which is disjoint from CelebA. Embeddings are centred using the mean of the calibration identities, L2-normalized, and held fixed throughout, so the encoder is never adapted to the counting task.

Each evaluation stream is built by drawing a target distinct count $k \in \{100, 200, \ldots, 4900\}$ from the held-out identities and then sampling $n = 20{,}000$ images of those identities, in which the same person typically appears many times. This is the regime motivating our setting in the first place: a video feed or a large photo collection in which repeated observations are the rule. We use $m = 4{,}096$ Gaussian projections and a one-dimensional monotone readout fit by isotonic regression on streams built from calibration identities; the readout is fit once and applied unchanged to every evaluation stream. Because k ranges into the thousands we report the accuracy within $\pm 5\%$ and $\pm 10\%$ of k.

The most directly comparable prior work is the streaming algorithm of Zhang for distinct count estimation in general metric spaces. The algorithm is parameterized by a near-duplicate distance threshold α (Definition 1.1 of [29]), and its correctness guarantee assumes the dataset is *well-shaped*: every two points belonging to different underlying objects must be at distance strictly greater than 2α, while every two points belonging to the same object must be at distance at most α. Real face embeddings do not satisfy this condition at any α as intra-identity and inter-identity distance distributions overlap. One should therefore expect the algorithm to be sensitive to the choice of α and to errors of both signs. We implement Algorithm 1 of [29] and choose $\alpha \in \{0.6, 0.7, \ldots, 1.4\}$ to minimize its relative error on calibration streams (smaller thresholds split nearly every identity; at $\alpha = 0.4$ the estimate is about 7.5 times the true k), which gives $\alpha = 1.1$. With this single threshold, Zhang's estimate is off by 61% on average: it is accurate only around $k \approx 500$ and saturates near 780 for larger k, since with many identities almost every new face lies within α of a stored representative.

By contrast, due to MaxSketch's nature as a "statistical embedding" mild violations of (η, ρ)-clusterability shift the expectation of $S(X)$ continuously, and the calibrated readout absorbs these biases. Over all evaluation streams (Figure 1), MaxSketch has a mean relative error of 4.7%, and 91% of its estimates lie within $\pm 10\%$ of k. Accuracy decreases slowly with k: within $\pm 10\%$, it is 99% for $k \leq 1{,}000$, 94% for $1{,}000 < k < 2{,}500$, and 85% for $k \geq 2{,}500$. Within the stricter $\pm 5\%$ band, around 60% of estimates are correct overall, falling from around 80% for $k \leq 1{,}000$ to around 50% for $k \geq 2{,}500$. Overall, this demonstrates that MaxSketch can extract multiplicity information directly from pretrained embedding spaces without requiring clustering or identity supervision at inference time. Such approximate counts may be useful in large-scale monitoring and analytics scenarios where rough population estimates are sufficient and explicit identity resolution is unnecessary.

6 Limitations

Our theoretical guarantees rely on an explicit geometric separation assumption: embeddings must be (η, ρ)-clusterable, with small within-cluster variance and sufficiently weak correlations across cluster

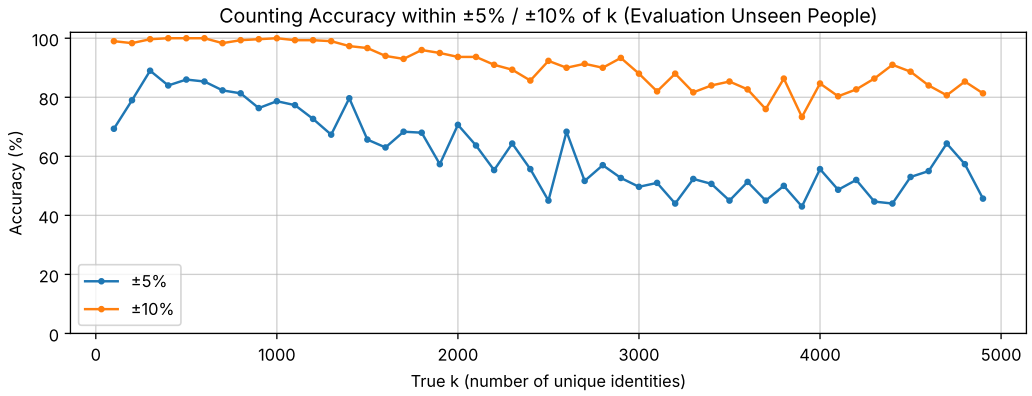

Figure 1: MaxSketch ($m = 4096$) on CelebA face streams of $n = 20{,}000$ images from k unseen identities: fraction of streams whose estimate is within $\pm 5\%$ and $\pm 10\%$ of k, averaged over three independent sketches, for $k \in \{100, 200, \ldots, 4900\}$.

centers. When this assumption is violated, e.g., in visual data with substantial intra-class variability or semantically overlapping categories, our guarantees no longer apply: within-cluster spread perturbs the projection maxima, and the gap between different cluster counts is no longer guaranteed. The face experiments illustrate this regime. MaxSketch still counts within about 5% on average, but its error grows with k.

Additionally, understanding when count supervision alone can provably induce clusterable embedding geometry, rather than merely exploit it when present, remains a central open problem.

7 Conclusion

We introduced MaxSketch, a max-linear streaming sketch built from random Gaussian projections, and showed that it provably estimates the number of distinct latent objects in a stream under a natural clusterability assumption. The central insight is that geometric structure common in modern learned representations, that is tight within-cluster alignment and weak cross-cluster correlation, is precisely the structure needed to make extreme-value statistics informative about cardinality and avoid the known $\sqrt{n}$ memory cost that is necessary for general noisy data. Under this assumption, MaxSketch achieves $(1+\varepsilon)$-multiplicative accuracy using only $O(\varepsilon^{-2}\log(2/\delta))$ Gaussian projections.

On the practical side, MaxSketch counts almost exactly when embeddings are well clustered: on MNIST with an end-to-end encoder, and on CIFAR-10 with a frozen label-free DINOv2 encoder, where a single MLP readout extrapolates to streams five times longer than any seen in training. On real face streams, where clusterability holds only approximately, the calibrated readout absorbs residual bias from mild violations, yielding robust approximate counts across a wide identity range.

Two open problems would sharpen the theoretical picture. First, Proposition 4.6 shows that MaxSketch's projection count is optimal up to absolute constants for estimators based only on independent Gaussian-max values. A lower bound for general streaming algorithms under our clusterability assumption remains open. Second, our guarantees assume clusterability is already present in the encoder; a theory of when aggregate count supervision alone can provably induce this geometry would bring the theoretical and empirical settings into alignment, replacing a geometric assumption on the encoder with a condition on the learning objective alone.

Acknowledgments

This work was done in Archimedes/Athena RC supported by a PhD fellowship.

Constantine Caramanis was partially supported by the NSF IFML Institute (NSF 2019844), and the NSF AI-EDGE Institute (NSF 2112471).

Competing interests

The authors declare no competing interests.

References

[1] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[2] Di Chen and Qin Zhang. Streaming algorithms for robust distinct elements. In *Proceedings of the 2016 International Conference on Management of Data*, pages 1433–1447, 2016.

[3] Jiecao Chen and Qin Zhang. Distinct sampling on streaming data with near-duplicates. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 369–382, 2018.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.

[5] Marianne Durand and Philippe Flajolet. Loglog counting of large cardinalities. In *European Symposium on Algorithms*, pages 605–617. Springer, 2003.

[6] Philippe Flajolet, Éric Fusy, Olivier Gandouet, and Frédéric Meunier. Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm. *Discrete mathematics & theoretical computer science*, (Proceedings), 2007.

[7] Philippe Flajolet and G Nigel Martin. Probabilistic counting algorithms for data base applications. *Journal of computer and system sciences*, 31(2):182–209, 1985.

[8] Guangshuai Gao, Junyu Gao, Qingjie Liu, Qi Wang, and Yunhong Wang. Cnn-based density estimation and crowd counting: A survey. *arXiv preprint arXiv:2003.12783*, 2020.

[9] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[10] XY Han, Vardan Papyan, and David L Donoho. Neural collapse under mse loss: Proximity to and dynamics on the central path. *arXiv preprint arXiv:2106.02073*, 2021.

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[13] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613, 1998.

[14] Piotr Indyk, Rajeev Motwani, Prabhakar Raghavan, and Santosh Vempala. Locality-preserving hashing in multidimensional spaces. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 618–625, 1997.

[15] Daniel M Kane, Jelani Nelson, and David P Woodruff. An optimal algorithm for the distinct elements problem. In *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 41–52, 2010.

[16] Ryan Kortvelesy, Steven Morad, and Amanda Prorok. Generalised f-mean aggregation for graph neural networks. *Advances in Neural Information Processing Systems*, 36:34439–34450, 2023.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[18] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[19] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

[20] Giovanni Pellegrini, Alessandro Tibo, Paolo Frasconi, Andrea Passerini, and Manfred Jaeger. Learning aggregation functions. *arXiv preprint arXiv:2012.08482*, 2020.

[21] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[22] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[23] Maximilian Soelch, Adnan Akhundov, Patrick van der Smagt, and Justin Bayer. On deep set learning and the choice of aggregations. In *International Conference on Artificial Neural Networks*, pages 444–457. Springer, 2019.

[24] Kevin Tanguy. Some superconcentration inequalities for extrema of stationary gaussian processes. *Statistics & Probability Letters*, 106:239–246, 2015.

[25] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[26] Edward Wagstaff, Fabian Fuchs, Martin Engelcke, Ingmar Posner, and Michael A Osborne. On the limitations of representing functions on sets. In *International conference on machine learning*, pages 6487–6494. PMLR, 2019.

[27] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

[28] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016.

[29] Qin Zhang. Robust statistical analysis on streaming data with near-duplicates in general metric spaces. *Proceedings of the ACM on Management of Data*, 3(2):1–25, 2025.

[30] Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features. In *International Conference on Machine Learning*, pages 27179–27202. PMLR, 2022.

[31] Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

A Appendix: Extended Related Work

A.1 Classical Sketching and Cardinality Estimation

The problem of estimating the number of distinct elements in a stream has been extensively studied in the data management and streaming algorithms literature. Flajolet and Martin [7] pioneered probabilistic counting, using $O(\varepsilon^{-2} \log n)$ bits for a $(1+\varepsilon)$-approximation. LogLog [5] and HyperLogLog [6] store $O(\varepsilon^{-2})$ counters of $O(\log \log n)$ bits each, for $O(\varepsilon^{-2} \log \log n)$ bits in total. Kane, Nelson and Woodruff [15] give an algorithm with the optimal $O(\varepsilon^{-2} + \log n)$ bits.

These classical methods share two defining features: (i) they assume deterministic, identical representations for repeated elements (via random hashing), and (ii) they use hand-designed random projections and analytical estimators derived from probabilistic analysis. While highly effective, this paradigm raises a natural question: can modern machine learning discover task-specific sketching statistics end-to-end?

A.2 Self-Supervised Learning and Representation Quality

Our method's success depends critically on good embeddings satisfying clusterability (Definition 2.1). Recent breakthroughs in self-supervised learning have produced embeddings where similar items cluster and dissimilar items separate, directly supporting our geometric assumptions.

SimCLR [4]: Uses contrastive loss to learn representations where augmentations of the same image cluster, and different images separate. Achieves clustering structure aligned with our geometric model, producing embeddings with strong within-class alignment and between-class separation.

MoCo [11]: Momentum contrast provides an alternative contrastive framework, showing robust clustering across diverse datasets without requiring large batch sizes. Demonstrates the robustness of clustering assumptions across different SSL architectures.

BYOL [9]: Learns clustering without explicit negative pairs, showing that clustering emerges naturally even without contrastive objectives. Further validates that learned representations naturally satisfy clusterability.

Neural Collapse Phenomenon [19, 30]: Recent work formalizes why deep classifiers learn near-orthogonal class representations in the terminal phase of training. [31] provide further geometric characterization, confirming that learned embeddings have the low-correlation structure we assume.

A.3 Counting Objects in Images

[8] surveys methods that regress the number of objects in a single image (e.g., people in a crowd). These methods count every instance in one image, so the same person seen twice is counted twice. They do not estimate the number of distinct objects across a stream of images, and they come with no theoretical guarantees.

A.4 Permutation-invariant models and set functions

Permutation-invariant neural networks aggregate unordered inputs via sum, mean, or max pooling. DeepSets [27] showed that, for countable input domains, every permutation-invariant function on finite sets can be written as $g\big(\sum_i \phi(x_i)\big)$, and this framework has become foundational for learning on variable-sized sets. For continuous inputs the picture is more delicate: fixed-dimensional sum-based representations have fundamental expressivity limitations, and exact decompositions can require latent dimension that grows with the set size [26, 23, 20]. Learnable aggregators have been proposed to address this, but approximating complex set functions often requires going beyond fixed sum or max pooling and can be hard to learn [16, 20]. The choice of aggregator also determines which statistics can be recovered: sum and mean pooling conflate frequency with distinctness. In contrast, we analyze a specific max-linear sketch and give finite-sample, multiplicative error guarantees under an explicit geometric clusterability assumption, explaining why maxima of random projections isolate extreme-value statistics that grow monotonically with the number of distinct objects.

B Proofs

B.1 Preliminaries and technical lemmas

Notation. As in Definition 3.1, for a sequence $\tilde{X} = (\tilde{x}_1, \dots, \tilde{x}_n) \subset \mathbb{S}^{d-1}$ and projections $w_1, \dots, w_m \in \mathbb{R}^d$ we write

$$M_j(\tilde{X}) := \max_{1 \le i \le n} \langle w_j, \tilde{x}_i \rangle, \qquad S(\tilde{X}) := \frac{1}{m} \sum_{j=1}^{m} M_j(\tilde{X}).$$

Lemma B.1 (Explicit form of Lemma 4.2)**.** *Let $\tilde{X} = (\tilde{x}_1, \dots, \tilde{x}_n) \subset \mathbb{S}^{d-1}$ be (η, ρ)-clusterable with centers $x^{(1)}, \dots, x^{(k)} \in \mathbb{S}^{d-1}$ and assignment $r : [n] \to [k]$, so that $\langle \tilde{x}_i, x^{(r(i))} \rangle \ge 1 - \eta/2$ for all $i \in [n]$. Let $w \sim \mathcal{N}(0, I_d)$. Then*

$$\Big| \, \mathbb{E}\Big[\max_{i \le n} \langle w, \tilde{x}_i \rangle \Big] - \mathbb{E}\Big[\max_{r \le k} \langle w, x^{(r)} \rangle \Big] \, \Big| \;\le\; \sqrt{2\eta \log(2n)}.$$

Proof. For each i, define $v_i := \tilde{x}_i - x^{(r(i))}$. By clusterability, $\|v_i\|^2 \le \eta$. Then $\langle w, v_i \rangle$ is centered Gaussian with variance $\|v_i\|^2 \le \eta$.

Since r is surjective, $\max_i \langle w, x^{(r(i))} \rangle = \max_{r \le k} \langle w, x^{(r)} \rangle$. Using $|\max_i a_i - \max_i b_i| \le \max_i |a_i - b_i|$,

$$\Big| \mathbb{E} \max_i \langle w, \tilde{x}_i \rangle - \mathbb{E} \max_{r \le k} \langle w, x^{(r)} \rangle \Big| \le \mathbb{E} \max_{i \le n} |\langle w, v_i \rangle| \le \sqrt{2\eta \log(2n)},$$

where the last step is the standard bound for the maximum of the $2n$ centered Gaussians $\pm\langle w, v_i \rangle$, each of variance at most η.

□

Lemma B.2 (Explicit form of Lemma 4.1)**.** *Let $(G_1, \dots, G_k)$ be a centered Gaussian vector with $\mathrm{Var}(G_r) = 1$ and $|\mathrm{Cov}(G_r, G_s)| \le \rho$ for $r \ne s$, where $\rho \in [0, 1)$. Let $Z_1, \dots, Z_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$. Then*

$$\sqrt{1-\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r \Big] \;\le\; \mathbb{E}\Big[\max_{r \le k} G_r \Big] \;\le\; \sqrt{1+\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r \Big].$$

Proof. We use the Sudakov–Fernique inequality [25]: if $(X_1, \dots, X_k)$ and $(Y_1, \dots, Y_k)$ are centered Gaussian vectors with $\mathbb{E}(X_r - X_s)^2 \le \mathbb{E}(Y_r - Y_s)^2$ for all r, s, then $\mathbb{E} \max_r X_r \le \mathbb{E} \max_r Y_r$. For our vector, $\mathbb{E}(G_r - G_s)^2 = 2 - 2\,\mathrm{Cov}(G_r, G_s)$ for $r \ne s$.

Lower bound. Let $Z \sim \mathcal{N}(0, 1)$ be independent of $Z_1, \dots, Z_k$, and set $G'_r := \sqrt{1-\rho}\, Z_r + \sqrt{\rho}\, Z$. Then $\mathbb{E}(G'_r - G'_s)^2 = 2 - 2\rho \le 2 - 2\,\mathrm{Cov}(G_r, G_s) = \mathbb{E}(G_r - G_s)^2$, since $\mathrm{Cov}(G_r, G_s) \le \rho$. By Sudakov–Fernique, $\mathbb{E} \max_r G_r \ge \mathbb{E} \max_r G'_r$. Since $\max_r G'_r = \sqrt{1-\rho} \max_r Z_r + \sqrt{\rho}\, Z$ and $\mathbb{E}Z = 0$,

$$\mathbb{E}\Big[\max_{r \le k} G_r \Big] \;\ge\; \sqrt{1-\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r \Big].$$

Upper bound. Set $Y_r := \sqrt{1+\rho}\, Z_r$. Then $\mathbb{E}(Y_r - Y_s)^2 = 2 + 2\rho \ge 2 - 2\,\mathrm{Cov}(G_r, G_s) = \mathbb{E}(G_r - G_s)^2$, since $\mathrm{Cov}(G_r, G_s) \ge -\rho$. By Sudakov–Fernique,

$$\mathbb{E}\Big[\max_{r \le k} G_r \Big] \;\le\; \mathbb{E}\Big[\max_{r \le k} Y_r \Big] \;=\; \sqrt{1+\rho}\, \mathbb{E}\Big[\max_{r \le k} Z_r \Big].$$

□

Lemma B.3 (Concentration of Gaussian max projections)**.** *Fix unit vectors $u_1, \dots, u_n \in \mathbb{S}^{d-1}$ and let $f(w) = \max_{i \le n} \langle w, u_i \rangle$. Then f is 1-Lipschitz, and for $w \sim \mathcal{N}(0, I_d)$,*

$$\big\| f(w) - \mathbb{E} f(w) \big\|_{\psi_2} \le C$$

for an absolute constant C.

Proof. For any w, w',

$$f(w) - f(w') \le \max_i \langle w - w', u_i \rangle \le \|w - w'\|,$$

and the same holds with w, w' swapped, so f is 1-Lipschitz. The bound then follows from Gaussian concentration for Lipschitz functions [25]. □

Lemma B.4 (Mean gap)**.** *Let $Z_1, \dots, Z_{k'} \overset{\text{i.i.d.}}{\sim} N(0,1)$ with $k' = \lceil (1+\varepsilon)k \rceil$. Define*

$$\Delta_1(k, \varepsilon) := \mathbb{E}\left[\max_{i \le k'} Z_i\right] - \mathbb{E}\left[\max_{i \le k} Z_i\right].$$

There exists an absolute constant $c > 0$ such that for all integers $k \ge 2$ and all $\varepsilon \in (0, 1]$,

$$\Delta_1(k, \varepsilon) \;\ge\; \frac{c\,\varepsilon}{\sqrt{\log k}}.$$

Proof. Let

$$A := \max_{i \le k} Z_i, \qquad B := \max_{k < i \le k'} Z_i, \qquad \ell := k' - k.$$

Since the index sets are disjoint and the Z_i are independent, A and B are independent. Also $\ell = \lceil (1+\varepsilon)k \rceil - k \ge \varepsilon k$, with $\ell \ge 1$ whenever $\varepsilon k > 0$.

We first show a pointwise inequality. For every realization and every $t \in \mathbb{R}$,

$$(B - A)_+ \;\ge\; (B - t)_+ \cdot \mathbf{1}_{\{A \le t\}}.$$

This follows by a simple case analysis:

- If $A \le t$ and $B \le t$: LHS $= (B - A)_+ \ge 0$, RHS $= 0$. OK.
- If $A \le t$ and $B > t$: LHS $= B - A$ (since $B > t \ge A$ implies $B > A$), RHS $= B - t$. We need $B - A \ge B - t$, equivalently $A \le t$, which holds.
- If $A > t$ and $B \le t$: RHS $= 0$ (indicator vanishes); LHS ≥ 0. OK.
- If $A > t$ and $B > t$: RHS $= 0$; LHS ≥ 0. OK.

Taking expectations and using the independence of A and B:

$$\Delta_1(k, \varepsilon) = \mathbb{E}[\max_{i \le k'} Z_i - \max_{i \le k} Z_i] \overset{(*)}{=} \mathbb{E}[(B - A)_+] \ge \Pr(A \le t) \cdot \mathbb{E}[(B - t)_+]. \tag{1}$$

The equality $(*)$ uses that $\max_{i \le k'} Z_i = \max(A, B)$, and $\max(A, B) - A = (B - A)_+$.

Set

$$t := \Phi^{-1}\big(1 - \tfrac{1}{2k}\big), \qquad \text{so} \qquad 1 - \Phi(t) = \tfrac{1}{2k},$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution.

We will lower bound $\mathbb{E}[(B - t)_+]$ and combine with (1).

First, we prove that $\Pr(A \le t) \ge \frac{1}{2}$ for all $k \ge 1$.

$$\Pr(A \le t) = \Phi(t)^k = \big(1 - \tfrac{1}{2k}\big)^k.$$

Since $(1 - \frac{1}{2k})^k$ is increasing in $k \ge 1$, we have

$$\Pr(A \le t) = \left(1 - \frac{1}{2k}\right)^k \ge \frac{1}{2}. \tag{2}$$

Additionally, we need to prove an upper bound on t, for all $k \ge 2$.

The standard Gaussian tail bound (Chernoff) gives $1 - \Phi(t) \le e^{-t^2/2}$ for $t \ge 0$. Thus $1/(2k) = 1 - \Phi(t) \le e^{-t^2/2}$, giving $t^2/2 \le \log(2k)$, i.e., $t \le \sqrt{2\log(2k)}$. For $k \ge 2$, $2k \le k^2$, so $\log(2k) \le 2\log k$, giving $t \le 2\sqrt{\log k}$.

Furthermore, we need to prove a bound on $t+2$. We see that the ratio $(2\sqrt{\log k}+2)/(5\sqrt{\log k})$ is decreasing in k, so the bound

$$t+2 \le 5\sqrt{\log k} \tag{3}$$

holds for all $k \ge 2$. We now derive a uniform lower bound on $\Pr(B > t+s)$ for small s, which will yield a lower bound on $\mathbb{E}[(B-t)_+]$ via integration.

With t as defined and $s_0 := 1/(2(t+2))$, for all $\varepsilon \in (0,1]$ and $k \ge 2$,

$$\mathbb{E}[(B-t)_+] \ge \frac{\kappa\,\varepsilon}{2(t+2)}, \qquad \kappa := \frac{1-e^{-1}}{2e^2}.$$

We now control $\Pr(B > t+s)$ uniformly for $s \in [0, s_0]$ and then integrate.

(a) Hazard-rate bound. The Gaussian hazard rate is $h(u) := \phi(u)/(1-\Phi(u))$ for $u \in \mathbb{R}$. We claim:

$$h(u) \le 2(u+1) \quad \text{for all } u \ge 0. \tag{4}$$

For $u \ge 1$: by Mills' ratio, $1-\Phi(u) \ge \frac{u}{1+u^2}\phi(u)$, so $h(u) \le \frac{1+u^2}{u} = u + \frac{1}{u} \le u+1 \le 2(u+1)$.

A direct computation shows $h(1) \approx 1.525$, and $h(u)$ is increasing on $[0,\infty)$, hence $h(u) \le h(1)$ for $u \in [0,1]$.

(b) Integral bound. For $s \in [0, s_0]$, by (4),

$$\int_t^{t+s} h(u)\,du \le 2\int_t^{t+s} (u+1)\,du = 2s(t+1) + s^2 \le 2s_0(t+1) + s_0^2.$$

Substituting $s_0 = 1/(2(t+2))$:

$$2s_0(t+1) + s_0^2 = \frac{t+1}{t+2} + \frac{1}{4(t+2)^2} \le 1 + \frac{1}{4(t+2)^2} \le 1 + \frac{1}{16} \le 2,$$

where we used $t \ge 0$, hence $t+2 \ge 2$ and $1/(4(t+2)^2) \le 1/16$.

(c) Tail of Φ. The fundamental theorem of calculus applied to $-\log(1-\Phi)$ (whose derivative is h) gives

$$1-\Phi(t+s) = (1-\Phi(t)) \cdot \exp\!\left(-\int_t^{t+s} h(u)\,du\right) \ge \frac{1}{2k}\cdot e^{-2} \quad \text{for all } s \in [0, s_0].$$

(d) Tail of B. Since $B = \max_{k<i\le k'} Z_i$ is the max of ℓ i.i.d. standard normals,

$$\Pr(B > t+s) = 1 - \Phi(t+s)^\ell \ge 1 - \exp(-\ell(1-\Phi(t+s))),$$

where we used $(1-p)^\ell \le e^{-\ell p}$ for $p \in [0,1]$. Combining with (c) and $\ell \ge \varepsilon k$:

$$\ell(1-\Phi(t+s)) \ge \varepsilon k \cdot \frac{e^{-2}}{2k} = \frac{\varepsilon\, e^{-2}}{2}.$$

Setting $x := \varepsilon e^{-2}/2 \in (0, e^{-2}/2] \subset (0,1]$, we use the elementary inequality $1-e^{-x} \ge (1-e^{-1})x$ for $x \in [0,1]$ (the function $g(x) = (1-e^{-x})/x$ is decreasing on $(0,\infty)$ with $g(1) = 1-e^{-1}$):

$$\Pr(B > t+s) \ge 1 - e^{-x} \ge (1-e^{-1})x = (1-e^{-1}) \cdot \frac{\varepsilon\, e^{-2}}{2} = \kappa\,\varepsilon.$$

(e) Integration. Using $\mathbb{E}[(B-t)_+] = \int_0^\infty \Pr(B > t+s)\,ds$:

$$\mathbb{E}[(B-t)_+] \ge \int_0^{s_0} \kappa\,\varepsilon\,ds = \kappa\,\varepsilon \cdot s_0 = \frac{\kappa\,\varepsilon}{2(t+2)}. \tag{5}$$

(f) Combining the bounds.

By Equations (1), (2), (3), and (5):

$$\Delta_1(k,\varepsilon) \ge \Pr(A \le t) \cdot \mathbb{E}[(B-t)_+] \ge \frac{1}{2}\cdot\frac{\kappa\,\varepsilon}{2(t+2)} \ge \frac{\kappa\,\varepsilon}{4\cdot 5\sqrt{\log k}} = \frac{\kappa\,\varepsilon}{20\sqrt{\log k}}.$$

which proves the lemma with $c = \kappa/20$. □

Lemma B.5 (Explicit form of Lemma 4.3)**.** *Let $k \geq 2$, $\varepsilon \in (0, 1]$, and $k' = \lceil (1+\varepsilon)k \rceil$. Let $w \sim \mathcal{N}(0, I_d)$, and let $x^{(1)}, \ldots, x^{(k')}$ be unit vectors satisfying $|\langle x^{(i)}, x^{(j)} \rangle| \leq \rho$ for $i \neq j$, where $0 \leq \rho \leq c_\rho \varepsilon / \log k$, where $c_\rho > 0$ is a sufficiently small absolute constant.*

Define

$$\Delta_{\text{centers}}(k) = \mathbb{E}\left[\max_{1 \leq i \leq k'} \langle w, x^{(i)} \rangle\right] - \mathbb{E}\left[\max_{1 \leq i \leq k} \langle w, x^{(i)} \rangle\right]$$

Then we have:

$$\Delta_{\text{centers}}(k) = \Omega\left(\frac{\varepsilon}{\sqrt{\log k}}\right) \quad \textit{for all } k \geq 2$$

Proof. We have that $G_i = \langle w, x^{(i)} \rangle$ is a mean-zero Gaussian with $\text{Var}(G_i) = 1$. Additionally, since we have that $|\langle x^{(r)}, x^{(s)} \rangle| \leq \rho$, we get for $r \neq s$,

$$|\text{Cov}(G_r,\, G_s)| \;=\; |\mathbb{E}\big[G_r\, G_s\big]| \;=\; |\big\langle x^{(r)},\, x^{(s)} \big\rangle| \;\leq\; \rho.$$

Therefore, from Lemma B.2, we get that lower bounding $\Delta_{\text{centers}}(k)$ reduces to lower bounding

$$\Delta(k) := \sqrt{1-\rho}\,\mathbb{E}\left[\max_{1 \leq i \leq k'} Z_i\right] - \sqrt{1+\rho}\mathbb{E}\left[\max_{1 \leq i \leq k} Z_i\right],$$

where $Z_1, \ldots, Z_{k'} \sim N(0,1)$ i.i.d.

We rearrange this to get:

$$\Delta(k) = \underbrace{\mathbb{E}\left[\max_{1 \leq i \leq k'} Z_i\right] - \mathbb{E}\left[\max_{1 \leq i \leq k} Z_i\right]}_{\Delta_1} - (1-\sqrt{1-\rho})\underbrace{\mathbb{E}\left[\max_{1 \leq i \leq k'} Z_i\right]}_{\Delta_2} - (\sqrt{1+\rho}-1)\underbrace{\mathbb{E}\left[\max_{1 \leq i \leq k} Z_i\right]}_{\Delta_3}$$

First, focus on $\Delta_1 := \mathbb{E}\left[\max_{1 \leq i \leq k'} Z_i\right] - \mathbb{E}\left[\max_{1 \leq i \leq k} Z_i\right]$. By Lemma B.4 we get $\Delta_1 \;\geq\; c\,\frac{\varepsilon}{\sqrt{\log k}}$,

For the other two terms Δ_2 and Δ_3, $1 - \sqrt{1-\rho} \leq \rho$ and $\sqrt{1+\rho} - 1 \leq \rho$ for $\rho \in [0, 1)$, and $\mathbb{E} \max_{i \leq s} Z_i \leq \sqrt{2 \log s}$. Thus

$$(1 - \sqrt{1-\rho})\Delta_2 + (\sqrt{1+\rho} - 1)\Delta_3 \leq \rho\big(\sqrt{2\log k'} + \sqrt{2 \log k}\big) \leq 4\rho\sqrt{\log k} \leq 4c_\rho\, \frac{\varepsilon}{\sqrt{\log k}},$$

using $k' \leq 2k \leq k^2$. Hence $\Delta_{\text{centers}}(k) \geq \Delta(k) \geq (c - 4c_\rho)\, \varepsilon / \sqrt{\log k} \geq \frac{c}{2}\, \varepsilon / \sqrt{\log k}$ for $c_\rho \leq c/8$. □

Lemma B.6 (Argmax spread under weak correlation)**.** *Let $G = (G_1, \ldots, G_k)$, $k \geq 2$, be a centered Gaussian vector with $\text{Var}(G_r) = 1$ and $\text{Cov}(G_r, G_s) \leq \rho$ for $r \neq s$, where $0 \leq \rho \leq 0.18$. Let $I = \arg\max_r G_r$. There is an absolute constant k_0 such that for all $k \geq k_0$, $\max_{1 \leq i \leq k} \Pr(I = i) \leq 3k^{-1/4}$.*

Proof. Since $\text{Var}(G_r - G_s) = 2 - 2\,\text{Cov}(G_r, G_s) \geq 2 - 2\rho > 0$, the argmax is a.s. unique. Fix i and set $t := \sqrt{\frac{1}{2}\log k}$ and $s := 0.3\sqrt{\log k}$. If $I = i$ then either $G_i \geq t$ or $\max_{j \neq i} G_j < t$. First, $\Pr(G_i \geq t) \leq e^{-t^2/2} = k^{-1/4}$.

For the second event, let $Y_j = \sqrt{1-\rho}\, Z_j + \sqrt{\rho}\, Z_0$ for $j \neq i$, with Z_0, Z_j i.i.d. $\mathcal{N}(0,1)$. Both families have unit variances, and $\mathbb{E}(Y_j - Y_l)^2 = 2 - 2\rho \leq \mathbb{E}(G_j - G_l)^2$. Slepian's inequality [25] gives $\Pr(\max_{j \neq i} Y_j \geq t) \leq \Pr(\max_{j \neq i} G_j \geq t)$, hence

$$\Pr\Big(\max_{j \neq i} G_j < t\Big) \leq \Pr\Big(\max_{j \neq i} Y_j < t\Big) \leq \Pr(\sqrt{\rho}\, Z_0 \leq -s) + \Pr\Big(\max_{j \neq i} Z_j < u\Big),$$

where $u := \frac{t+s}{\sqrt{1-\rho}} \le 1.12\sqrt{\log k}$. First, if $\rho = 0$, then $\Pr(\sqrt{\rho}\, Z_0 \le -s) = 0$. If $0 < \rho \le 0.18$, $\Pr(\sqrt{\rho}\, Z_0 \le -s) \le e^{-s^2/(2\rho)} = k^{-0.045/\rho} \le k^{-1/4}$. For $k \ge 3$ we have $u \ge 1$, so the Mills-ratio bound $1 - \Phi(u) \ge \varphi(u)/(2u)$ and $u^2/2 \le 0.628 \log k$ give

$$\Pr\Big(\max_{j \ne i} Z_j < u\Big) = \Phi(u)^{k-1} \le \exp\big(-(k-1)(1-\Phi(u))\big) \le \exp\Big(-c\,\frac{k^{0.37}}{\sqrt{\log k}}\Big) \le k^{-1/4}$$

for $k \ge k_0$. Summing the three bounds proves the claim. □

Corollary B.7 (Superconcentration of weakly correlated maxima)**.** *Under the assumptions of Lemma B.6, if moreover $\rho \le c_0/\log k$ for an absolute $c_0 \le 0.12$, then $M := \max_r G_r$ satisfies $\|M - \mathbb{E}M\|_{\psi_1} \le C/\sqrt{\log k}$ for an absolute constant C.*

Proof. Let $k_1 := \max(k_0, 3^8)$. Let $\Gamma = (\Gamma_{ij})$ be the covariance matrix of G, so that $\Gamma_{ii} = 1$ and $\Gamma_{ij} = \mathrm{Cov}(G_i, G_j) \le \rho$ for $i \ne j$. For $k \ge k_1$, apply [24, Thm. 5] with the covering of $\{1, \dots, k\}$ by singletons and $r_0 := 2\rho$ (or $r_0 := 1/\log k$ if $\rho = 0$). Since $r_0 > \rho$, the only pairs with $\Gamma_{ij} \ge r_0$ have $i = j$, which singletons cover, and $\sum_D \mathbf{1}\{I \in D\} = 1$. By Lemma B.6, $\max_D \Pr(I \in D) \le 3k^{-1/4} \le k^{-1/8}$, because $k \ge 3^8$. Theorem 5 then gives $\Pr(|M - \mathbb{E}M| \ge t) \le 6e^{-ct/\sqrt{K}}$ with $K = \max(r_0, 8/\log k) \le (2c_0 + 8)/\log k$, hence the claim. For $2 \le k < k_1$, Lemma B.3 gives $\|M - \mathbb{E}M\|_{\psi_1} \le C_0 \le C_0\sqrt{\log k_1}/\sqrt{\log k}$. □

Lemma B.8 (Superconcentration of MaxSketch registers)**.** *Let $\widetilde{X} = (\widetilde{x}_1, \dots, \widetilde{x}_n) \subseteq \mathbb{S}^{d-1}$ be (η, ρ)-clusterable with $k \ge 2$ centers, and define $M(\widetilde{X}; w) = \max_{1 \le i \le n} \langle w, \widetilde{x}_i \rangle$, where $w \sim \mathcal{N}(0, I_d)$. Assume $\rho \le c_\rho \frac{\varepsilon}{\log n}$, $\eta \le c_\eta \frac{\varepsilon^2}{(\log n)^2}$ and $\varepsilon \in (0, 1/2]$. Then*

$$\Big\| M(\widetilde{X}; w) - \mathbb{E}M(\widetilde{X}; w) \Big\|_{\psi_1} \le \frac{C}{\sqrt{\log k}},$$

where

$$\|Y\|_{\psi_1} := \inf\left\{ s > 0 : \mathbb{E}\exp\left(\frac{|Y|}{s}\right) \le 2 \right\}.$$

Consequently, if

$$S(\widetilde{X}) = \frac{1}{m}\sum_{j=1}^{m} M(\widetilde{X}; w_j)$$

for independent Gaussian projections, then for every $u > 0$,

$$\Pr\Big(|S(\widetilde{X}) - \mathbb{E}S(\widetilde{X})| > u\Big) \le 2\exp\Big[-cm\min\Big\{u^2 \log k, u\sqrt{\log k}\Big\}\Big].$$

Proof. Let $x^{(1)}, \dots, x^{(k)}$ be the centers of $\widetilde{X}$, and define $M_{\mathcal{C}}(w) = \max_{1 \le r \le k} \langle w, x^{(r)} \rangle$.

For each r, set $G_r = \langle w, x^{(r)} \rangle$. Then $(G_1, \dots, G_k)$ is a centered Gaussian vector with

$$\mathrm{Var}(G_r) = 1, \qquad \mathrm{Cov}(G_r, G_s) = \langle x^{(r)}, x^{(s)} \rangle \le \rho \quad (r \ne s).$$

By Corollary B.7, which applies because $\rho \le c_\rho\, \varepsilon/\log n \le c_\rho/(2\log k) \le 0.12/\log k$ for $c_\rho \le 0.24$ (using $\varepsilon \le 1/2$ and $k \le n$),

$$\|M_{\mathcal{C}} - \mathbb{E}M_{\mathcal{C}}\|_{\psi_1} \le \frac{C}{\sqrt{\log k}}. \tag{6}$$

We must also control the within-cluster perturbations. For each observation, write $v_i = \widetilde{x}_i - x^{(r(i))}$. By clusterability, $\|v_i\|^2 = 2 - 2\langle \widetilde{x}_i, x^{(r(i))} \rangle \le \eta$. Define $D(w) = \max_{1 \le i \le n} |\langle w, v_i \rangle|$. The deterministic maximum inequality gives

$$\Big| M(\widetilde{X}; w) - M_{\mathcal{C}}(w) \Big| \le D(w). \tag{7}$$

Each $\langle w, v_i \rangle$ is centered Gaussian with variance at most η. A standard union bound gives us, $\Pr(D > u) \le 2n \exp\left(-\frac{u^2}{2\eta}\right)$. The standard maximal inequality for subgaussian random variables therefore implies $\|D\|_{\psi_2} \le C\sqrt{\eta \log(2n)}$, and hence also

$$\|D\|_{\psi_1} \le C\sqrt{\eta \log(2n)}. \tag{8}$$

Define $R(w) := M(\widetilde{X}; w) - M_{\mathcal{C}}(w)$ which measures the difference between the maxima over the observed sample $\widetilde{X}$ and over the set $\mathcal{C}$ of cluster centers.

By Equation (7), $|R| \le D$. By the triangle inequality, $\|R - \mathbb{E}R\|_{\psi_1} \le \|R\|_{\psi_1} + \|\mathbb{E}R\|_{\psi_1}$, and $\|\mathbb{E}R\|_{\psi_1} \le \|R\|_{\psi_1}$ by Jensen's inequality. Consequently,

$$\|R - \mathbb{E}R\|_{\psi_1} \le 2\|R\|_{\psi_1} \le 2\|D\|_{\psi_1} \le C\sqrt{\eta \log(2n)}. \tag{9}$$

Under the assumed bound on η we also get that,

$$\sqrt{\eta \log(2n)} \le C\frac{\varepsilon}{\sqrt{\log n}} \le \frac{C}{\sqrt{\log k}}. \tag{10}$$

Therefore, combining Equations (6), (9), and (10), using

$$M(\widetilde{X}) - \mathbb{E}M(\widetilde{X}) = \big(M_{\mathcal{C}} - \mathbb{E}M_{\mathcal{C}}\big) + (R - \mathbb{E}R),$$

gives

$$\Big\|M(\widetilde{X}) - \mathbb{E}M(\widetilde{X})\Big\|_{\psi_1} \le \frac{C}{\sqrt{\log k}}.$$

Finally, Bernstein's inequality for independent centered subexponential random variables [25] gives

$$\Pr\left(\left|\frac{1}{m}\sum_{j=1}^{m}\big(M(\widetilde{X}; w_j) - \mathbb{E}M(\widetilde{X}; w_j)\big)\right| > u\right) \le 2\exp\left[-cm\min\left\{u^2 \log k, u\sqrt{\log k}\right\}\right].$$

□

B.2 Proof of Theorem 4.4

Theorem B.9 (Explicit form of Theorem 4.4)**.** *Fix $n \ge 2$, $\varepsilon \in (0, 1/2]$, and $\delta \in (0, 1)$, and set $\beta := \frac{\varepsilon}{3}$. Let $\widetilde{X} = (\widetilde{x}_1, \dots, \widetilde{x}_n) \subseteq \mathbb{S}^{d-1}$ be (η, ρ)-clusterable with $2 \le k^\star \le n$ centers, where $\rho \le c_\rho \frac{\varepsilon}{\log n}, \eta \le c_\eta \frac{\varepsilon^2}{(\log n)^2}$, for sufficiently small absolute constants $c_\rho, c_\eta > 0$.*

Draw $w_1, \dots, w_m \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d)$ with $m \ge C\frac{\log(2/\delta)}{\varepsilon^2}$, for a sufficiently large constant C.

For every $s \ge 1$, let $\mu_s = \mathbb{E}[\max_{1 \le j \le s} Z_j]$ with $Z_1, \dots, Z_s \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$. Set $\bar{\rho} := c_\rho\, \varepsilon / \log n$, $\bar{\eta} := c_\eta\, \varepsilon^2/(\log n)^2$, $\tau_n := \sqrt{2\bar{\eta}\log(2n)}$, and set $L(s) := \sqrt{1-\bar{\rho}}\,\mu_s - \tau_n$, and $U(s) := \sqrt{1+\bar{\rho}}\,\mu_s + \tau_n$. Define $t_0 := 2, t_{r+1} := \lceil (1+\beta)\, t_r \rceil$, and let R be the smallest nonnegative integer such that $(1+\beta)t_R \ge n$. For every $r \in \{0, \dots, R-1\}$, define $q_r := \lfloor (1+\beta)t_r \rfloor$ and $\theta_r := \frac{U(t_r)+L(q_r+1)}{2}$. Given

$$S(\widetilde{X}) = \frac{1}{m}\sum_{j=1}^{m} \max_{1 \le i \le n} \langle w_j, \widetilde{x}_i \rangle,$$

let

$$\widehat{r} = \min\left\{r \in \{0, \dots, R-1\} : S(\widetilde{X}) \le \theta_r\right\}.$$

If this set is nonempty, output $\widehat{k} = q_{\widehat{r}}$; otherwise output $\widehat{k} = n$. Then we get the guarantee that $k^\star \le \widehat{k} \le (1+\varepsilon)k^\star$ with probability at least $1 - \delta$.

Moreover, the estimator is nondecreasing in $S(\widetilde{X})$ and uses $R = O\left(\frac{\log n}{\varepsilon}\right)$ precomputed thresholds.

Proof. Write $k := k^\star, S := S(\widetilde{X})$.

Step 1: Expectation bounds. Let $x^{(1)}, \dots, x^{(k)}$ be the centers. By the perturbation Lemma B.1 and $\eta \le \bar{\eta}$ (by assumption), we get

$$\left| \mathbb{E} \max_{i \le n} \langle w, \tilde{x}_i \rangle - \mathbb{E} \max_{r \le k} \langle w, x^{(r)} \rangle \right| \le \sqrt{2\eta \log(2n)} \le \tau_n.$$

By the Gaussian comparison Lemma B.2, and $\rho \le \bar{\rho}$ (by assumption) and $\mu_k \ge 0$,

$$\sqrt{1-\bar{\rho}}\, \mu_k \le \mathbb{E} \max_{r \le k} \langle w, x^{(r)} \rangle \le \sqrt{1+\bar{\rho}}\, \mu_k.$$

Therefore

$$L(k) \le \mathbb{E}S \le U(k). \tag{11}$$

Both L and U are nondecreasing, since $s \mapsto \mu_s$ is.

Step 2: Local gaps. This step does the same computation as in the proof of Lemma B.5, in the worst-case form needed here. Fix $r \in \{0, \dots, R-1\}$. Since $q_r = \lfloor (1+\beta) t_r \rfloor$, we have $q_r + 1 \ge \lceil (1+\beta) t_r \rceil$. Because $s \mapsto \mu_s$ is nondecreasing, Lemma B.4, applied with $k = t_r$ and accuracy parameter β, gives

$$\mu_{q_r+1} - \mu_{t_r} \ge \mu_{\lceil (1+\beta) t_r \rceil} - \mu_{t_r} \ge c \frac{\beta}{\sqrt{\log t_r}}. \tag{12}$$

Since $\beta \le 1/6$ and $t_r \ge 2$, we have that $q_r + 1 \le (1+\beta) t_r + 1 \le 2 t_r$. Using the standard bound $\mu_s \le \sqrt{2 \log s}$ for $s \ge 2$, we obtain

$$\mu_{q_r+1} + \mu_{t_r} \le C \sqrt{\log t_r}.$$

Consequently,

$$\begin{aligned} L(q_r+1) - U(t_r) &= \sqrt{1-\bar{\rho}}\, \mu_{q_r+1} - \sqrt{1+\bar{\rho}}\, \mu_{t_r} - 2\tau_n \\ &\ge (1-\bar{\rho}) \mu_{q_r+1} - (1+\bar{\rho}) \mu_{t_r} - 2\tau_n \\ &= \mu_{q_r+1} - \mu_{t_r} - \bar{\rho}(\mu_{q_r+1} + \mu_{t_r}) - 2\tau_n \\ &\ge \mu_{q_r+1} - \mu_{t_r} - C\bar{\rho} \sqrt{\log t_r} - 2\tau_n. \end{aligned}$$

Since $\bar{\rho} = c_\rho \varepsilon / \log n = 3 c_\rho \beta / \log n$ and $t_r \le n$, choosing $c_\rho > 0$ sufficiently small gives $C\bar{\rho}\sqrt{\log t_r} \le \frac{c}{4} \frac{\beta}{\sqrt{\log t_r}}$. Similarly, $2\tau_n = 2\sqrt{2\bar{\eta} \log(2n)} \le C\sqrt{c_\eta}\, \frac{\varepsilon}{\sqrt{\log n}} \le \frac{c}{4} \frac{\beta}{\sqrt{\log t_r}}$ for sufficiently small $c_\eta > 0$.

Combining these bounds with (12), there exists an absolute constant $c_1 > 0$ such that

$$L(q_r+1) - U(t_r) \ge c_1 \frac{\beta}{\sqrt{\log t_r}}. \tag{13}$$

Since

$$\theta_r = \frac{U(t_r) + L(q_r+1)}{2},$$

it follows that

$$\theta_r - U(t_r) \ge \frac{c_1 \beta}{2\sqrt{\log t_r}}, \qquad L(q_r+1) - \theta_r \ge \frac{c_1 \beta}{2\sqrt{\log t_r}}. \tag{14}$$

Step 3: Concentration at the scale of the true count. Define $a_k := \frac{c_2 \beta}{\sqrt{\log k}}$, where $c_2 > 0$ is a sufficiently small absolute constant.

Lemma B.8 gives

$$\Pr\left(|S - \mathbb{E}S| > a_k \right) \le 2 \exp\left[-cm \min\{c_2^2 \beta^2, c_2 \beta\} \right] \le 2 e^{-c' m \beta^2},$$

since $\beta \le 1/6$.

The assumed projection bound therefore implies

$$\Pr(\mathcal{E}_k) \ge 1 - \delta, \qquad \mathcal{E}_k := \{ |S - \mathbb{E}S| \le a_k \}. \tag{15}$$

From now on and until the end we work on $\mathcal{E}_k$.

Step 4: Every output below k is rejected. Suppose that $q_r < k$. Then $t_r \le q_r < k$. By Equation (11), monotonicity of L, and Equation (14),

$$\begin{aligned}\mathbb{E}S - \theta_r &\ge L(k) - \theta_r \\ &\ge L(q_r + 1) - \theta_r \\ &\ge \frac{c_1\beta}{2\sqrt{\log(t_r)}} \\ &\ge \frac{c_1\beta}{2\sqrt{\log k}}.\end{aligned}$$

Choose $c_2 < c_1/4$. Then, on $\mathcal{E}_k$, and because

$$a_k < \frac{c_1}{4}\frac{\beta}{\sqrt{\log k}} < \frac{c_1}{2}\frac{\beta}{\sqrt{\log k}} \le \mathbb{E}S - \theta_r,$$

we have that

$$S \ge \mathbb{E}S - a_k > \theta_r. \tag{16}$$

Thus no threshold whose associated output is below k can be accepted.

Step 5: A nearby threshold is accepted whenever one exists. Suppose there exists $r_+ \in \{0, \dots, R-1\}$ with $t_{r_+} \ge k$, and let r_+ be the smallest such index.

If $r_+ = 0$, then $k = 2 = t_0$. Otherwise, $t_{r_+-1} < k$ and hence

$$t_{r_+} = \left\lceil (1+\beta)t_{r_+-1} \right\rceil \le (1+\beta)(k-1) + 1 \le (1+\beta)k. \tag{17}$$

Thus, in either case,

$$t_{r_+} \le (1+\beta)k.$$

In particular,

$$\log(t_{r_+}) \le C \log k. \tag{18}$$

Using Equation (11), monotonicity of U, and Equation (14),

$$\begin{aligned}\theta_{r_+} - \mathbb{E}S &\ge \theta_{r_+} - U(k) \\ &\ge \theta_{r_+} - U(t_{r_+}) \\ &\ge \frac{c_1\beta}{2\sqrt{\log(t_{r_+})}} \\ &\ge \frac{c_3\beta}{\sqrt{\log k}}.\end{aligned}$$

Taking $c_2 < c_3$, on $\mathcal{E}_k$ we have

$$S \le \theta_{r_+}. \tag{19}$$

Therefore, if such an r_+ exists, the estimator accepts some index $\widehat{r} \le r_+$.

By Equation (16), its output cannot be below k, and hence

$$k \le q_{\widehat{r}}. \tag{20}$$

Since (q_r) is nondecreasing, $q_{\widehat{r}} \le q_{r_+} \le (1+\beta)t_{r_+} \le (1+\beta)^2 k$, and since $\beta = \varepsilon/3$ and $\varepsilon \le 1/2$, $(1+\beta)^2 \le 1+\varepsilon$, which in turn gives us:

$$k \le \widehat{k} \le (1+\varepsilon)k. \tag{21}$$

Step 6: The terminal output n. It remains to consider the case where no suitable r_+ exists or no threshold is accepted.

If a threshold is accepted, Equation (16) implies that its output is at least k. Since every $t_r < k$ in the absence of r_+,

$$\widehat{k} = q_{\widehat{r}} \le (1+\beta)t_{\widehat{r}} < (1+\beta)k \le (1+\varepsilon)k.$$

If no threshold is accepted, the estimator outputs n.

If $R = 0$, then

$$n \le (1+\beta)t_0 = 2(1+\beta) \le (1+\beta)k.$$

Suppose now that $R \ge 1$. If $t_{R-1} \ge k$, then Step 5 would imply that some threshold is accepted. Therefore

$$t_{R-1} < k.$$

Since k is integer,

$$t_{R-1} + 1 \le k.$$

Moreover,

$$t_R = \lceil (1+\beta)t_{R-1} \rceil \le (1+\beta)(t_{R-1}+1) \le (1+\beta)k.$$

By the definition of R,

$$n \le (1+\beta)t_R.$$

Consequently,

$$\widehat{k} = n \le (1+\beta)^2 k \le (1+\varepsilon)k.$$

The lower bound $k \le n = \widehat{k}$ is immediate.

Combining all cases, on $\mathcal{E}_k$,

$$k^\star \le \widehat{k} \le (1+\varepsilon)k^\star.$$

By Equation (15), this occurs with probability at least $1-\delta$.

Finally, the threshold sequence is nondecreasing, so the estimator is nondecreasing in S. Also,

$$t_r \ge 2(1+\beta)^r,$$

and therefore

$$R = O\left(\frac{\log n}{\log(1+\beta)}\right) = O\left(\frac{\log n}{\varepsilon}\right).$$

□

B.3 Finite readouts and deterministic weights

Corollary B.10 (Finite monotone threshold readout)**.** *Under the geometric assumptions of Theorem B.9, fix $\zeta \in (0,1)$ and take $m \ge C\varepsilon^{-2}\log(2/\zeta)$. Apply the construction of Theorem B.9 with its failure parameter set to $\delta = \zeta$.*

There exist a finite ordered set

$$A_{\mathrm{th}} = \{q_0, \dots, q_{R-1}, n\} \subseteq \{2, \dots, n\}, \qquad |A_{\mathrm{th}}| = O\left(\frac{\log n}{\varepsilon}\right),$$

and a deterministic nondecreasing function

$$f^{\mathrm{th}} : \mathbb{R} \to A_{\mathrm{th}}$$

such that, for every fixed (η,ρ)-clusterable sequence X of length $n' \le n$, with $2 \le k(X) \le n'$,

$$\Pr_W\big[k(X) \le f^{\mathrm{th}}(S_W(X)) \le (1+\varepsilon)k(X)\big] \ge 1-\zeta.$$

Both A_{th} and f^{th} depend only on n and ε and the deterministic Gaussian-maxima bounds used in Theorem B.9; they do not depend on X, the distribution of X, or the realized sketch W.

Proof. Take the grid values and thresholds constructed in Theorem B.9, and define

$$f^{\mathrm{th}}(s) = \begin{cases} q_{\min\{r : s \le \theta_r\}}, & \text{if this set is nonempty,} \\ n, & \text{otherwise.} \end{cases}$$

Theorem B.9 proves the approximation guarantee for sequences of length n. For a sequence of length $n' \le n$ the same proof applies with the same thresholds: Step 1 still holds because $\sqrt{2\eta \log(2n')} \le \tau_n$, Lemma B.8 because the bounds on ρ and η in terms of n imply the same bounds in terms of n', and Steps 2 and 4–6 depend only on the grid and on $k \le n$. Theorem B.9 also shows that the thresholds are nondecreasing, hence f^{th} is nondecreasing. The bound on the range size follows from $R = O(\log n/\varepsilon)$. □

Corollary B.11 (Explicit form of Corollary 4.7)**.** *Under the conditions of Theorem 4.4, let $\mathcal{D}$ be any distribution over (η, ρ)-clusterable sequences of length at most n with at least two centers. Then there exist fixed $w_1, \dots, w_m$ such that MaxSketch achieves a $(1+\varepsilon)$-multiplicative approximation with probability at least $1 - \delta$ over a fresh draw $X \sim \mathcal{D}$.*

Proof. Let $\mathcal{D}$ be a distribution as in the statement and let $X \sim \mathcal{D}$ be a sequence with an unknown number of centers k.

Let $W = (w_1, \dots, w_m)$, where $w_j \sim \mathcal{N}(0, I_d)$ independently. Define $S(X; W) := \frac{1}{m} \sum_{j=1}^{m} \max_{x \in X} \langle w_j, x \rangle$. Let $r = \min\{s \in \{0, \dots, R-1\} : S(X; W) \le \theta_s\}$ when this set is nonempty, and define

$$\hat{k}(X; W) := \begin{cases} q_r, & \text{if such an } r \text{ exists,} \\ n, & \text{otherwise.} \end{cases}$$

Now we define the failure indicator

$$F(X, W) := \mathbb{1}\left\{ \hat{k}(X; W) \notin [k, (1+\varepsilon)k] \right\} \tag{22}$$

From Corollary B.10, applied with $\zeta = \delta$, every (η, ρ)-clusterable sequence X of length at most n with at least two centers satisfies

$$\Pr_W\big(F(X, W) = 1\big) \le \delta.$$

Moreover, we can take the *expectation over the randomness of $X \sim \mathcal{D}$* to bound the average failure probability across sequences. That is, defining

$$g(W) := \mathbb{E}_X[F(X, W)] = \Pr_{X \sim \mathcal{D}}\left[\hat{k}(X; W) \notin [k, (1+\varepsilon)k]\right],$$

exchanging the order of expectation by Tonelli's theorem we obtain the average probability of failure:

$$\mathbb{E}_W[g(W)] = \mathbb{E}_W \mathbb{E}_X[F(X, W)] = \mathbb{E}_X \Pr_W[F(X, W) = 1] \le \delta. \tag{23}$$

Since $\mathbb{E}_W[g(W)] \le \delta$, by the probabilistic method there exists a deterministic $W^\star$ such that $g(W^\star) \le \delta$, i.e., $\Pr_{X \sim \mathcal{D}}[\hat{k}(X; W^\star) \notin [k, (1+\varepsilon)k]] \le \delta$, which completes the proof. □

B.4 Proof of Theorem 4.8

Theorem B.12 (Theorem 4.8 (explicit statement))**.** *Fix $n \geq 2$, $\varepsilon \in (0, 1/2]$, and $\delta \in (0, 1)$. Let $\mathcal{D}$ be a distribution over pairs (X, k), where X is an (η, ρ)-clusterable sequence of length at most n, and $k \in \{2, \dots, n\}$ is its number of distinct latent centers. Assume that η and ρ satisfy the conditions of Theorem 4.4.*

Let m be an integer satisfying $m \geq C_0 \frac{\log(128/\delta^2)}{\varepsilon^2}$, where $C_0 > 0$ is a sufficiently large absolute constant. Draw $w_1, \dots, w_m \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_d)$ and define

$$S_W(X) = \frac{1}{m} \sum_{j=1}^{m} \max_{x \in X} \langle w_j, x \rangle.$$

Let f^{th} and its finite output range $A := \text{range}(f^{\text{th}}) = \{a_1 < \cdots < a_T\}$ be as in Corollary B.10, applied with failure probability $\delta^2/64$, with $T = O\Big(\frac{\log n}{\varepsilon}\Big)$.

Given N i.i.d. training samples $(X_1, k_1), \dots, (X_N, k_N) \sim \mathcal{D}$, define

$$\widehat{R}_W(f) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\left\{ f(S_W(X_i)) < k_i \text{ or } f(S_W(X_i)) > (1+\varepsilon) k_i \right\}.$$

Let $\mathcal{H}_T$ be the class of all nondecreasing functions, $f : \mathbb{R} \to A$ and let $\widehat{f} \in \mathcal{H}_T$ be an empirical risk minimizer:

$$\widehat{f} \in \arg\min_{f \in \mathcal{H}_T} \widehat{R}_W(f).$$

There exists a universal constant $C > 0$ such that if

$$N \geq \frac{C}{\delta^2} \left[T \log\left(\frac{T}{\delta}\right) + \log\left(\frac{1}{\delta}\right) \right],$$

then

$$\Pr\left[k \leq \widehat{f}(S_W(X)) \leq (1+\varepsilon) k \right] \geq 1 - \delta,$$

where the probability is over the random sketch W, the training sample, and an independent test sample $(X, k) \sim \mathcal{D}$.

Moreover, after the values $S_W(X_1), \dots, S_W(X_N)$ have been computed, $\widehat{f}$ can be found in time $O(N \log N + NT)$.

Proof. For a fixed sketch W and a readout $f \in \mathcal{H}_T$, define the population risk

$$R_W(f) = \Pr_{(X,k) \sim \mathcal{D}} \left[f(S_W(X)) < k \text{ or } f(S_W(X)) > (1+\varepsilon) k \right].$$

By Corollary B.10, applied with failure probability $\delta^2/64$, there exists a deterministic monotone readout $f^{\text{th}} \in \mathcal{H}_T$ such that, for every fixed (η, ρ)-clusterable sequence X with count k,

$$\Pr_W \left[f^{\text{th}}(S_W(X)) < k \text{ or } f^{\text{th}}(S_W(X)) > (1+\varepsilon) k \right] \leq \frac{\delta^2}{64}.$$

Taking expectation over $(X, k) \sim \mathcal{D}$ and using Fubini's theorem gives $\mathbb{E}_W R_W(f^{\text{th}}) \leq \frac{\delta^2}{64}$. Therefore, by Markov's inequality, $\Pr_W \left[R_W(f^{\text{th}}) > \frac{\delta}{8} \right] \leq \frac{\delta}{8}$. Hence, with probability at least $1 - \delta/8$ over the draw of W,

$$R_W(f^{\text{th}}) \leq \frac{\delta}{8}.$$

Conditioned on this event, f^{th} is a low population risk hypothesis that is contained in $\mathcal{H}_T$. For the remainder of the proof we have to show that ERM recovers a hypothesis with nearly the same population risk. So from now on, we consider the sketch W fixed.

We next prove uniform convergence over $\mathcal{H}_T$. For each $f \in \mathcal{H}_T$, define the binary loss

$$\ell_f(X,k) = \mathbf{1}\left\{f(S_W(X)) < k \text{ or } f(S_W(X)) > (1+\varepsilon)k\right\}.$$

Since $f^{\mathrm{th}} \in \mathcal{H}_T$, it suffices to show that empirical risk minimization over $\mathcal{H}_T$ generalizes. We therefore bound the complexity of $\mathcal{H}_T$ via its growth function.

Consider any M examples and sort them so that

$$S_W(X_1) \le S_W(X_2) \le \cdots \le S_W(X_M).$$

Since f is nondecreasing and takes values in $A = \{a_1, \ldots, a_T\}$, its predictions on the sorted examples are of the form $a_{j_1}, a_{j_2}, \ldots, a_{j_M}, 1 \le j_1 \le j_2 \le \cdots \le j_M \le T$. The number of such nondecreasing sequences is $\binom{M+T-1}{T-1}$. Thus the growth function of the induced binary loss class is at most

$$\Pi(M) \le \binom{M+T-1}{T-1} \le \left(\frac{e(M+T)}{T}\right)^T.$$

A standard growth-function uniform convergence bound [22] implies that there is universal constant $C_1 > 0$ such that, with probability at least $1 - \delta/8$ over the training sample,

$$\sup_{f \in \mathcal{H}_T} \left|R_W(f) - \widehat{R}_W(f)\right| \le C_1 \sqrt{\frac{T \log\left(\frac{e(N+T)}{T}\right) + \log(8/\delta)}{N}}.$$

By choosing the universal constant C in the theorem sufficiently large, the assumed lower bound on N ensures that

$$\sup_{f \in \mathcal{H}_T} \left|R_W(f) - \widehat{R}_W(f)\right| \le \frac{\delta}{8}.$$

On this event, and using the fact that $\widehat{f}$ is the empirical risk minimizer,

$$\begin{aligned} R_W(\widehat{f}) &\le \widehat{R}_W(\widehat{f}) + \frac{\delta}{8} \\ &\le \widehat{R}_W(f^{\mathrm{th}}) + \frac{\delta}{8} \\ &\le R_W(f^{\mathrm{th}}) + \frac{\delta}{4} \\ &\le \frac{3\delta}{8}. \end{aligned}$$

The good-sketch event fails with probability at most $\delta/8$, and the uniform-convergence event fails with probability at most $\delta/8$. Therefore, by the union bound, with probability at least $1 - \delta/4$ over the sketch and the training sample,

$$R_W(\widehat{f}) \le \frac{3\delta}{8}.$$

Finally, draw an independent test sample $(X,k) \sim \mathcal{D}$. Conditional on the good sketch-and-training event, the probability that $\widehat{f}$ fails on this test sample is exactly $R_W(\widehat{f})$, which is at most $3\delta/8$. Hence the total failure probability is at most

$$\frac{\delta}{4} + \frac{3\delta}{8} = \frac{5\delta}{8} \le \delta$$

Thus

$$\Pr\left[k \le \widehat{f}(S_W(X)) \le (1+\varepsilon)k\right] \ge 1 - \delta.$$

It remains only to justify the claimed running time. Compute the scores $s_i = S_W(X_i)$ and sort them, which takes $O(N \log N)$ time. For every sorted example i and output value $a_j \in A$, define

$$c_{i,j} = \mathbf{1}\left\{a_j < k_i \text{ or } a_j > (1+\varepsilon)k_i\right\}.$$

Empirical risk minimization over nondecreasing readouts is equivalent to finding indices

$$1 \le j_1 \le j_2 \le \cdots \le j_N \le T$$

minimizing $\sum_{i=1}^{N} c_{i,j_i}$. This is solved by dynamic programming:

$$D(i,j) = c_{i,j} + \min_{q \le j} D(i-1,q).$$

Using prefix minima, all values $D(i,j)$ can be computed in $O(NT)$ time. The recovered sequence of indices defines an empirical risk minimizer $\widehat{f} \in \mathcal{H}_T$. This proves the theorem. □

B.5 Lower bound for Gaussian-max observations

Proposition B.13 (Explicit form of Proposition 4.6)**.** *Fix $\varepsilon \in (0, 1/4]$ and $\delta \in (0, 1/8]$. Let $k^\star$ be an unknown integer. Suppose an estimator observes only m independent samples from the distribution of $M_{k^\star} = \max_{1 \le r \le k^\star} Z_r$, where $Z_r \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$, and must output $\widehat{k}$ with $k^\star \le \widehat{k} \le (1+\varepsilon)k^\star$ with probability at least $1-\delta$ for every $k^\star \ge \lceil 1/\varepsilon \rceil$. Then $m = \Omega(\varepsilon^{-2} \log(1/\delta))$.*

Proof. For an integer $h \ge 1$, let P_h be the distribution of $M_h = \max_{1 \le r \le h} Z_r$ for independent standard normal variables Z_r. Write Φ for the standard normal distribution function. The random variable $Y_h := -\log \Phi(M_h)$ has an exponential distribution with rate h. To see this, for $y > 0$,

$$\Pr(Y_h > y) = \Pr\big(M_h < \Phi^{-1}(e^{-y})\big) = \Phi\big(\Phi^{-1}(e^{-y})\big)^h = e^{-hy}.$$

The transformation from M_h to Y_h is bijective, so it preserves the difficulty of testing two possible values of h.

Fix integers $k < k'$ and set $\gamma := k'/k - 1 \in (0, 1]$. The Hellinger affinity between exponential distributions with rates k and k' is

$$A(P_k, P_{k'}) = \int_0^\infty \sqrt{k e^{-ky}\, k' e^{-k'y}}\, dy = \frac{2\sqrt{kk'}}{k+k'}.$$

Affinity tensorizes over independent observations. Setting $a := A(P_k, P_{k'})^2 = \frac{4kk'}{(k+k')^2}$, the affinity between $P_k^{\otimes m}$ and $P_{k'}^{\otimes m}$ has square a^m. The total variation distance satisfies

$$\mathrm{TV}\big(P_k^{\otimes m}, P_{k'}^{\otimes m}\big) \le \sqrt{1-a^m}.$$

The sum of the error probabilities of any test between these two distributions is at least one minus their total variation distance. If both errors are at most δ, then

$$2\delta \ge 1 - \sqrt{1-a^m} \ge \frac{a^m}{2}, \qquad \text{hence} \qquad m \ge \frac{\log(1/(4\delta))}{-\log a}.$$

Since $a = \frac{4(1+\gamma)}{(2+\gamma)^2} = 1 - \frac{\gamma^2}{(2+\gamma)^2}$, we have, for $0 < \gamma \le 1$, $-\log a \le \frac{\gamma^2/(2+\gamma)^2}{1-\gamma^2/(2+\gamma)^2} \le \frac{9}{32}\gamma^2$. Consequently,

$$m \ge \frac{32}{9\gamma^2} \log\left(\frac{1}{4\delta}\right).$$

Now choose $k := \lceil 1/\varepsilon \rceil$ and $k' := \lceil (1+2\varepsilon)k \rceil$. Their one-sided approximation intervals are disjoint because $k' > (1+\varepsilon)k$. Moreover,

$$2\varepsilon \le \gamma = \frac{k'-k}{k} \le 2\varepsilon + \frac{1}{k} \le 3\varepsilon \le \frac{3}{4}.$$

Any estimator that succeeds at both k and k' yields a test: decide k' if its output is at least k', and decide k otherwise. Both testing errors are at most δ. Since $\delta \le 1/8$ gives $\log(1/(4\delta)) \ge \frac{1}{3}\log(1/\delta)$, the preceding bound implies

$$m \ge \frac{32}{243\varepsilon^2} \log\left(\frac{1}{\delta}\right).$$

To see that this observation model occurs within our clusterability regime, take $d, n \geq k'$ and, under either hypothesis $h \in \{k, k'\}$, form a length-n stream containing each of the orthogonal unit vectors $e_1, \ldots, e_h$ at least once, padding with exact copies of e_1. For independent $w_j \sim \mathcal{N}(0, I_d)$, the jth register is

$$\max_{x \in X_h} \langle w_j, x \rangle = \max_{1 \leq r \leq h} (w_j)_r \sim P_h,$$

independently across j. These streams have zero center correlation and zero within-cluster perturbation. The lower bound concerns the resulting register values alone; it does not give the estimator access to the vectors w_j. □

C Experimental Details

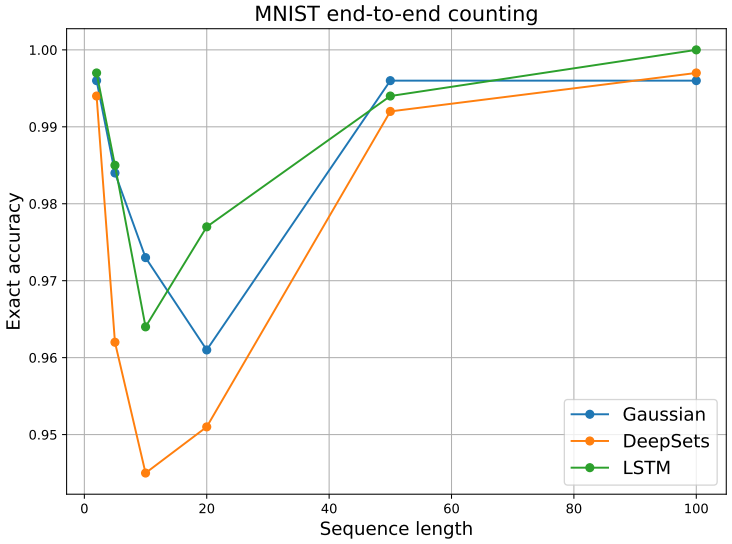

Figure 2: Exact counting accuracy of three baseline models, MaxSketch (Gaussian), DeepSets, and LSTM, on MNIST sequences of varying lengths. All models achieve near-perfect accuracy across all sequence lengths, and the ± 1 accuracy reaches 100%, indicating that predictions are at most one count away from the true number of digits.

C.1 MNIST

Dataset and preprocessing. We use the standard MNIST dataset consisting of 60,000 training images and 10,000 test images. All images are scaled to the range $[0, 1]$ and standardized using the training set mean and standard deviation. No data augmentation is applied.

Encoder. We use a small convolutional neural network composed of two convolutional blocks. The first block applies a 3×3 convolution with 32 channels, followed by batch normalization, a ReLU activation, and 2×2 max pooling. The second block applies a 3×3 convolution with 64 channels, again followed by batch normalization, ReLU, and 2×2 max pooling. The resulting feature maps are flattened and passed through a linear layer producing an embedding of dimension $d = 256$. All embeddings are L2-normalized before being passed to the aggregation module.

Aggregators. We evaluate three aggregation mechanisms. MaxSketch uses $m = 4{,}096$ fixed random Gaussian projections and computes the maximum projection along each direction. These statistics are then passed through a two-layer MLP with hidden width 256 and ReLU activation to produce the final count prediction. DeepSets applies a two-layer MLP with hidden width 256 and output width 256 with ReLU activation to each embedding individually, aggregates representations via summation, and then applies a final two-layer MLP with hidden width 256 and ReLU activation to predict the count. The LSTM baseline processes the sequence of embeddings using a single-layer LSTM with hidden size 256, and the final hidden state is mapped to the count via a linear layer.

Training. All models are trained using the Adam optimizer with learning rate 0.001 and batch size 32 sequences. The loss function is mean squared error between predicted and true counts. Training is performed for 50 epochs, with evaluation after each epoch.

Stream construction. Each training sequence is generated by first sampling $k \sim \mathrm{Uniform}\{1, \ldots, 10\}$. We then sample k classes uniformly without replacement, and construct a sequence of length n by repeatedly sampling a class uniformly from the chosen set and then sampling an image uniformly from that class. Each sequence is resampled independently at every epoch. Test sequences are constructed using the same procedure but with test images.

Experiments. We evaluate accuracy across sequence lengths by training and testing at each $n \in \{2, 5, 10, 20, 50, 100\}$ independently. For extrapolation, MaxSketch is trained at $n_{\text{train}} \in \{10, 20, 50\}$ and tested at $n_{\text{test}} \in \{150, 250, 500\}$. In addition, we compute 1-nearest-neighbour accuracy using cosine distance to the nearest training embedding.

C.2 CIFAR-10

Dataset. Standard CIFAR-10 split (50,000 training, 10,000 test images). We hold out a random 10% of the training images for validation, which is used only to choose the number of principal directions. No data augmentation is used.

Encoders. The publicly released DINOv2 ViT-L/14 and ViT-B/14 models (`timm`: `vit_large_patch14_dinov2.lvd142m`, `vit_base_patch14_dinov2.lvd142m`), trained by self-distillation without labels. Images are resized to 224×224 (bicubic) and normalised with the ImageNet statistics; the embedding is the class-token output. The encoder is frozen and never sees CIFAR-10 labels.

Label-free preprocessing. On the unlabeled training embeddings we compute the mean and the top principal directions. Each embedding is centred, projected onto the top q directions and L2-normalised; test embeddings use the same map. We use $q = 64$ for DINOv2-L with the MLP readout and $q = 16$ for DINOv2-B and for the isotonic readout, chosen on validation streams from $q \in \{16, 64\}$ (MLP) and $\{16, 64, 128\}$ plus raw and centred-only embeddings (isotonic).

MaxSketch. $m = 4{,}096$ fixed Gaussian projections drawn from $\mathcal{N}(0, I_q)$, redrawn for each seed.

MLP readout. LayerNorm on the m registers, followed by a two-layer MLP (hidden width 512, GELU, dropout 0.1) with a softmax over $k \in \{1, \ldots, 10\}$, trained with cross-entropy on count labels only. We use AdamW (learning rate 10^{-3}, weight decay 10^{-4}) with a one-cycle schedule for 8,000 steps. Each step draws a length $n \in \{2, 3, 5, 10, 20, 50, 100\}$ and a batch of 512 training-image streams. A single model is used for every test length.

Isotonic readout. For each test length n, we fit `scikit-learn`'s `IsotonicRegression` from $S(X)$ to k on 4,000 training-image streams of length n and round its output to the nearest integer in $\{1, \ldots, 10\}$. The calibration depends on n, as do the thresholds of the estimator in Theorem 4.4.

Stream construction. We sample $k \sim \mathrm{Uniform}\{1, \ldots, 10\}$, sample k classes without replacement, and draw n images by repeatedly picking one of the k classes uniformly and then an image uniformly from that class; the target is the number of distinct classes in the stream. Training and calibration streams use training images, and test streams use test images. We evaluate 2,000 test streams per length $n \in \{10, 20, 50, 100, 250, 500\}$ and report the mean over three seeds (projections, streams and readout initialisation).

C.3 Faces (CelebA)

Dataset. We use CelebA [17] (202,599 images, 10,177 identities). We keep identities with at least two images and remove identities whose mean embeddings have cosine similarity above 0.85 with another identity's, which likely denote the same person under two labels. The remaining $9{,}868$ identities ($197{,}063$ images) are split uniformly at random by identity into a calibration set and an evaluation set of $4{,}934$ identities each.

Embeddings. Faces are detected and aligned with MTCNN [28]; for the 310 images with no detected face we use a centre crop. Each 160×160 face crop is embedded with an Inception-ResNet-v1 FaceNet encoder [21] pretrained on VGGFace2 [1] (the `facenet-pytorch` implementation), giving 512-dimensional embeddings. Embeddings are centred by the mean of the calibration identities and L2-normalized. The encoder is frozen throughout.

Streams. A stream with k identities samples k identities without replacement, includes each at least once, and fills the remaining positions of a length-$n = 20{,}000$ stream by sampling an identity uniformly and then one of its images uniformly. Evaluation uses $k \in \{100, 200, \ldots, 4900\}$ with 100 streams per k. Calibration streams use 200 geometrically spaced values of k between 80 and 4,934, with 10 streams each.

MaxSketch. $m = 4{,}096$ projections drawn from $\mathcal{N}(0, I_{512})$. The readout is an isotonic regression of $\log k$ on $S(X)$ (scikit-learn), fit on the calibration streams. Results are averaged over three independent draws of the projections.

Zhang baseline. We implement Algorithm 1 of [29] with Euclidean distance on the normalized embeddings and $\varepsilon = 0.1$. With the paper's constants and $n = 20{,}000$ the algorithm never subsamples, so it keeps one representative per group. The threshold $\alpha \in \{0.6, 0.7, \ldots, 1.4\}$ is chosen to minimize mean relative error on 30 calibration streams ($\alpha = 1.1$), and the algorithm is evaluated on 5 streams per k.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer:[Yes]

   Justification: The abstract and introduction accurately reflect the paper's theoretical contributions, assumptions, and scope. In particular, they clearly state the clusterability assumptions, the Gaussian sketching framework, and the multiplicative approximation guarantees proved in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the main limitations and assumptions of the approach, including the (η, ρ)-clusterability assumption, and what happens when it is violated. It also covers the gap between the theory, which uses fixed projections and a monotone readout of $S(X)$, and the MLP readout used in the experiments.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All lemmas, corollaries, and theorems are stated with explicit assumptions, and complete proofs are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides sufficient implementation and experimental details, including dataset construction, model architecture, training setup, and evaluation protocol, to reproduce the main empirical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The main contribution of the paper is theoretical. While code and data are not currently released, the paper provides sufficient algorithmic, mathematical, and experimental details to reproduce the proposed method and empirical evaluations.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the full experimental setup, including data construction, model architecture, hyperparameters, optimization procedure, and evaluation protocol, with additional implementation details provided in the appendix sufficient to understand and interpret the reported results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: The experiments are reported based on multiple runs of the proposed method over random draws of Gaussian projections and dataset samples.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The paper does not provide detailed information on compute resources such as hardware specifications, memory usage, or exact runtime per experiment, as the experiments are lightweight and primarily involve sampling Gaussian projections and computing simple statistics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses only synthetic or standard benchmark-style data and mathematical modeling, does not involve human subjects or personal data, and does not raise known ethical concerns related to privacy, safety, discrimination, or misuse, thus conforming to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The work is primarily theoretical in nature, focusing on Gaussian sketching and statistical guarantees for cluster estimation, and does not involve deployment-oriented systems or direct applications; therefore, broader societal impacts are minimal and not explicitly discussed.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No safeguards are described, as the paper does not release high-risk data or models and is primarily theoretical in nature.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and models are cited in the references.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets are released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.