SKETCHING FASTER THAN DIMENSION TIMES UPDATE TIME

Anonymous authors
Paper under double-blind review

ABSTRACT

In many applications of sketching, such as distributed computation, the input is available as a fixed vector or matrix rather than as a stream of updates. Let d denote the ambient input dimension. In this offline setting, applying a streaming sketch by simulating coordinate updates can introduce unnecessary overhead: Larsen et al. (2015) prove $\omega(1)$ update-time lower bounds for several natural turnstile streaming problems. In addition, achieving high success probability, say $1 - 1/d$, is often done by repetitions that introduce logarithmic overheads. We ask whether, given full access to a vector or matrix, one can apply an optimal-size linear sketch in truly linear time while retaining this high-success-probability regime. We show that this is possible for a range of basic sketching problems by composing standard sketching primitives in new ways. For ℓ_2 estimation and ℓ_2 heavy hitters, we give sketches with optimal sketching dimension, success probability at least $1 - 1/d$, and truly linear application time. We further combine this construction with ExpanderSketch to obtain fast decoding, and extend the same ideas to rank-one tensor sketches and ℓ_2 sampling. We also obtain fast sketches for general ℓ_p estimation. For $p > 2$, we use our heavy-hitters primitive to obtain a linear-time F_p moment sketch with near-optimal $\widetilde{O}(d^{1-2/p})$ dependence on dimension. Finally, we apply the same composition-and-selection template to linear regression and low-rank approximation. In the natural tall and wide regimes, respectively, these sketches have optimal or near-optimal sketch size and can be applied in input-sparsity time. Since all of our sketches are linear, the results apply directly in the coordinator model for distributed computation.

1 INTRODUCTION

As data becomes larger, it becomes increasingly important to develop algorithms that scale to massive datasets. One approach for dealing with such big data is to compress it into a smaller data structure from which useful statistics can still be computed. This is the central idea of sketching. Formally, a (linear) sketch takes as input a data vector $v \in \mathbb{R}^d$, and compresses it via a linear map $S \in \mathbb{R}^{m \times d}$, to produce a vector $Sv \in \mathbb{R}^m$. Typically the goal is for m to be dramatically smaller than d, so that storing Sv is much more efficient than storing v itself.

Sketches have found important applications: for example, a fast sketch can be applied as a preprocessing step to reduce the size of a large-scale optimization or machine learning problem so that the resulting problem can be solved more quickly (Woodruff et al., 2014). In distributed and federated learning, workers compress locally computed gradients or model updates before communicating them (Vogels et al., 2019; Rothchild et al., 2020). In these settings, the vector to be compressed is already available locally, and both the size of the compressed representation and the time required to compute it matter.

More relevant to our setting, sketches have also found applications in improving the *communication complexity* of various problems (Kannan et al., 2014; Woodruff & Zhang, 2013; Huang et al., 2021). A typical setup is one in which the data is partitioned among two or more servers and the goal is to compute some quantity related to the underlying data, while minimizing the total communication among the servers (Roughgarden et al., 2016). Although communication complexity is often the primary concern, the computational load on local servers is also a critical factor, especially for massive datasets. Sketches are useful here as the servers may first compress their data before sending it over

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

the network, to reduce communication. When the sketches are linear, they can then be aggregated and decoded by a "coordinator" who receives all the sketches.

On the other hand, historically, sketching was developed as a technique in the *streaming* setting where the goal is to process a huge number of updates. Here the model is that there is an underlying vector $v \in \mathbb{R}^d$ that is updated coordinate-wise. That is, one receives updates u_i, each with a single nonzero entry, that modify v by $v \leftarrow v + u_i$. When the updates u_i are unrestricted, this is typically referred to as turnstile streaming, in contrast with streams that only allow for non-negative updates.

In the streaming setting one is usually most interested in the space requirements, but other features such as update time are also important. To maintain a linear sketch Sv of v, one needs to compute Su_i for each update. Since the updates may be numerous, it is often important to design S so that Su_i can be efficiently calculated, while maintaining low storage requirements (Alman & Yu, 2020; Kane et al., 2011; Chou et al., 2019).

While the sketches used for streaming algorithms and communication are often similar, the specific features that we want are different in these settings. Suppose one wants to apply a sketch S, which has been optimized for a streaming algorithm, in a communication setting where the full input vector v is available offline. If $v \in \mathbb{R}^d$ is the vector to be sketched by one of the servers, then the server could just feed the streaming updates one coordinate of v at a time into the sketch, resulting in a runtime of $d \cdot (\text{update time})$.

Simulating coordinate updates is sometimes optimal. For instance if S is a CountSketch (Charikar et al., 2002), then each update requires $O(1)$ time, resulting in $O(d)$ time to form the sketch. This is not always the case. For example, if one is interested in an ℓ_2 heavy-hitters sketch with the optimal $O(\varepsilon^{-2} \log d)$ sketching dimension and failure probability $1/\text{poly}(d)$, then all known sketches require $\omega(1)$ update time. The same is true for related problems, such as ℓ_p estimation. In fact, Larsen et al. (2015) even show that such problems require $\omega(1)$ worst-case update time for turnstile streaming algorithms.[1]

In our setting, we have access to v directly, and so there is no reason to use the streaming updates to compute the sketch of v. In this situation, a natural question is: when given full access to a vector, is it possible to apply a sketch in truly linear time without logarithmic factors, while simultaneously achieving an optimal sketch size and high success probability, such as $1 - 1/d$? More broadly, are there other situations where sketches can be applied faster than simulating the execution of a turnstile streaming algorithm?

We show that such sketches are possible in a variety of situations by composing existing sketching techniques. We start by addressing the fundamental heavy-hitters problem, where we give a sketch that can be applied quickly, and for which the heavy hitters can be recovered quickly. We use these algorithms as a stepping stone toward ℓ_p estimation and related applications, as well as for solving least-squares regression and low-rank approximation, which are core problems in machine learning. As a consequence of our heavy-hitters algorithm, we also observe that one can perform ℓ_2 sampling with similar guarantees.

2 Problem Formulation and Preliminaries

We now make the offline setting precise. Throughout, our goal is to construct linear sketches with optimal sketch size and high success probability, while keeping the time to apply the sketch truly linear in the input size. This is particularly useful in distributed settings where data is split across several servers as vectors or matrices $X_1, \ldots, X_s$, and the desired quantity depends on their sum $\sum_i X_i$. If S is linear, each server can send SX_i to a coordinator, who aggregates these messages into $S\sum_i X_i$. Thus optimal sketch size gives low communication, while truly linear application time keeps the local computation at each server small. In particular, this avoids both the $d \cdot (\text{update time})$ overhead of simulating a turnstile streaming algorithm, whose update time can often be $\omega(1)$, and the logarithmic overheads usually introduced when boosting a sketch to high success probability.

We next collect the problem formulations considered in the paper. Each problem includes a failure probability parameter δ; when δ is omitted, we take $\delta = 1/d$ by default.

[1] Under the assumption that the sketch is non-adaptive, and in the so-called "cell probe model" (see Larsen et al., 2015).

ℓ_p estimation. For the $(\varepsilon, \delta, \ell_p)$-estimation problem (also referred to as the F_p-moment estimation problem in the streaming literature), the goal is to construct a linear sketch S that can be applied to $x \in \mathbb{R}^d$, from which one can recover an estimate M satisfying $(1-\varepsilon)\|x\|_p \le M \le (1+\varepsilon)\|x\|_p$ with probability at least $1-\delta$.

Heavy hitters. For the $(\varepsilon, \delta, \ell_p)$-heavy-hitter problem, the goal is to construct a linear sketch S that can be applied to $x \in \mathbb{R}^d$, from which one can recover a set $\mathcal{H}$ containing every coordinate i with $|x_i| \ge \varepsilon \|x\|_p$, while keeping $|\mathcal{H}| = O(1/\varepsilon^p)$, with probability at least $1-\delta$. Our main heavy-hitter results focus on $p = 2$, and then serve as a primitive for later applications.

Tensor sketches. We first fix notation. For vectors $x_1, \dots, x_q \in \mathbb{R}^d$, write $x = x_1 \otimes \cdots \otimes x_q \in (\mathbb{R}^d)^{\otimes q} \cong \mathbb{R}^{d^q}$ for their tensor product. We use the same notation $A \otimes B$ for the Kronecker product of matrices, and repeatedly use the identity $(A \otimes B)(x \otimes y) = Ax \otimes By$. We index a q-mode tensor by a multi-index $i = (i_1, \dots, i_q) \in [d]^q$. For the rank-one tensor $x = x_1 \otimes \cdots \otimes x_q$, this means $x_i = x_{i_1,\dots,i_q} = (x_1)_{i_1} \cdots (x_q)_{i_q}$.

For the (ε, δ)-distortion tensor sketching problem, the goal is to construct a linear sketch S that can be applied to a rank-one tensor $x = x_1 \otimes \cdots \otimes x_q$, given through its factor vectors, from which one can estimate $\|x\|_2$ and answer point queries with probability at least $1-\delta$. Answering a point query means that, given a multi-index $i \in [d]^q$, the decoder outputs an estimate $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \le \varepsilon \|x\|_2$.

Regression. Given $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$, the goal of the (approximate) regression problem is to construct a linear sketch S that can be applied to A and b on the left, and from which one can recover a vector $\hat{x}$ satisfying $\|A\hat{x} - b\|_2^2 \le (1+\varepsilon)\min_x \|Ax - b\|_2^2$ with probability at least $1-\delta$. We call this the (ε, δ)-regression problem. We are mainly interested in the highly overdetermined regime where n is much larger than d.

Low-rank approximation. For the (ε, δ, k) low-rank approximation problem, the goal is to construct a linear sketch S that can be applied to $A \in \mathbb{R}^{d \times n}$, from which one can recover a rank-k orthogonal projection Π satisfying $\|\Pi A - A\|_F^2 \le (1+\varepsilon)\|A_k - A\|_F^2$ with probability at least $1-\delta$, where A_k is the optimal rank-k approximation to A. We mainly focus on the wide-matrix regime $d \ll n$.

2.1 Additional Related Work

Our results are also motivated by distributed applications of sketching. For example, Ivkin et al. (2019) use heavy-hitters sketches to compress gradients and reduce the communication cost of SGD. This setting is well aligned with our perspective: local gradient contributions are explicitly available on each server, rather than only through a stream, and the guarantees benefit from sketches with small failure probability. Another related example is the kernel classification setting of Mahankali & Woodruff (2021), where polynomial kernels naturally produce tensor-structured vectors that can be sketched using our tensor sketching ideas.

For the heavy-hitters problem, our fast-decoding construction builds on the ExpanderSketch of Larsen et al. (2019), which gives nearly optimal space together with fast recovery. Heavy hitters have a long history, going back to CountSketch (Charikar et al., 2002), and have also been studied in distributed functional monitoring (Huang et al., 2012; Cormode et al., 2011; Woodruff & Zhang, 2012). In particular, Woodruff & Zhang (2012) study distributed heavy hitters in a model where the parties must maintain an estimate of the answer over time; the usual one-shot communication setting can be viewed as a special case.

The Misra-Gries algorithm (Misra & Gries, 1982) gives another classical approach to heavy hitters, but it applies to nonnegative updates. Our setting allows negative entries, as in the turnstile model, so linear sketches such as CountSketch and ExpanderSketch are the relevant primitives.

Finally, our reshaping argument is related in spirit to the use of fast matrix multiplication in Alman & Yu (2020), where updates are batched to speed up sketch maintenance in a stream. In our offline setting the input vector is available all at once, so the matrix multiplication speedup can be applied more directly. We also mention that Kacham et al. (2023) slightly improve the update time of the sketch of Kane et al. (2011), using a pseudorandom generator to save a $\log\log(1/\varepsilon)$ factor.

3 OUR RESULTS

In this section, We introduce a framework for boosting sketch performance, with our detailed results as follows. Proofs of the lemmas and theorems not proved in this section are deferred to the appendix.

3.1 F_p MOMENT ESTIMATION

ℓ_2 estimation. One approach to sketch the ℓ_2 norm of a vector $x \in \mathbb{R}^d$ in $O(d)$ time with $1/\text{poly}(d)$ failure probability is to apply Johnson-Lindenstrauss embeddings. An (ε, δ) Johnson-Lindenstrauss embedding $S : \mathbb{R}^d \to \mathbb{R}^m$ preserves the ℓ_2 norm of a fixed vector x within a $(1 \pm \varepsilon)$ factor with failure probability at most δ. A matrix of i.i.d. random signs gives the optimal dimension $m = O(\varepsilon^{-2}\log(1/\delta))$ (Freksen, 2021), but this matrix-vector product takes $O(d \log d/\varepsilon^2)$ time, which is too slow for our purposes. This can be improved by choosing a structured matrix for the JL embedding. For example, if one chooses a random-sign Toeplitz matrix with $O(\frac{\log^2 d}{\varepsilon^2})$ rows (which is known to be a JL embedding; see Freksen & Larsen, 2020), then the matrix-vector multiplication can be carried out in $O(d \log d)$ time (Kailath & Sayed, 1999)[2].

On the other hand, there are sketches that preserve the ℓ_2 norm and can be applied much faster. CountSketch (Charikar et al., 2002) hashes the entries of $x \in \mathbb{R}^d$ into buckets and assigns each entry a random sign:

Definition 3.1 (CountSketch)**.** *A CountSketch matrix $S \in \mathbb{R}^{m\times d}$ is specified by a hash function $h : [d] \to [m]$ and signs $\sigma_1, \ldots, \sigma_d \in \{-1, 1\}$. For each $j \in [d]$, the jth column of S has a single nonzero entry, namely $S_{h(j),j} = \sigma_j$. Equivalently, for every $x \in \mathbb{R}^d$ and bucket $i \in [m]$,*

$$(Sx)_i = \sum_{j:h(j)=i} \sigma_j x_j.$$

Fact 3.2. *Let $S \in \mathbb{R}^{m\times d}$ be a CountSketch matrix as in Definition 3.1, where h is pairwise independent and the signs $\sigma_1, \ldots, \sigma_d$ are 4-wise independent. For any $\varepsilon, \delta \in (0, 1)$ and every fixed $x \in \mathbb{R}^d$, if $m = \Omega(1/(\varepsilon^2\delta))$, then $\|Sx\|_2 = (1 \pm \varepsilon)\|x\|_2$ holds with probability at least $1 - \delta$. Moreover, Sx can be computed in $O(\text{nnz}(x))$ time*[3].

The drawback is that the number of rows depends polynomially on $1/\delta$, rather than logarithmically as in a JL embedding. Thus CountSketch alone cannot simultaneously give optimal sketch size and arbitrary inverse-polynomial failure probability: for instance, achieving failure probability $1/d^2$ would require $m = \Omega(d^2/\varepsilon^2)$ rows, which is larger than the original dimension.

Our approach is simple, but appears to be new, and serves as the starting point for our later results. We would like to apply a CountSketch only as a first-stage dimension reduction. We choose its parameters so that it reduces the dimension slightly while achieving some $1/\text{poly}(d)$ failure probability; for example, one may sketch down to dimension $d^{0.8}$ and obtain failure probability $d^{-0.5}$, as long as $1/\varepsilon$ is not too large. Once the dimension has been reduced, we apply independent constant-failure CountSketches and take the median of their norm estimates. The additional cost is only $d^{0.8}O(\log d) = o(d)$ in this example, while this second stage achieves logarithmic dependence on its failure probability. Finally, by repeating this composed sketch a constant number of times and taking the median estimate, we boost the success probability to arbitrary $1 - 1/\text{poly}(d)$ while keeping the total application time linear in d. We formalize this as follows:

Lemma 3.3. *For $d \geq 2$ and $\varepsilon \in (0, 1)$ satisfying $\varepsilon \geq d^{-0.5+c}$, where $c > 0$ is a constant, there exists an ε-distortion linear ℓ_2 estimation sketch with sketching dimension $O(\varepsilon^{-2}\log d)$ that can be applied to a vector $x \in \mathbb{R}^d$ in $O_c(d)$ time and, for every fixed x, succeeds with probability at least $1 - 1/d$.*

More generally, for any failure probability $\delta \in (0, 1/2]$, distortion parameter $\varepsilon \in (0, 1)$, and positive integer C, there exists an ε-distortion linear ℓ_2 estimation sketch with failure probability at most δ

[2] In fact, one can do slightly better. By decomposing S into d/m blocks of size $m \times m$, each of which is Toeplitz, multiplication by S can be carried out in $O((d/m)m \log m) = O(d \log m)$ time (Freksen, 2021). For $m = O(\log d/\varepsilon^2)$, this would give a runtime of $O(d(\log\log d + \log \frac{1}{\varepsilon}))$.

[3] We work in the word-RAM model, where arithmetic operations on words of size $\log d$ bits can be carried out in constant time. This allows hash functions to be evaluated in linear time, for example by implementing CountSketch with a linear hash function family.

and sketching dimension $O(\varepsilon^{-2}\log(1/\delta))$ *that can be applied in time*

$$O\left(C\,\mathrm{nnz}(x) + \varepsilon^{-2}\delta^{-1/C}\log\frac{1}{\delta}\right).$$

We note that the sketching dimension $O(\varepsilon^{-2}\log(1/\delta))$ matches the standard Johnson–Lindenstrauss dependence on both ε and the failure probability (Johnson et al., 1984; Jayram & Woodruff, 2013). When discussing optimality, this bound should be capped at d, since the identity map is an exact linear sketch. This approach also applies when $\varepsilon = O(d^{-0.5})$, although its stated application-time bound need not be linear in this regime. For $\delta = 1/d$, the dimension bound $\varepsilon^{-2}\log d$ already exceeds d in this regime, so the identity map avoids unnecessary expansion.

F_p moment estimation. We also consider general F_p moment estimation. For $p > 2$, standard F_p moment sketches use subsampling and ℓ_p heavy hitters to estimate the sizes of approximate level sets. The expensive part is that the highest sampling rates are needed to catch the large coordinates. We therefore split the entries into a heavy part and a light part, using a single high-probability coordinate-estimation sketch to identify an approximate heavy set and estimate its contribution.

Independently, we subsample the original vector at rate $q = \Theta(\log^{-3}(2d))$ and apply the F_p sketch of Ganguly (2011, Theorem 12) with internal accuracy $\Theta(\varepsilon q)$. During decoding, we subtract the estimated contribution of the sampled heavy coordinates, rescale the remaining estimate by $1/q$, and add the estimated contribution of the full heavy part. The analysis requires only additive $O(\varepsilon)F_p(x)$ error for the light part, rather than relative accuracy with respect to its own moment.

The crucial property of the cited sketch is its $O(\log d)$ update time, with no polynomial dependence on the inverse accuracy. We repeat only the independently sampled moment sketches $O(\log d)$ times and take the median, while sharing the single high-probability coordinate-estimation sketch. By generating the sampled positions without scanning the full vector separately for each repetition, we keep the total application time linear. As a result, we obtain the following theorem.

Theorem 3.4. *Fix any constants $p > 2$ and $c > 0$, and let $d \geq 2$. For $\varepsilon \in (0,1)$ satisfying $\varepsilon^{1+2/p} \geq d^{-1/p+c}$, there exists a linear F_p-moment estimation sketch with sketching dimension*

$$O_{p,c}\Big(d^{1-2/p}\,\mathrm{poly}(\log d/\varepsilon)\Big)$$

that can be applied in $O_{p,c}(d)$ time to a vector $x \in \mathbb{Z}^d$ with $O(\log d)$-bit coordinates, given random access to its entries. For every fixed such x, the decoder returns an estimate $\widehat{F}_p$ satisfying

$$\left|\widehat{F}_p - F_p(x)\right| \leq \varepsilon F_p(x), \qquad F_p(x) = \sum_{i=1}^{d}|x_i|^p,$$

with probability at least $1 - 1/d$.

For fixed ε, this sketch retains the $\tilde{O}(d^{1-2/p})$ dependence on dimension, which is optimal up to logarithmic factors in the bounded-word model. While the dependence on ε and logarithmic factors can be improved (Ganguly, 2011), our contribution is to combine truly linear offline application time with failure probability $1/\mathrm{poly}(d)$ in the stated parameter regime.

Related work on distributed moment estimation considers non-negative local vectors (see, e.g., Esfandiari et al., 2024, and references therein). Our construction does not require non-negativity: for fixed shared randomness, all stored measurements are linear functions of the input, and the heavy-part corrections are performed only during decoding. Thus the sketches can be aggregated in the distributed setting, with the recovery guarantee applying whenever the aggregate vector satisfies the theorem's input assumptions.

Reshaping and mixed norms. We also consider applying dense sketches to matrix-structured inputs using fast rectangular matrix multiplication. For fixed $1 < r < p < 2$, let $X \in \mathbb{R}^{B\times s}$ have columns $X_1, \ldots, X_s$, and define the mixed norm

$$\mathcal{N}_{p,r}(X) = \left(\sum_{j=1}^{s}\|X_j\|_p^r\right)^{1/r}.$$

Here the matrix shape is part of the definition of the quantity being estimated.

Suppose that $T \in \mathbb{R}^{m\times B}$ is the normalized p-stable map described in Appendix B.2. Applying the same map to every column amounts to computing the matrix product TX, which can be evaluated using fast rectangular matrix multiplication in the appropriate dimension regimes (Coppersmith, 1982). The normalization of T gives

$$\mathrm{E}\,\|TX\|_r^r = \sum_{j=1}^{s} \mathrm{E}\,\|TX_j\|_r^r = \sum_{j=1}^{s} \|X_j\|_p^r = \mathcal{N}_{p,r}(X)^r,$$

where $\|TX\|_r$ denotes the entrywise ℓ_r norm. Moreover, the one-vector moment bound and Minkowski's inequality imply that, for a sufficiently large polynomial number $m = \varepsilon^{-O_{p,r}(1)}$ of rows,

$$\|TX\|_r = (1 \pm \varepsilon)\mathcal{N}_{p,r}(X)$$

with probability at least $9/10$ for every fixed X. This argument does not require independence between the sketched columns.

Thus when $m < B$, the construction reduces the number of rows while approximately preserving a mixed norm. It does not, in general, preserve the entrywise ℓ_p norm

$$\|X\|_p = \left(\sum_{j=1}^{s} \|X_j\|_p^p\right)^{1/p}.$$

Consequently, reshaping an arbitrary vector into X does not by itself give an ℓ_p-norm estimation sketch through this construction.

3.2 Heavy Hitters and Application

ℓ_2 heavy hitters. Again, we start from ℓ_2 norm. As a consequence of the ℓ_2 estimation sketch, we obtain an ℓ_2 heavy-hitters sketch almost for free, using a standard approach. To recover the ε-heavy hitters of x, it suffices to be able to estimate the values of $\langle e_i, x\rangle$ to within $O(\varepsilon\,\|x\|_2)$ additive error, simultaneously for all i. For each i, we can write

$$\langle e_i, x\rangle = \frac{1}{2}\left(\|x + e_i\|_2^2 - \|x\|_2^2 - \|e_i\|_2^2\right),$$

so accurate ℓ_2 norm estimates for x and $x + e_i$ give the desired point estimate. Therefore, taking the ℓ_2 estimation sketch with arbitrary $1/\mathrm{poly}(d)$ failure probability and union bounding over all $i \in [d]$ yields a linear-time, high-probability ℓ_2 heavy-hitters sketch.

Lemma 3.5. *For $\varepsilon \geq d^{-0.5+c}$, where $c > 0$ is a constant, there exists a (ε, ℓ_2)-heavy-hitter sketch with sketching dimension $O(\varepsilon^{-2}\log d)$ that can be applied to $x \in \mathbb{R}^d$ in $O(d)$ time and such that recovery succeeds with probability at least $1 - \frac{1}{d}$.*

Proof. We first apply the ℓ_2 estimation sketch S of Lemma 3.3 with $1/d^2$ failure probability. Since we know S and Sx, it is possible to estimate the $\|x\|_2$ and $\|x + e_i\|_2$ within a $(1+\varepsilon)$ multiplicative factor for each standard basis vector e_i $(1 \leq i \leq d)$. Here, we have the relation

$$\langle x, e_i\rangle = \frac{1}{2}(\|x + e_i\|_2^2 - \|x\|_2^2 - \|e_i\|_2^2).$$

By taking a union bound over the d coordinates, all of these estimates are correct with probability at least $1 - 1/d$, after setting the failure probability of each norm estimate to be sufficiently small. This gives an additive $O(\varepsilon(\|x\|_2^2 + 1))$ estimate of each coordinate, which is not yet the scale-invariant error that we need. To remove this dependence, we first use an independent copy of the ℓ_2 estimation sketch to compute a number Z satisfying $Z = (1 \pm 0.1)\,\|x\|_2$ with probability at least $1 - 1/d$. We then apply the preceding argument to the normalized vector x/Z: the resulting estimates $\widetilde{y}_i$ of the coordinates of $y = x/Z$ have additive error at most $O(\varepsilon)$ simultaneously for all i. After replacing ε by a sufficiently small constant multiple in the construction, rescaling by Z gives estimates $\widetilde{x}_i = Z\widetilde{y}_i$ satisfying

$$|\widetilde{x}_i - x_i| \leq \frac{\varepsilon}{10}\,\|x\|_2$$

for all $i \in [d]$ with probability at least $1 - 1/d$. The recovery algorithm outputs $\mathcal{H} = \{i \in [d] : |\widetilde{x}_i| \geq (\varepsilon/2)Z\}$. Every coordinate with $|x_i| \geq \varepsilon \|x\|_2$ is included in $\mathcal{H}$, while every output coordinate has $|x_i| = \Omega(\varepsilon \|x\|_2)$. Hence $|\mathcal{H}| = O(1/\varepsilon^2)$, and the recovery succeeds with probability at least $1 - 1/d$. □

Fast decoding time for heavy hitters. The drawback of the simple heavy-hitters sketch above is its decoding time: although the sketch can be applied in linear time, recovering the actual heavy coordinates by estimating all d coordinates also takes linear time. We would like to keep the same guarantees – optimal sketch size, $O(d)$ application time, and high success probability – while making the decoding time much smaller.

To do this, we replace the final JL sketch in the previous construction with ExpanderSketch (Larsen et al., 2019). ExpanderSketch is a turnstile heavy-hitters sketch with near-optimal space and fast recovery time: given a sketch of a vector, it can recover the heavy coordinates in $\text{poly}(\log d, \frac{1}{\varepsilon})$ time. Thus, if S is the initial CountSketch used to reduce the dimension and T is an ExpanderSketch, then applying T to Sx lets us quickly recover the heavy coordinates of the sketched vector Sx. We first introduce a slightly modified ExpanderSketch for convenience.

Proposition 3.6. *For any $\varepsilon \in (0, 1)$, there exists a turnstile sketch T over $\mathbb{R}^d$ with update time $O(\log d)$ and sketching dimension $O(\varepsilon^{-2} \log d)$, together with a recovery algorithm that runs in time $O(\varepsilon^{-2} \operatorname{polylog} d)$. Given the sketch Tx, the recovery algorithm outputs a set $\mathcal{H}$ containing every ε-heavy-hitter of x and containing only $(\varepsilon/3)$-heavy-hitters. The recovery succeeds with probability at least $1 - 1/\text{poly}(d)$.*

Proof. Run the ExpanderSketch of Larsen et al. (2019) in parallel with an independent ensemble of CountSketches for point queries and norm estimation. With probability at least $1 - 1/\text{poly}(d)$, the ExpanderSketch recovery algorithm returns a set $\mathcal{H}'$ of size $O(1/\varepsilon^2)$ that contains every ε-heavy hitter of x. Conditioned on this event, the CountSketch ensemble gives, simultaneously for all $i \in \mathcal{H}'$, estimates $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \leq (\varepsilon/10) \|x\|_2$, and also gives an estimate Z satisfying $Z = (1 \pm 0.1) \|x\|_2$, again with failure probability at most $1/\text{poly}(d)$. We output

$$\mathcal{H} = \{i \in \mathcal{H}' : |\widetilde{x}_i| \geq (\varepsilon/2)Z\}.$$

By the point-query and norm-estimation guarantees, every ε-heavy hitter is retained, while any coordinate that is not an $(\varepsilon/3)$-heavy hitter is discarded, after adjusting the constants in the CountSketch accuracy. A union bound over the two sketching components gives the claimed success probability. □

The remaining issue is that the heavy coordinates of Sx are buckets, not coordinates of the original vector x. We handle this in two steps. First, we show that every heavy coordinate of x is mapped by the CountSketch to a heavy bucket of Sx with high probability. Second, we choose the CountSketch hash functions so that the relation between coordinates of x and buckets of Sx can be inverted efficiently. Since CountSketch only requires limited independence, we may use linear hash functions over $\mathbb{F}_2$. A single hash function leaves too many preimages for each heavy bucket, so we repeat the construction a constant number of times and look for coordinates that appear as candidates in many repetitions. With linear hashes, finding these common preimages reduces to solving small linear systems, which gives fast decoding while preserving the sketch size and linear application time.

Theorem 3.7. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists an $(\varepsilon, 1/d, \ell_2)$-heavy-hitter sketch with sketching dimension $O_c(\varepsilon^{-2} \log d)$ that can be applied to a vector $v \in \mathbb{R}^d$ in $O_c(d)$ time. Moreover, the sketch can be decoded in time $O_c(\text{poly}(\log d/\varepsilon))$.*

The space bound $O(\varepsilon^{-2} \log d)$ matches the optimal heavy-hitters sketch size achieved by ExpanderSketch (Larsen et al., 2019). As above, as c approaches 0, the sketch size approaches $O(d)$, which harms the advantage of applying the sketch. Therefore, although this constraint might seem restrictive, it covers a wide range of practical scenarios where ε is not vanishingly small.

For $p > 2$, we get the following result.

Corollary 3.8. *For $p > 2$ and $\varepsilon \geq d^{-1/p+c}$, where $c > 0$ is a constant, there exists a (ε, ℓ_p)-heavy-hitter sketch with sketching dimension $O(\frac{1}{\varepsilon^2} d^{1-2/p} \log d)$ that can be applied to $x \in \mathbb{R}^d$ in $O(d)$ time and such that recovery succeeds with at least $1 - \frac{1}{d}$ probability. Moreover, this sketch recovers all coordinates to additive error $\varepsilon \|x\|_p$.*

Proof. For any vector $x \in \mathbb{R}^d$, $\|x\|_p \geq d^{\frac{1}{p}-\frac{1}{2}} \|x\|_2$ holds. This implies that an (ε, ℓ_p)-heavy-hitter is an $(\varepsilon d^{\frac{1}{p}-\frac{1}{2}}, \ell_2)$-heavy-hitter. The claim follows from the ℓ_2 heavy-hitter of Theorem 3.7. □

We note that this bound has the same $d^{1-2/p}$ dimensional dependence as the optimal scale for F_p moment estimation (Indyk & Woodruff, 2005).

ℓ_2 sampling. The standard exponential rescaling trick reduces ℓ_2 sampling to finding ℓ_2 heavy hitters of a randomly rescaled vector (Jowhari et al., 2011). Thus, by plugging in the fast ℓ_2 heavy-hitters sketch described above, we obtain the same sampling primitive with linear application time.

Lemma 3.9. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists a linear sketch for approximate ℓ_2 sampling with sketching dimension $O_c\left(\varepsilon^{-2} \operatorname{polylog} d\right)$. The sketch can be applied to a nonzero vector $x \in \mathbb{R}^d$ in $O_c(d)$ time. From the sketch, the decoder outputs an approximate ℓ_2 sample with constant probability. Moreover, when the decoder outputs an index i with $x_i \neq 0$, it also outputs an estimate $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \leq \varepsilon |x_i|$ in $O(\operatorname{polylog}(d, \frac{1}{\varepsilon}))$ time.*

Tensor Sketching. We show that by applying similar ideas from the heavy hitters problem explained above, but combining with a tensor sketch, there exists a sketch that can be applied to rank-one tensors in linear time $O(dq)$, rather than in the ambient tensor dimension d^q.

Theorem 3.10. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists an ℓ_2-heavy-hitters sketch for rank-one q-mode tensors with sketching dimension $O_c(q\varepsilon^{-2} \log d)$ and failure probability at most $\frac{1}{d}$. The sketch can be applied to $x = x_1 \otimes \cdots \otimes x_q$ in $O_c(dq)$ time, and the heavy hitters can be decoded in $O_c(\varepsilon^{-O(q)} \operatorname{polylog} d)$ time. Additionally, the same sketch supports point queries: given a multi-index $i = (i_1, \ldots, i_q) \in [d]^q$, the decoder outputs an estimate $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \leq \varepsilon \|x\|_2$.*

Linear regression and low-rank approximation. Our algorithms for linear regression and low-rank approximation follow the same CountSketch-then-fast-JL template as our ℓ_2 estimation sketch. For regression, a CountSketch gives a subspace embedding in input-sparsity time but with too many rows; composing it with a fast JL transform reduces the dimension while keeping the total application time linear when A is sufficiently overdetermined. We repeat this construction a constant number of times and use the ℓ_2 estimation sketch from above to estimate the residual errors of the candidate solutions and select the best one.

Theorem 3.11. *For $\varepsilon \geq 1/d$, where $c > 0$ is a constant, there exists a $(\varepsilon, \frac{1}{d})$-regression sketch for least squares regression with sketching dimension $O(\frac{d}{\varepsilon} \log d)$, that can be applied to a matrix $A \in \mathbb{R}^{n \times d}$ in $O(\operatorname{nnz}(A) + \frac{1}{\varepsilon} d^{3+c})$ time. In particular, if $\operatorname{nnz}(A) \geq \frac{1}{\varepsilon} d^{3+c}$, the application time is $O(\operatorname{nnz}(A))$.*

Oblivious sketches for least-squares regression have been studied since Sarlos (2006), and Clarkson & Woodruff (2017) gave input-sparsity time embeddings for regression and low-rank approximation. Our result keeps the standard residual-error sketching dimension, while showing that high success probability can be obtained with truly linear application time in the highly overdetermined regime.

We note that this approach can also handle the case $\varepsilon < 1/d$ though it gives a more complicated guarantee. But if $\varepsilon < 1/d$, the sketch size is already $\Omega(d^2 \log d)$: one could instead run Gaussian elimination on the augmented matrix $[A \mid b]$ and communicate the resulting top $d \times d$ part directly, which means that sketching no longer gives the advantage.

This formulation is also useful in distributed regression. If server r holds a local contribution (A_r, b_r) to the global instance $(A, b) = (\sum_r A_r, \sum_r b_r)$, then each server can apply the same linear sketch locally and send (SA_r, Sb_r) to the coordinator.

For low-rank approximation, the same composition gives the projection-cost-preserving sketch needed for the standard low-rank recovery guarantee (Musco & Musco, 2020). Repeating the construction produces a constant number of candidate projections, and we use the Frobenius-norm analogue of our ℓ_2 estimation sketch to estimate their errors and select the best candidate.

Theorem 3.12. *Fix any constant $c > 0$. There exists a sketch that, given AS for $A \in \mathbb{R}^{d \times n}$, recovers a rank-$k$ orthogonal projection Π satisfying $\|\Pi A - A\|_F^2 \leq (1 + \varepsilon) \|A_k - A\|_F^2$ with probability at least $1 - \frac{1}{d}$, where A_k is the optimal rank-k approximation to A. The sketching dimension is*

$O(\frac{kd}{\varepsilon^2}\log d)$, *and the sketch can be applied in time*

$$O_c\left(\operatorname{nnz}(A) + \frac{k^2}{\varepsilon^2} d^{1+c} \log \frac{dk}{\varepsilon}\right).$$

In particular, if $\operatorname{nnz}(A) = \widetilde{\Omega}(\frac{k^2}{\varepsilon^2} d^{1+c})$, *the application time is* $O_c(\operatorname{nnz}(A))$.

We note that the truly linear-time sketching for the two linear algebraic problems above only applies for certain parameter regimes – when A is highly overdetermined for regression, and when A is very wide for low-rank approximation. This is not too restrictive, however. For example, when sketching regression problems, one is typically interested in the highly overdetermined case.

4 Experiments

We give proof-of-concept synthetic experiments for Lemma 3.3 by implementing its JL stage with a subsampled randomized Hadamard transform (SRHT). We call the resulting CountSketch–SRHT method Ours and test whether it combines faster application than direct SRHT with fewer final rows than CountSketch at similar failure probabilities.

Setup. We start with $x \in \mathbb{R}^D$, $D = 10^7$: roughly 25% of its entries are zero, and the rest have random signs and heavy-tailed Pareto magnitudes (exponent 2.3). Let b be the second-largest background magnitude. We replace 100 other entries with 90 signed values of magnitude b and 10 of magnitude $5b$, then normalize. Thus two random background entries match or exceed the smaller planted entries, while the squared values of the ten large planted entries sum to 60.2% of $\|x\|_2^2$.

We fix $\varepsilon = 0.05$ and count a failure when the relative norm error exceeds ε. Each setting uses 10,000 sketch seeds. Bars show 95% confidence intervals. A point has m total output rows. Ours first hashes into $r = 65{,}536$ rows, then applies SRHT down to m rows. Three-copy CountSketch assigns $m/3$ rows to each copy and takes the median of their squared-norm estimates. Labels in the time panel give m.

We reapply the first 200 seeded sketches serially for median application time, excluding vector construction and SRHT row sampling. Direct SRHT prunes unselected Hadamard outputs in $O(D \log m)$ time, compared with $O(D \log D)$ for a full transform. A seeded hash chooses buckets and signs.

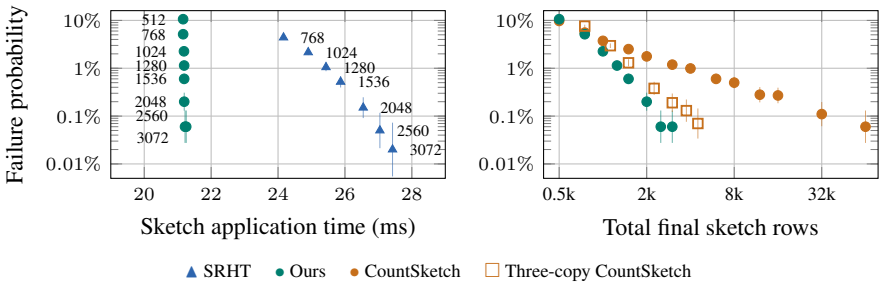

Figure 1: Failure probability at $\varepsilon = 0.05$ versus application time (left) and final sketch rows (right).

Results and scope. At fixed final sketch size, Ours has similar observed failure probability but faster application than direct SRHT. At $m = 1{,}536$, the two give 0.60% versus 0.52% and 21.2 versus 25.9 ms. At $m = 2{,}048$, they give 0.20% versus 0.15% and 21.2 versus 26.5 ms. Thus Ours is 18% and 20% faster at these sizes. The implemented costs are $O(D + r \log r)$ for Ours and $O(D \log m)$ for pruned SRHT. The time gap could widen as required m grows.

On the tested grid, at observed failure targets of 0.6% and 0.2%, Ours uses 1,536 and 2,048 rows, versus 6,144 and 32,768 for single-copy CountSketch. The row saving grows from fourfold to sixteenfold as the target tightens. Three-copy CountSketch improves on single-copy but still uses 2,304 and 3,072 rows. The gap may widen further if larger inputs demand still lower failure probabilities and $r \ll D$ remains feasible. Together, these measured gains place Ours in a practically useful middle ground.

AI Use Statement Generative AI tools were not used to develop the theoretical results or conduct the core scientific reasoning of this work. They were used only as auxiliary tools for writing and editing, literature search, and verification of exposition, calculations, and implementation details. All technical claims, proofs, experiments, and final content were reviewed and validated by the authors.

REFERENCES

Thomas D Ahle, Michael Kapralov, Jakob BT Knudsen, Rasmus Pagh, Ameya Velingker, David P Woodruff, and Amir Zandieh. Oblivious sketching of high-degree polynomial kernels. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 141–160. SIAM, 2020.

Josh Alman and Huacheng Yu. Faster update time for turnstile streaming algorithms. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1803–1813. SIAM, 2020.

Jean Bourgain, Sjoerd Dirksen, and Jelani Nelson. Toward a unified theory of sparse dimensionality reduction in euclidean space. In *Proceedings of the forty-seventh annual ACM symposium on Theory of Computing*, pp. 499–508, 2015.

Karl Bringmann and Tobias Friedrich. Exact and efficient generation of geometric random variates and random graphs. In *Automata, Languages, and Programming - 40th International Colloquium, ICALP, Proceedings*, pp. 267–278, 2013.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *International Colloquium on Automata, Languages, and Programming*, pp. 693–703. Springer, 2002.

Chi-Ning Chou, Zhixian Lei, and Preetum Nakkiran. Tracking the ℓ_2 norm with constant update time. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2019.

Kenneth L Clarkson and David P Woodruff. Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6):1–45, 2017.

Michael B Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 163–172, 2015.

Don Coppersmith. Rapid multiplication of rectangular matrices. *SIAM Journal on Computing*, 11(3): 467–471, 1982.

Graham Cormode, Shanmugavelayutham Muthukrishnan, and Ke Yi. Algorithms for distributed functional monitoring. *ACM Transactions on Algorithms (TALG)*, 7(2):1–20, 2011.

Hossein Esfandiari, Praneeth Kacham, Vahab Mirrokni, David P Woodruff, and Peilin Zhong. Optimal communication for classic functions in the coordinator model and beyond. *arXiv preprint arXiv:2403.20307*, 2024.

Casper Benjamin Freksen. An introduction to johnson-lindenstrauss transforms. *arXiv preprint arXiv:2103.00564*, 2021.

Casper Benjamin Freksen and Kasper Green Larsen. On using toeplitz and circulant matrices for johnson–lindenstrauss transforms. *Algorithmica*, 82(2):338–354, 2020.

Sumit Ganguly. Polynomial estimators for high frequency moments. *arXiv preprint arXiv:1104.4552*, 2011.

Zengfeng Huang, Ke Yi, and Qin Zhang. Randomized algorithms for tracking distributed count, frequencies, and ranks. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGAI symposium on Principles of Database Systems*, pp. 295–306, 2012.

Zengfeng Huang, Xuemin Lin, Wenjie Zhang, and Ying Zhang. Communication-efficient distributed covariance sketch, with application to distributed pca. *The Journal of Machine Learning Research*, 22(1):3643–3680, 2021.

Piotr Indyk and David Woodruff. Optimal approximations of the frequency moments of data streams. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 202–208, 2005.

Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Ion Stoica, Raman Arora, et al. Communication-efficient distributed sgd with sketching. *Advances in Neural Information Processing Systems*, 32, 2019.

Thathachar S Jayram and David P Woodruff. Optimal bounds for johnson-lindenstrauss transforms and streaming problems with subconstant error. *ACM Transactions on Algorithms (TALG)*, 9(3): 1–17, 2013.

William B Johnson, Joram Lindenstrauss, et al. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26(189-206):1, 1984.

Hossein Jowhari, Mert Sağlam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In *Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 49–58, 2011.

Praneeth Kacham, Rasmus Pagh, Mikkel Thorup, and David P Woodruff. Pseudorandom hashing for space-bounded computation with applications in streaming. *arXiv preprint arXiv:2304.06853*, 2023.

Thomas Kailath and Ali H Sayed. *Fast reliable algorithms for matrices with structure*. SIAM, 1999.

Daniel M Kane, Jelani Nelson, Ely Porat, and David P Woodruff. Fast moment estimation in data streams in optimal space. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pp. 745–754, 2011.

Ravi Kannan, Santosh Vempala, and David Woodruff. Principal component analysis and higher correlations for distributed data. In *Conference on Learning Theory*, pp. 1040–1057. PMLR, 2014.

Kasper Green Larsen, Jelani Nelson, and Huy L Nguyên. Time lower bounds for nonadaptive turnstile streaming algorithms. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 803–812, 2015.

Kasper Green Larsen, Jelani Nelson, Huy L Nguyen, and Mikkel Thorup. Heavy hitters via cluster-preserving clustering. *Communications of the ACM*, 62(8):95–100, 2019.

Arvind Mahankali and David Woodruff. Linear and kernel classification in the streaming model: Improved bounds for heavy hitters. *Advances in Neural Information Processing Systems*, 34: 14407–14420, 2021.

Jayadev Misra and David Gries. Finding repeated elements. *Science of computer programming*, 2(2): 143–152, 1982.

Cameron Musco and Christopher Musco. Projection-cost-preserving sketches: Proof strategies and constructions. *arXiv preprint arXiv:2004.08434*, 2020.

Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. Fetchsgd: Communication-efficient federated learning with sketching. In *International Conference on Machine Learning*, pp. 8253–8265. PMLR, 2020.

Tim Roughgarden et al. Communication complexity (for algorithm designers). *Foundations and Trends® in Theoretical Computer Science*, 11(3–4):217–404, 2016.

Tamas Sarlos. Improved approximation algorithms for large matrices via random projections. In *2006 47th annual IEEE symposium on foundations of computer science (FOCS'06)*, pp. 143–152. IEEE, 2006.

Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. Powersgd: Practical low-rank gradient compression for distributed optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

David Woodruff and Qin Zhang. Subspace embeddings and\ell_p-regression using exponential random variables. In *Conference on Learning Theory*, pp. 546–567. PMLR, 2013.

David P Woodruff and Qin Zhang. Tight bounds for distributed functional monitoring. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pp. 941–960, 2012.

David P Woodruff et al. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.

A HEAVY-HITTERS AND APPLICATIONS

A.1 COMBINING WITH EXPANDERSKETCH

We show how to combine the above construction with ExpanderSketch (Larsen et al., 2019) to get fast recovery. Instead of taking each T to be an arbitrary JL sketch as above (Section 3), we can let each T be an ExpanderSketch. As before, we form the sketches TSx. We will then apply the recovery algorithm for the T's to obtain the heavy coordinates of (Sx). Fortunately, since S is a CountSketch, we can use the well-known observation that CountSketch preserves heavy coordinates. We formulate this as follows:

Lemma A.1. *For any $\varepsilon, \delta \in (0,1)$, let S be a CountSketch matrix as in Definition 3.1 with sketching dimension $m = \Omega(\varepsilon^{-2}\delta^{-1})$. Let $x \in \mathbb{R}^d$ and suppose that $|x_k| \geq \varepsilon \|x\|_2$. Then $\left|(Sx)_{h(k)}\right| \geq (\varepsilon/4)\|Sx\|_2$ holds with probability at least $1-\delta$.*

Proof. Assume without loss of generality that $\|x\|_2 = 1$. First, by the standard CountSketch norm-preservation guarantee, we have $\|Sx\|_2 \leq 1+\varepsilon$ with probability at least $1-\delta$ after adjusting constants in the number of buckets.

Let $Z = \sum_{i \neq k : h(i)=h(k)} \sigma_i x_i$ be the collision noise in this bucket. Since the hash family is pairwise independent and the signs satisfy $\mathrm{E}[\sigma_i \sigma_j] = 0$ for $i \neq j$, we have

$$\mathrm{E}[Z^2] = \sum_{i \neq k} \Pr[h(i) = h(k)] x_i^2 = \frac{1}{m}\|x_{-k}\|_2^2 \leq \frac{1}{m} = O(\varepsilon^2\delta).$$

Thus by Markov's inequality, we get $\Pr[|Z| \geq \varepsilon/2] \leq \frac{4}{\varepsilon^2}\mathrm{E}[Z^2] \leq \delta$. Except with probability at most δ, $|(Sx)_{h(k)}| \geq |x_k| - |Z| \geq \varepsilon/2 \geq \frac{\varepsilon}{4}\|Sx\|_2$ for $\varepsilon < 1$. Adjusting constants in the sketching dimension gives total failure probability at most δ. □

Thus, by finding the heavy hitters of Sx, we can in principle recover the heavy hitters of the original vector x. The remaining issue is that a heavy coordinate of x is represented in Sx only through the bucket to which it is mapped by the hash function h. Therefore, to turn heavy buckets of Sx back into heavy coordinates of x, we would like to construct our CountSketch distribution S using hash functions that are easy to invert.

Suppose now that ExpanderSketch decodes the heavy buckets of Sx and outputs a set $\mathcal{H}$. Then the corresponding candidate set for the heavy hitters of x is $\mathcal{C} = h^{-1}(\mathcal{H})$. This set may also contain light coordinates of x, so we repeat the procedure several times. Namely, for a parameter C, we take C independent repetitions. By Lemma A.1, with high probability every heavy coordinate of x is included in the candidate set in every repetition, while a light coordinate may appear by chance in a few repetitions but will not persist across many of them. We therefore keep only those coordinates that appear as heavy-hitter candidates in at least half of the repetitions.

Lemma A.2. *Let $S_1, \ldots, S_C$ be independent CountSketches, and let $T_1, \ldots, T_C$ be the corresponding ExpanderSketches. For each repetition r, let $\mathcal{H}_r$ be the set of heavy buckets recovered from $T_r S_r x$, and let $\mathcal{C}_r = h_r^{-1}(\mathcal{H}_r)$. Define the final candidate set $\mathcal{C} = \{k \in [d] : k \in \mathcal{C}_r \text{ for at least } C/2 \text{ values of } r\}$. Then for any $\delta \in (0,1)$ used as the CountSketch error parameter, there is a parameter $\eta = O(\delta + 1/\mathrm{poly}(d))$ such that, for every fixed coordinate k:*

1. *If $|x_k| \geq \varepsilon\|x\|_2$, then $\Pr[k \notin \mathcal{C}] \leq (4\eta)^{C/2}$.*
2. *If $|x_k| \leq (\varepsilon/10)\|x\|_2$, then $\Pr[k \in \mathcal{C}] \leq (4\eta)^{C/2}$.*

In particular, for C a sufficiently large constant and $\eta = 1/\mathrm{poly}(d)$, both failure probabilities are $1/\mathrm{poly}(d)$.

Proof. Consider one repetition and write $\mathcal{C}_r = h_r^{-1}(\mathcal{H}_r)$. If $|x_k| \geq \varepsilon\|x\|_2$, then Lemma A.1 implies that the bucket $h_r(k)$ is $\varepsilon/4$-heavy in $S_r x$ with probability at least $1-\delta$. Conditioned on this event, by Proposition 3.6 $h_r(k) \in \mathcal{H}_r$ with probability at least $1 - 1/\mathrm{poly}(d)$. Hence $k \in \mathcal{C}_r$ with probability at least $1-\eta$.

On the other hand, if $|x_k| \leq (\varepsilon/10)\|x\|_2$, then by the same second-moment/Markov bound used in Lemma A.1, the bucket $h_r(k)$ is not $(\varepsilon/3)$-heavy in $S_r x$ except with probability at most $O(\delta)$. Conditioned on this, by Proposition 3.6 which outputs only $(\varepsilon/3)$-heavy-hitters, implies that $h_r(k) \notin \mathcal{H}_r$ except with probability $1/\text{poly}(d)$. Thus $k \in \mathcal{C}_r$ with probability at most η.

The repetitions are independent. Therefore, in the first case k fails to appear in at least half of the $\mathcal{C}_r$'s with probability at most

$$\binom{C}{C/2}\eta^{C/2} \leq (4\eta)^{C/2},$$

and in the second case k appears in at least half of the $\mathcal{C}_r$'s with the same bound. This proves the lemma. □

The main remaining task is to choose the hash functions so that this entire filtering procedure can be carried out efficiently. For this, assume without loss of generality that d and n are powers of two. We define the hash function family $h_{A,b} : [d] \to [m]$ where $b \in \mathbb{F}_2^{\log m}$ and $A \in \mathbb{F}_2^{\log m \times \log d}$ and A and b are chosen uniformly from the respective spaces. We define

$$h = h_{A,b}(N) = i_m(A\vec{N} + b)$$

where $\vec{N}$ is the $\mathbb{F}_2$ vector corresponding to the binary expansion of N, and where $i_m(y)$ gives the number in $[m]$ whose binary expansion is given by y. For the CountSketch distribution take S such that $Se_k = \sigma(k)e_{h(k)}$ where $\sigma : [d] \to \{-1, 1\}$ is drawn from any 4-wise independent hash function family. For completeness, we sketch an argument that this hash function family is indeed pairwise independent.

Lemma A.3. *Let $h'_{A,b}(N) = A\vec{N} + b \in \mathbb{F}_2^M$ as above, where A and b have i.i.d entries over $\mathbb{F}_2$. Then $h'_{A,b}(x)$ is uniform over $\mathbb{F}_2^M$, and $h'_{A,b}(v_1), h'_{A,b}(v_2)$ are uniform and independent for $v_1 \neq v_2$.*

Proof. For any fixed v, the value $h'_{A,b}(v) = Av + b$ is uniform over $\mathbb{F}_2^M$, since b is uniform. Now fix $v_1 \neq v_2$. It suffices to show that $h'_{A,b}(v_1) - h'_{A,b}(v_2) = A(v_1 - v_2)$ is uniform over $\mathbb{F}_2^M$, and is independent of $h'_{A,b}(v_2)$. Since $v_1 - v_2 \neq 0$ and the entries of A are i.i.d. over $\mathbb{F}_2$, each coordinate of $A(v_1 - v_2)$ is an unbiased bit, independently across rows. Thus $A(v_1 - v_2)$ is uniform over $\mathbb{F}_2^M$. Moreover, this difference depends only on A, while $h'_{A,b}(v_2) = Av_2 + b$ is uniform over $\mathbb{F}_2^M$ even conditioned on A, because b is uniform and independent. Hence $h'_{A,b}(v_1)$ and $h'_{A,b}(v_2)$ are uniform and independent. □

Now we can prove the main theorem.

Theorem 3.7. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists an $(\varepsilon, 1/d, \ell_2)$-heavy-hitter sketch with sketching dimension $O_c(\varepsilon^{-2}\log d)$ that can be applied to a vector $v \in \mathbb{R}^d$ in $O_c(d)$ time. Moreover, the sketch can be decoded in time $O_c(\text{poly}(\log d/\varepsilon))$.*

Proof. Take C CountSketches $S_1, \ldots, S_C$ in parallel. In each repetition r, we first apply such a CountSketch S_r and then apply the modified ExpanderSketch of Proposition 3.6 to $S_r x$. We set the failure parameter of each CountSketch to be $\delta_0 = d^{-c}$. Then each CountSketch has sketching dimension $m = O(\varepsilon^{-2}\delta_0^{-1}) \leq O(d^{1-c})$, since $\varepsilon^{-2} \leq d^{1-2c}$. Hence the CountSketch reduces the dimension to $d^{1-\Omega(c)}$. The CountSketch takes $O(d)$ time to apply, while the ExpanderSketch is applied only to a vector of dimension $m \leq O(d^{1-c})$. Thus one repetition can be applied in $O_c(d)$ time.

Let $\mathcal{H}_r$ be the set of heavy buckets recovered in the r-th repetition, and let $\mathcal{C}_r = h_r^{-1}(\mathcal{H}_r)$. By Lemma A.2, there is a parameter $\eta = O(\delta_0 + 1/\text{poly}(d)) = O(d^{-c})$ such that every ε-heavy coordinate is included from the majority candidate set with probability at least $1 - (4\eta)^{C/2}$, and every coordinate of magnitude at most $(\varepsilon/10)\|x\|_2$ is excluded in the majority candidate set with probability at least $1 - (4\eta)^{C/2}$.

Here, we choose $C = \Theta(1/c)$ large enough so that $(4\eta)^{C/2} \leq 1/d^3$. Taking a union bound over all coordinates, with probability at least $1 - O(\frac{1}{d})$, all ε-heavy-hitters are retained and no coordinate with magnitude at most $(\varepsilon/10)\|x\|_2$ is retained. Therefore the final candidate set has size $O(1/\varepsilon^2)$.

Thus it suffices to find all coordinates of x that occur in at least half of the $\mathcal{C}_i$'s. To do this, we simply take a brute-force approach. There are $\binom{C}{C/2}$ ways to choose a subset $L \subseteq [C]$ indexing half of the $\mathcal{C}_i$'s. For each $i \in L$ there are then $O((1/\varepsilon^2)^{C/2}) = \text{poly}(1/\varepsilon)$ ways to choose elements $k_1, \ldots, k_{C/2} \in [d]$ from each set in $\{\mathcal{H}_i : i \in L\}$, since each set $\mathcal{H}_i$ has $O(1/\varepsilon^2)$ elements. We would like to check if these elements have a common preimage under their respective hash functions. By the construction of our hash functions, this reduces to solving a linear system over $\mathbb{F}_2$. Since the CountSketch dimension is $m \le d^{1-c}$, each preimage constraint contributes only $\log m = O(\log d)$ linear equations over $\mathbb{F}_2$, and the unknown index has only $\log d$ bits; hence each common-preimage check is a linear system of size $O(C \log d) \times \log d$, which can be solved in $\text{poly}(\log d)$ time for constant C. So the total decoding time is

$$O\left(\binom{C}{C/2}(1/\varepsilon)^{O(C)}\text{poly}(\log d)\right) = O\left((1/\varepsilon)^{O(1/c)}\text{poly}(\log d)\right).$$

Finally, the total space is $C \cdot O(\varepsilon^{-2} \log m) = O_c(\varepsilon^{-2} \log d)$. This proves the theorem. □

A.2 Extension to ℓ_2 Sampling

In this section, we show that the heavy-hitters sketches developed above also give fast linear sketches for ℓ_2 sampling. An ℓ_2 sampler for a nonzero vector $x \in \mathbb{R}^d$ outputs an index $i \in [d]$ with probability $x_i^2 / \|x\|_2^2$. We use the standard reduction from ℓ_2 sampling to heavy hitters based on exponential rescaling, and then instantiate the required heavy-hitters primitive using Theorem 3.7.

Lemma A.4 (Sampling via exponential rescaling)**.** *Let $x \in \mathbb{R}^d$ be nonzero. Let $E_1, \ldots, E_d$ be independent* $\text{Exp}(1)$ *random variables and define $z_i = x_i/\sqrt{E_i}$. Then the index of the maximum magnitude entry of z is an exact ℓ_2 sample:*

$$\Pr\left[\arg\max_{j \in [d]} |z_j| = i\right] = \frac{x_i^2}{\|x\|_2^2}.$$

Moreover, the analysis of Jowhari et al. (2011) shows that finding the heavy hitters of z, for appropriate parameters, gives an approximate ℓ_2 sample of x. In particular, the maximizer is, with good probability, an $(\Omega(1/\log d), \ell_2)$-heavy hitter of z.

Thus the sampling problem is reduced to a single heavy-hitters problem on the rescaled vector. The fast-decoding ℓ_2 heavy-hitters sketch of Theorem 3.7, together with the point-query estimates included in Proposition 3.6, is exactly the primitive we need.

Lemma 3.9. *Fix any constant $c > 0$. For $\varepsilon \ge d^{-1/2+c}$, there exists a linear sketch for approximate ℓ_2 sampling with sketching dimension $O_c\left(\varepsilon^{-2} \operatorname{polylog} d\right)$. The sketch can be applied to a nonzero vector $x \in \mathbb{R}^d$ in $O_c(d)$ time. From the sketch, the decoder outputs an approximate ℓ_2 sample with constant probability. Moreover, when the decoder outputs an index i with $x_i \neq 0$, it also outputs an estimate $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \le \varepsilon |x_i|$ in $O(\text{polylog}(d, \frac{1}{\varepsilon}))$ time.*

Proof. Let $D = \text{diag}(E_1^{-1/2}, \cdots, E_d^{-1/2})$ be a diagonal matrix, where $\{E_i\}$ are i.i.d. draws from $\text{Exp}(1)$. Let $z = Dx$. For fixed randomness, D is a diagonal linear map, and therefore composing it with a heavy-hitters sketch remains a linear sketch of x.

Set the internal heavy-hitters accuracy to $\rho = \Theta(\varepsilon/\operatorname{polylog} d)$, with the hidden polylogarithmic factor chosen large enough for the reduction in Lemma A.4. By Theorem 3.7 and Proposition 3.6, there is a sketch which recovers the required ℓ_2 heavy hitters of z and estimates their values to additive error at most $\rho \|z\|_2$. Since $\varepsilon \ge d^{-1/2+c}$, we still have $\rho \ge d^{-1/2+c'}$ for some constant $c' > 0$. Thus the sketching dimension is $O_c(\rho^{-2} \log d) = O_c(\varepsilon^{-2} \operatorname{polylog} d)$, and it can be applied in $O_c(d)$ time.

We run this sketch on z and pass the recovered heavy coordinates to the reduction of Jowhari et al. (2011). By Lemma A.4, the maximizer of the rescaled vector is a heavy hitter with good probability, and recovering such heavy hitters is enough to obtain an approximate ℓ_2 sample with constant probability.

It remains to justify the value estimate. If the decoder returns an estimate $\widetilde{z}_i$ for a recovered coordinate satisfying $|\widetilde{z}_i - z_i| \le \rho \|z\|_2$, then the sampler only uses it when i is heavy at the threshold guaranteed by Lemma A.4. Our choice of ρ therefore gives $|\widetilde{z}_i - z_i| \le \varepsilon |z_i|$. Since the randomness is shared, the decoder knows the diagonal entry D_{ii} and outputs $\widetilde{x}_i = \widetilde{z}_i / D_{ii}$. Thus $|\widetilde{x}_i - x_i| \le \varepsilon |x_i|$ holds. □

A.3 Extension to Tensor Sketching

In this section, we show how one can extend the same argument as above to get a JL sketch for tensor sketching. We first record the tensor analogue of the CountSketch norm-preservation fact.

Lemma A.5. *Let $S = S_1 \otimes \cdots \otimes S_q$, where each S_j is an independent CountSketch with $O(q/(\varepsilon^2\delta))$ rows. Then for every fixed q-mode tensor $v \in (\mathbb{R}^d)^{\otimes q}$,*

$$\|Sv\|_2^2 = (1 \pm \varepsilon)\,\|v\|_2^2$$

with probability at least $1 - \delta$.

Proof. Each CountSketch satisfies the $(\varepsilon/\sqrt{q}, \delta, 2)$-JL moment property for the stated number of rows. By the tensor-product JL moment bound of Ahle et al. (2020), the tensor product $S_1 \otimes \cdots \otimes S_q$ satisfies the $(\varepsilon, \delta, 2)$-JL moment property, which gives the norm guarantee for a fixed tensor. □

Now composing the tensor CountSketch reduction with the same final norm-estimation step gives the following tensor sketching result.

Lemma A.6. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists an ε-distortion tensor sketch with sketching dimension $O_c(\varepsilon^{-2}\log d)$ that can be applied to a rank-one tensor $x = x_1 \otimes \cdots \otimes x_q$ in $O_c(dq)$ time and succeeds with probability at least $1 - 1/\mathrm{poly}(d)$ for an arbitrary fixed polynomial. The same sketch also supports point queries: given a multi-index $i = (i_1, \ldots, i_q) \in [d]^q$, the decoder outputs an estimate $\widetilde{x}_i$ with*

$$|\widetilde{x}_i - x_i| \leq \varepsilon\,\|x\|_2\,.$$

Proof. Sample $C = O_c(1)$ independent tensor products of CountSketch $Q_1, \ldots, Q_C$, with the CountSketch failure parameter set to $\delta_0 = d^{-c}$. As in Lemma 3.3, we first apply a tensor product of CountSketches to reduce the dimension. By Lemma A.5, each mode is reduced to $m = O(q\varepsilon^{-2}\delta_0^{-1}) \leq O(qd^{1-c})$ rows, so this first stage can be applied to the factor vectors in $O(dq)$ time. Once the dimension has been reduced, we compose with a dense tensor sketch of Ahle et al. (2020); its application time is lower order in the stated parameter regime, with constants depending on c. Repeating a constant number of times and taking the median boosts the success probability to $1 - 1/\mathrm{poly}(d)$. The point-query guarantee follows from applying the same norm-estimation identity used for vector point queries to the tensor basis vector $e_{i_1} \otimes \cdots \otimes e_{i_q}$. □

We can also combine the tensor CountSketch reduction with a fast tensor heavy-hitters sketch, in the same way that Section 4.2 combines CountSketch with ExpanderSketch. Before introducing the main theorem, we first prove the following lemma, which is a tensor version of the heavy-coordinate-to-heavy-bucket statement used in Section A.1.

Lemma A.7. *Let $S = S_1 \otimes \cdots \otimes S_q$, where each S_j is a CountSketch with hash function $h_j : [d] \to [m]$ and $m = O(q/(\varepsilon^2\delta))$. Let x be a q-mode tensor indexed by $[d]^q$, and let Sx be indexed by $[m]^q$.*

1. *If $i = (i_1, \ldots, i_q)$ is an ε-heavy index of x, then $(h_1(i_1), \ldots, h_q(i_q))$ is an $\varepsilon/2$-heavy index of Sx with probability at least $1 - \delta$.*
2. *If a coordinate value $a \in [d]$ is not the rth coordinate of any ε-heavy index of x, then $h_r(a)$ is not the rth coordinate of any $\varepsilon/10$-heavy index of Sx except with probability $O(1/(m\varepsilon^2))$.*

Proof. For the first part, the tensor CountSketch has the JL moment property by Lemma A.5. Thus the point-query estimate $\langle Se_i, Sx\rangle$ approximates x_i to additive error $O(\varepsilon\,\|x\|_2)$, and $\|Sx\|_2 = (1 \pm O(\varepsilon))\,\|x\|_2$, so heavy entries remain heavy after adjusting constants. For the second part, fix a coordinate position, say the first one. The probability that $h_1(a)$ is hit by the first coordinate of an $\varepsilon/5$-heavy tensor entry is at most $O(1/(m\varepsilon^2))$. Conditioned on no such collision, zero out all $\varepsilon/10$-heavy entries of x; the same point-query bound shows that the sketch has no heavy entry with first coordinate $h_1(a)$. □

Now we can prove the main theorem. At a high level, this is the same filtering idea as in the vector case, but applied separately in each tensor mode. CountSketch reduces each mode quickly, the tensor heavy-hitters sketch identifies heavy buckets in the reduced tensor, and the linear hash functions let us

invert the bucket information by solving small linear systems. The only extra cost is the combinatorial search over candidate coordinates across the q modes, which accounts for the $\varepsilon^{-O(q)}$ decoding time.

Theorem 3.10. *Fix any constant $c > 0$. For $\varepsilon \geq d^{-1/2+c}$, there exists an ℓ_2-heavy-hitters sketch for rank-one q-mode tensors with sketching dimension $O_c(q\varepsilon^{-2}\log d)$ and failure probability at most $\frac{1}{d}$. The sketch can be applied to $x = x_1 \otimes \cdots \otimes x_q$ in $O_c(dq)$ time, and the heavy hitters can be decoded in $O_c(\varepsilon^{-O(q)}\operatorname{polylog} d)$ time. Additionally, the same sketch supports point queries: given a multi-index $i = (i_1, \ldots, i_q) \in [d]^q$, the decoder outputs an estimate $\widetilde{x}_i$ satisfying $|\widetilde{x}_i - x_i| \leq \varepsilon \|x\|_2$.*

Proof. We use $C = O_c(1)$ independent repetitions. In each repetition, we apply a tensor product CountSketch $S_1 \otimes \cdots \otimes S_q$, using linear hash functions of the form $A\vec{N} + b$ in each mode, with failure parameter $\delta_0 = d^{-c}$. Since $\varepsilon \geq d^{-1/2+c}$, each mode is reduced to dimension $m = O(q\varepsilon^{-2}\delta_0^{-1}) \leq O(qd^{1-c})$. We then run the tensor heavy-hitters sketch of Mahankali & Woodruff (2021) on the reduced tensor. By Lemma A.7, every heavy tensor entry maps to a heavy entry of the reduced tensor with high probability, while light coordinates do not persist across many repetitions.

It remains to recover the original multi-indices from the heavy indices in the reduced tensors. We do this mode by mode. For the first mode, for example, each recovered heavy index gives a candidate bucket value, and we look for original indices that map to such buckets in at least half of the repetitions by iterating over all $\binom{C}{C/2}$ subsets. Because the hash functions are linear over $\mathbb{F}_2$, checking whether a proposed collection of buckets has a common preimage reduces to solving a linear system of size $\operatorname{poly}(\log d)$. Repeating this for all q modes gives $O(\varepsilon^{-O(q)})$ candidate multi-indices. Finally, we verify these candidates using the point-query guarantee of Lemma A.6. We also include an independent copy of the sketch from Lemma A.6 in parallel, which gives the stated point-query guarantee for any queried multi-index. This adds only $O_c(\varepsilon^{-2}\log d)$ space and $O_c(dq)$ application time, and a union bound over the two components gives the claimed failure probability. □

B General ℓ_p norm estimation

In this section, we analyze F_p moment estimation for $p > 2$ and a related mixed-norm observation based on reshaping. We first give the proof of the fast ℓ_2 estimation algorithm.

Lemma 3.3. *For $d \geq 2$ and $\varepsilon \in (0,1)$ satisfying $\varepsilon \geq d^{-0.5+c}$, where $c > 0$ is a constant, there exists an ε-distortion linear ℓ_2 estimation sketch with sketching dimension $O(\varepsilon^{-2}\log d)$ that can be applied to a vector $x \in \mathbb{R}^d$ in $O_c(d)$ time and, for every fixed x, succeeds with probability at least $1 - 1/d$.*

More generally, for any failure probability $\delta \in (0, 1/2]$, distortion parameter $\varepsilon \in (0,1)$, and positive integer C, there exists an ε-distortion linear ℓ_2 estimation sketch with failure probability at most δ and sketching dimension $O(\varepsilon^{-2}\log(1/\delta))$ that can be applied in time

$$O\left(C\,\mathrm{nnz}(x) + \varepsilon^{-2}\delta^{-1/C}\log\frac{1}{\delta}\right).$$

Proof. We first prove the general claim for $1 \leq C \leq \lceil \log(1/\delta) \rceil$. Set

$$\varepsilon_1 = \varepsilon_2 = \frac{\varepsilon}{4}, \qquad \delta_1 = \delta_2 = \frac{\delta^{1/C}}{8}.$$

Take $S_1, \ldots, S_{2C}$ to be independent CountSketches, each with

$$m_1 = \Theta(\varepsilon_1^{-2}\delta_1^{-1})$$

rows, with a sufficiently large constant so that its distortion is at most ε_1 with failure probability at most δ_1.

Let r be the smallest even integer at least $\log_2(1/\delta_2)$, so that

$$r = O\left(1 + \frac{1}{C}\log\frac{1}{\delta}\right).$$

For each $i \in [2C]$, take independent CountSketches $T_{i,1}, \ldots, T_{i,r}$ from $\mathbb{R}^{m_1}$ to $\mathbb{R}^{m_2}$, where $m_2 = \Theta(\varepsilon_2^{-2})$ is chosen so that each has distortion at most ε_2 with failure probability at most $1/16$. All

CountSketches are mutually independent, and within each CountSketch the bucket hash and signs are independent.

The sketch stores all vectors $T_{i,j}S_ix$. Their concatenation is a linear sketch Qx. Its decoder computes

$$\alpha_i = \mathrm{median}_{j\in[r]} \left\|T_{i,j}S_ix\right\|_2, \qquad \alpha = \mathrm{median}_{i\in[2C]}\, \alpha_i.$$

For an even number of values, we take the average of the two middle values. In particular, the norm estimate is this nested median, not the Euclidean norm of the concatenated sketch.

Fix i and condition on S_i. By Fact 3.2 and independence of the $T_{i,j}$, the probability that α_i fails to approximate $\left\|S_ix\right\|_2$ within a $(1 \pm \varepsilon_2)$ factor is at most

$$\binom{r}{r/2}\left(\frac{1}{16}\right)^{r/2} \le 2^{-r} \le \delta_2.$$

Combining this with the guarantee for S_i, we obtain

$$\alpha_i = (1 \pm 2(\varepsilon_1 + \varepsilon_2))\left\|x\right\|_2 = (1 \pm \varepsilon)\left\|x\right\|_2$$

with failure probability at most $\delta_1 + \delta_2$.

The $2C$ outer repetitions are independent. Thus the probability that at least C of their estimates fail is at most

$$\binom{2C}{C}(\delta_1 + \delta_2)^C \le \big(4(\delta_1 + \delta_2)\big)^C = \delta.$$

Consequently, $\alpha = (1 \pm \varepsilon)\left\|x\right\|_2$ with probability at least $1 - \delta$.

The total sketching dimension is

$$2Crm_2 = O\left(\varepsilon^{-2}\left(C + \log\frac{1}{\delta}\right)\right) = O\left(\varepsilon^{-2}\log\frac{1}{\delta}\right),$$

where the last equality uses $C \le \lceil \log(1/\delta) \rceil$ and $\delta \le 1/2$. Only the composed sketches are retained; the intermediate vectors S_ix need not be included in the final sketch.

Including initialization, each S_ix takes $O(\mathrm{nnz}(x) + m_1)$ time to compute, and each $T_{i,j}$ takes $O(m_1 + m_2)$ time to apply to S_ix. Since $m_2 = O(m_1)$, the total application time is

$$\begin{aligned} O(C\,\mathrm{nnz}(x) + Crm_1) &= O\left(C\,\mathrm{nnz}(x) + \varepsilon^{-2}\delta^{-1/C}\left(C + \log\frac{1}{\delta}\right)\right) \\ &= O\left(C\,\mathrm{nnz}(x) + \varepsilon^{-2}\delta^{-1/C}\log\frac{1}{\delta}\right). \end{aligned}$$

For $C > \lceil \log(1/\delta) \rceil$, use the construction with $C' = \lceil \log(1/\delta) \rceil$ instead. Since $C' \le C$ and

$$\delta^{-1/C'} \le e \le e\,\delta^{-1/C},$$

the same dimension and application-time bounds hold for every positive integer C.

Finally, set $\delta = 1/d$ and choose $C = \lceil 1/c \rceil$. Since $\varepsilon^{-2} \le d^{1-2c}$ and $1/C \le c$, the general application-time bound becomes

$$O\Big(C\,\mathrm{nnz}(x) + \varepsilon^{-2}d^{1/C}\log d\Big) = O_c\big(d + d^{1-c}\log d\big) = O_c(d).$$

This proves the first claim. □

B.1 F_p MOMENT ESTIMATION FOR $p > 2$

First, we construct fast sketches for F_p moment estimation when $p > 2$. For convenience, we focus on the F_p moment $\left\|x\right\|_p^p$ rather than the ℓ_p norm $\left\|x\right\|_p$. The typical approach for F_p moment estimation of $x \in \mathbb{R}^d$, introduced by Indyk & Woodruff (2005), is to partition the coordinates of x into approximate level sets, and then to approximate the size of each level set. Estimating the level set sizes can be done by subsampling at geometric rates, and then extracting the ℓ_p-heavy-hitters for

each sample. This gives an F_p estimation sketch that can be applied in $O(d \cdot \text{poly}(\log d/\varepsilon))$ time, with $9/10$ success probability. Our goal is to boost this to $1/\text{poly}(d)$ failure probability, and linear sketching time without increasing the space substantially.

The key intuition is that the highest sampling levels contribute the bulk of the runtime of F_p estimation sketches. These levels are needed to catch the heavy coordinates of x. However, we can identify an approximate heavy set and estimate its coordinates using a single high-probability coordinate-estimation sketch that can be applied in truly linear time. We then sketch independently subsampled copies of the original vector, subtract the estimated sampled heavy contributions during decoding, and rescale the remaining estimates. Using an F_p sketch whose update time is $O(\log d)$, independent of its accuracy parameter, a sampling rate $q = \Theta(\log^{-3} d)$ allows us to repeat the sampled sketches $O(\log d)$ times without exceeding linear application time. The coordinate-estimation sketch is shared across all repetitions.

The following lemma shows how we can deal with the light part of the vector; it shows that the F_p moment of a flat vector is approximately preserved with high probability, when we subsample at a high enough rate.

Lemma B.1. *Let $x \in \mathbb{R}^d$, $\varepsilon, \delta \in (0,1)$, and $q \in (0,1]$. Let $S \subseteq [d]$ be a random subset formed by including each coordinate independently with probability q. Suppose that $\|x\|_\infty \le M$, and let $F \ge \|x\|_p^p$. Then, with probability at least $1-\delta$,*

$$\left| \frac{1}{q} \|x_S\|_p^p - \|x\|_p^p \right| \le \varepsilon F,$$

provided that

$$M^p \le \frac{q\varepsilon^2}{4\log(2/\delta)} F.$$

In particular, taking $F = \|x\|_p^p$ gives $\frac{1}{q}\|x_S\|_p^p = (1 \pm \varepsilon)\|x\|_p^p$.

Proof. The claim is immediate when $x = 0$, so assume $x \neq 0$. Let

$$X = \sum_{i=1}^{d} \sigma_i |x_i|^p, \qquad \sigma_i \sim \frac{1}{q}\,\text{Bernoulli}(q)$$

independently, and set $X_i = \sigma_i |x_i|^p$. Then $\mathrm{E}\, X = \|x\|_p^p$,

$$\mathrm{E}\left(|X_i - |x_i|^p|^2\right) \le \mathrm{E}(X_i^2) = \frac{1}{q}|x_i|^{2p}, \qquad |X_i - |x_i|^p| \le \frac{M^p}{q}.$$

Thus, by the two-sided Bernstein inequality,

$$\Pr\left(\left|X - \|x\|_p^p\right| \ge t\right) \le 2\exp\left(-\frac{t^2 q}{2M^p(\|x\|_p^p + t/3)}\right).$$

Taking $t = \varepsilon F$, and using $\|x\|_p^p \le F$ and $\varepsilon < 1$, gives

$$\Pr\left(\left|X - \|x\|_p^p\right| \ge \varepsilon F\right) \le 2\exp\left(-\frac{\varepsilon^2 q F}{4M^p}\right) \le \delta.$$

□

Using this, we show the following result for F_p moment estimation.

Theorem 3.4. *Fix any constants $p > 2$ and $c > 0$, and let $d \ge 2$. For $\varepsilon \in (0,1)$ satisfying $\varepsilon^{1+2/p} \ge d^{-1/p+c}$, there exists a linear F_p-moment estimation sketch with sketching dimension*

$$O_{p,c}\Big(d^{1-2/p}\,\text{poly}(\log d/\varepsilon)\Big)$$

that can be applied in $O_{p,c}(d)$ time to a vector $x \in \mathbb{Z}^d$ with $O(\log d)$-bit coordinates, given random access to its entries. For every fixed such x, the decoder returns an estimate $\widehat{F}_p$ satisfying

$$\left|\widehat{F}_p - F_p(x)\right| \le \varepsilon F_p(x), \qquad F_p(x) = \sum_{i=1}^{d} |x_i|^p,$$

with probability at least $1 - 1/d$.

Proof. We use the bounded-word input and random-access assumptions in the theorem statement. It suffices to prove the claim for sufficiently large d, depending only on p and c; the remaining bounded dimensions can be handled by the identity sketch, with its dimension absorbed into the stated $O_{p,c}(\cdot)$ bound. The zero vector is handled by the convention that an all-zero sketch decodes to zero, so assume

$$F = \|x\|_p^p > 0.$$

Set

$$\delta = \frac{1}{d}, \qquad \Lambda = \lceil \log_2(2d) \rceil, \qquad q = \Lambda^{-3},$$

and choose

$$\alpha = \frac{q\varepsilon^2}{2^{p+10}\Lambda}, \qquad \eta = \frac{\varepsilon q}{8}, \qquad \beta = \frac{\eta\alpha^{1/p}}{2^{p+1}p}.$$

First, apply the simultaneous ℓ_2 point-estimation construction used for our heavy-hitters sketch, with accuracy $\beta d^{1/p-1/2}$ and failure probability $\delta/2$. Since

$$\|x\|_2 \le d^{1/2-1/p} \|x\|_p ,$$

this gives estimates $\widehat{x}_i$ satisfying

$$|\widehat{x}_i - x_i| \le \beta \|x\|_p \quad \text{simultaneously for all } i \in [d]$$

with probability at least $1 - \delta/2$. The sketch has dimension

$$m_H = O_{p,c}\left(d^{1-2/p}\beta^{-2}\Lambda\right)$$

and application time $O_{p,c}(d)$, provided that $\beta \ge d^{-1/p+c/2}$, which we verify below. We use only one such sketch, independently of all subsequent sampling and moment sketches.

Let

$$K = \min\{d, \lceil 1/\alpha \rceil\},$$

and let H consist of the K largest estimated magnitudes $|\widehat{x}_i|$, breaking ties by index. Set $L = [d] \setminus H$. This definition uses only the coordinate estimates and does not require knowledge of F. The theorem does not impose a decoding-time bound, so the decoder may evaluate all d point estimates.

Condition on any realization of the coordinate-estimation sketch for which all its estimates are correct. We claim that

$$\|x_L\|_\infty \le 2\alpha^{1/p} \|x\|_p .$$

This is immediate if $K = d$. Otherwise, if an omitted coordinate i had magnitude greater than $2\alpha^{1/p} \|x\|_p$, then every $j \in H$ would satisfy

$$|x_j| \ge |\widehat{x}_j| - \beta \|x\|_p \ge |\widehat{x}_i| - \beta \|x\|_p > (2\alpha^{1/p} - 2\beta) \|x\|_p \ge \frac{3}{2}\alpha^{1/p} \|x\|_p .$$

Here we used $\beta \le \alpha^{1/p}/4$. Since $K\alpha \ge 1$, summing the pth powers over H would give a value greater than F, a contradiction.

The same coordinate estimates approximate the moment of every subset of H. Indeed, since $|H| \le 2/\alpha$, for every $A \subseteq H$,

$$\|\widehat{x}_A - x_A\|_p \le \beta(2/\alpha)^{1/p} \|x\|_p \le \frac{\eta}{p2^p} \|x\|_p .$$

Both $\|x_A\|_p$ and $\|\widehat{x}_A\|_p$ are at most $2\|x\|_p$. The reverse triangle inequality and the mean value theorem therefore imply

$$\left| \|\widehat{x}_A\|_p^p - \|x_A\|_p^p \right| \le p(2\|x\|_p)^{p-1} \|\widehat{x}_A - x_A\|_p \le \eta F.$$

We now construct the sampled moment sketches. For the inner primitive, we use Ganguly (2011, Theorem 12). For fixed $p > 2$, its sketch uses at most

$$m_G = O_p\left(d^{1-2/p}\eta^{-2}\Lambda\right)$$

words and has $O_p(\Lambda)$ update time. On every fixed input vector y satisfying the bounded-word assumptions, its decoder returns an estimate Z satisfying

$$|Z - \|y\|_p^p| \le \eta \|y\|_p^p$$

with probability at least $4/5$. Its maintained counters are linear measurements for fixed randomness.

Take an odd number $R = \Theta(\Lambda)$ of repetitions, with a sufficiently large constant. For each $r \in [R]$, independently sample $S_r \subseteq [d]$ by including each coordinate with probability q, and apply an independent copy of the inner sketch to x_{S_r}. The ambient universe remains $[d]$; only the number of processed coordinates is reduced. Let Z_r denote its decoded moment estimate.

Define

$$\widehat{H} = \sum_{i \in H} |\widehat{x}_i|^p, \qquad \widehat{H}'_r = \sum_{i \in H \cap S_r} |\widehat{x}_i|^p,$$

and let the estimate in repetition r be

$$Y_r = \widehat{H} + \frac{Z_r - \widehat{H}'_r}{q}.$$

Thus, the sampled heavy contribution is subtracted only during decoding.

Since H is independent of S_r, we may apply Lemma B.1 to x_L, using the reference value $F = \|x\|_p^p$. Our bound on $\|x_L\|_\infty$ and the choice of α give

$$\|x_L\|_\infty^p \le 2^p \alpha F \le \frac{q(\varepsilon/4)^2}{4\log(200d)} F.$$

Consequently, with probability at least $1 - \delta/100$,

$$\left| \frac{1}{q} \|x_{L \cap S_r}\|_p^p - \|x_L\|_p^p \right| \le \frac{\varepsilon}{4} F.$$

Independently of the coordinate sketch, and conditionally on S_r, the inner moment sketch succeeds with probability at least $4/5$. On its success event,

$$|Z_r - \|x_{S_r}\|_p^p| \le \eta \|x_{S_r}\|_p^p \le \eta F.$$

Combining these estimates with the moment-error bound for subsets of H, we obtain

$$\begin{aligned} |Y_r - F| &\le |\widehat{H} - \|x_H\|_p^p| + \frac{|Z_r - \|x_{S_r}\|_p^p|}{q} + \frac{|\widehat{H}'_r - \|x_{H \cap S_r}\|_p^p|}{q} \\ &\quad + \left| \frac{1}{q} \|x_{L \cap S_r}\|_p^p - \|x_L\|_p^p \right| \\ &\le \left(\eta + \frac{2\eta}{q} + \frac{\varepsilon}{4} \right) F \le \varepsilon F. \end{aligned}$$

To generate the samples without scanning x separately for each repetition, enumerate sampled positions using successive bounded geometric gaps. The exact word-RAM sampler of Bringmann & Friedrich (2013, Theorem 5) generates each gap in $O(1)$ expected time for our rational choice of q. Thus generating one sample, including writing its list of positions, takes $O(1 + qd)$ expected time and does not inspect x.

To obtain a worst-case application-time bound, stop the sample-generation procedure after $C_0(1+qd)$ operations, where C_0 is a sufficiently large absolute constant. By Markov's inequality, this timeout has probability at most $1/100$. If it occurs, use the zero linear map for that repetition and set $Y_r = 0$. Otherwise, the generated list contains $O(1 + qd)$ positions. The timeout depends only on public randomness, not on x. The preceding probability estimates apply to the ideal Bernoulli samples, and the timeout is charged as an additional failure event, without conditioning the sampling distribution on successful completion.

For every fixed successful realization of the shared coordinate sketch, each repetition therefore satisfies $|Y_r - F| \le \varepsilon F$ with probability at least

$$\frac{4}{5} - \frac{\delta}{100} - \frac{1}{100} \ge \frac{3}{4}.$$

The repetitions are independent after fixing the shared coordinate sketch. Hence Hoeffding's inequality gives

$$\Pr\left[\left|\text{median}_{r\in[R]} Y_r - F\right| > \varepsilon F \;\middle|\; \text{the fixed successful coordinate sketch}\right] \le e^{-R/8} \le \frac{\delta}{2},$$

for our choice of R. Adding the coordinate-estimation failure probability $\delta/2$ proves the required overall failure probability at most δ.

It remains to verify the parameters and application time. Our choices give

$$\beta = \Theta_p\left(\varepsilon^{1+2/p}\Lambda^{-(3+4/p)}\right).$$

The assumption $\varepsilon^{1+2/p} \ge d^{-1/p+c}$ therefore implies $\beta \ge d^{-1/p+c/2}$ for sufficiently large d. This justifies the linear-time coordinate-estimation sketch. It also gives $\eta \ge \beta \ge d^{-1/p}$, so the nontrivial parameter regime of the cited moment sketch applies for sufficiently large d.

Including the sample-generation cutoffs, the total cost of generating the samples and processing their coordinates is

$$O_p\left(R(1+qd)\Lambda\right) = O_p\left(\frac{d}{\Lambda} + \Lambda^2\right) = O_p(d).$$

The inner sketches can be initialized in time proportional to their number of counters; their input-independent rescalings are evaluated only at processed coordinates. The total number of inner counters is

$$Rm_G = O_p\left(d^{1-2/p}\varepsilon^{-2}\Lambda^8\right).$$

Moreover, the final coordinate-sketch dimension is

$$m_H = O_{p,c}\left(d^{1-2/p}\varepsilon^{-2-4/p}\Lambda^{7+8/p}\right).$$

Since $\varepsilon < 1$ and

$$d^{1-2/p}\varepsilon^{-2} \le d^{1-2/p}\varepsilon^{-2-4/p} \le d^{1-2c},$$

both Rm_G and m_H are $o(d)$. Thus initialization of the inner sketches also takes $o(d)$ time, while forming the shared coordinate sketch, including its intermediate initialization, takes $O_{p,c}(d)$ time. The total application time is consequently $O_{p,c}(d)$.

Finally, the total sketching dimension is

$$\begin{aligned} m_H + Rm_G &= O_{p,c}\left(d^{1-2/p}\left(\varepsilon^{-2-4/p}\Lambda^{7+8/p} + \varepsilon^{-2}\Lambda^8\right)\right) \\ &= O_{p,c}\left(d^{1-2/p}\text{poly}(\log d/\varepsilon)\right). \end{aligned}$$

For fixed public randomness, the stored measurements consist only of the shared coordinate sketch and the independent linear sketches of subsampled vectors, with zero blocks for timed-out repetitions. The heavy set and all heavy-part corrections are computed only during decoding. Hence the entire construction is a linear sketch. □

B.2 Reshaping and mixed norms

We record a mixed-norm consequence of applying the same p-stable map to the columns of a matrix.

Fix constants $1 < r < p < 2$ and $r' \in (r, p)$, and set $a = r'/r$. Let Z be a standard symmetric p-stable random variable, and write

$$\mu_{p,r} = \mathrm{E}\,|Z|^r.$$

Let $T \in \mathbb{R}^{m\times B}$ have independent entries

$$T_{\ell i} = \frac{Z_{\ell i}}{(m\mu_{p,r})^{1/r}},$$

where the $Z_{\ell i}$ are independent copies of Z.

Lemma B.2. *For every fixed $y \in \mathbb{R}^B$,*

$$\mathrm{E}\,\|Ty\|_r^r = \|y\|_p^r,$$

and

$$\mathrm{E}\left|\|Ty\|_r^r - \|y\|_p^r\right|^a \le C_{p,r,r'} m^{1-a}\,\|y\|_p^{ra},$$

where $C_{p,r,r'}$ depends only on p, r, r'.

Proof. By p-stability and independence of the rows,

$$\|Ty\|_r^r \overset{\mathrm{d}}{=} \frac{\|y\|_p^r}{m}\sum_{\ell=1}^{m}\frac{|Z_\ell|^r}{\mu_{p,r}},$$

where the Z_ℓ are independent copies of Z. This gives the expectation identity. Set $V_\ell = |Z_\ell|^r/\mu_{p,r} - 1$. Then $\mathrm{E}\,V_\ell = 0$ and $\mathrm{E}\,|V_\ell|^a < \infty$, since $ra = r' < p$. As $1 < a < 2$, the von Bahr–Esseen inequality applied to the independent rows gives

$$\mathrm{E}\left|\|Ty\|_r^r - \|y\|_p^r\right|^a \le \frac{2\,\|y\|_p^{ra}}{m^a}\sum_{\ell=1}^{m}\mathrm{E}\,|V_\ell|^a,$$

which proves the claim. □

We now extend the one-vector moment bound to the mixed norm

$$\mathcal{N}_{p,r}(X) = \left(\sum_{j=1}^{s}\|X_j\|_p^r\right)^{1/r}$$

of a fixed matrix $X \in \mathbb{R}^{B\times s}$. Fix $r' \in (r,p)$, let $a = r'/r > 1$, and write

$$D_j = \|TX_j\|_r^r - \|X_j\|_p^r.$$

By the one-vector moment bound,

$$(\mathrm{E}\,|D_j|^a)^{1/a} \le C_{p,r,r'}^{1/a} m^{-(a-1)/a}\,\|X_j\|_p^r.$$

Although the D_j need not be independent, Minkowski's inequality gives

$$\begin{aligned}\left(\mathrm{E}\,|\|TX\|_r^r - \mathcal{N}_{p,r}(X)^r|^a\right)^{1/a} &= \left(\mathrm{E}\left|\sum_{j=1}^{s}D_j\right|^a\right)^{1/a}\\ &\le \sum_{j=1}^{s}(\mathrm{E}\,|D_j|^a)^{1/a}\\ &\le C_{p,r,r'}^{1/a} m^{-(a-1)/a}\mathcal{N}_{p,r}(X)^r.\end{aligned}$$

For $X \neq 0$, Markov's inequality therefore yields

$$\Pr\left[|\|TX\|_r^r - \mathcal{N}_{p,r}(X)^r| > \varepsilon\mathcal{N}_{p,r}(X)^r\right] \le \frac{C_{p,r,r'}}{\varepsilon^a m^{a-1}}.$$

Taking

$$m \ge \left(10C_{p,r,r'}\varepsilon^{-a}\right)^{1/(a-1)}$$

makes this probability at most $1/10$. Taking rth roots then gives

$$\|TX\|_r = (1\pm\varepsilon)\mathcal{N}_{p,r}(X).$$

The case $X = 0$ is immediate. For example, choosing $r' = (r+p)/2$ gives the sufficient row count

$$m = O_{p,r}\left(\varepsilon^{-(p+r)/(p-r)}\right),$$

with a sufficiently large constant.

C Linear Algebraic Problems

C.1 Regression

In this section, we observe that the same idea used to speed up heavy-hitters sketches can be used to speed up sketches for linear regression. The idea is to first apply a CountSketch to reduce the dimension in input-sparsity time, and then apply a fast JL transform only after this first reduction. Repeating this construction a constant number of times lets us boost the success probability while keeping the total sketching time linear, provided that the regression instance is sufficiently overdetermined.

We compose the CountSketch with a fast JL transform. The CountSketch only needs to reduce the ambient dimension far enough so that the fast JL transform can be applied to all d columns of A within the target running time.

Lemma C.1. *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$. Fix $\varepsilon, \delta \in (0,1)$. Let S_i be a CountSketch matrix with $m_S = O\left(\frac{d^2}{\varepsilon\delta}\right)$ rows, and let T_i be an independent fast JL transform from $\mathbb{R}^{m_S}$ to $\mathbb{R}^{m_T}$ with $m_T = O\left(\frac{d}{\varepsilon}\log\frac{d}{\delta}\right)$ rows. Then, with probability at least $1-\delta$, if $x_i = \arg\min_x \|T_iS_i(Ax-b)\|_2^2$, then*

$$\|Ax_i - b\|_2^2 \le (1+\varepsilon)\min_x \|Ax-b\|_2^2 .$$

Moreover, T_iS_iA and T_iS_ib can be formed in time $O\left(\mathrm{nnz}(A) + d\, m_S \log m_S\right)$.

Proof. We first recall a well-known fact, essentially due to Woodruff et al. (2014), that CountSketch is a subspace embedding for $A \in \mathbb{R}^{n\times d}$ when S_i has $O(d^2/\varepsilon^2\delta)$ rows. Thus with probability at least $1-\frac{\delta}{2}$, for every $y \in \mathrm{span}(A,b)$, $\|S_iy\|_2^2 = (1\pm O(\sqrt{\varepsilon}))\,\|y\|_2^2$.

Condition on this event and on the realization of S_i. Since T_i is independent of S_i, the standard fast JL subspace embedding guarantee, with distortion parameter $\sqrt{\varepsilon}$ and failure probability at most $\frac{\delta}{2}$, gives $\|T_iz\|_2^2 = (1\pm O(\sqrt{\varepsilon}))\,\|z\|_2^2$ simultaneously for all $z \in \mathrm{col}(S_i)$. Therefore, with probability at least $1-\delta$,

$$\|T_iS_iy\|_2^2 = (1\pm O(\sqrt{\varepsilon}))\,\|y\|_2^2$$

for all $y \in \mathrm{span}(A,b)$ by a union bound.

It is standard that a $(1\pm O(\sqrt{\varepsilon}))$ subspace embedding for $\mathrm{span}(A,b)$ suffices for a $(1+\varepsilon)$ least-squares regression solution (see Bourgain et al., 2015). Thus the minimizer of the sketched regression problem satisfies

$$\|Ax_i - b\|_2^2 \le (1+\varepsilon)\min_x \|Ax-b\|_2^2 .$$

Finally, applying S_i to A and b takes input-sparsity time. The matrix S_iA has m_S rows and d columns, so applying the fast JL transform T_i to all columns of S_iA, as well as to S_ib, takes $O(d\, m_S \log m_S)$ time. This gives the claimed application time. □

By applying this lemma for a constant number of repetitions and estimating the resulting residuals, we can keep the sketching time at truly $\mathrm{nnz}(A)$, while obtaining failure probability $1/\mathrm{poly}(d)$, at least provided that A is quite tall.

Theorem 3.11. *For $\varepsilon \ge 1/d$, where $c > 0$ is a constant, there exists a $(\varepsilon, \frac{1}{d})$-regression sketch for least squares regression with sketching dimension $O(\frac{d}{\varepsilon}\log d)$, that can be applied to a matrix $A \in \mathbb{R}^{n\times d}$ in $O(\mathrm{nnz}(A) + \frac{1}{\varepsilon}d^{3+c})$ time. In particular, if $\mathrm{nnz}(A) \ge \frac{1}{\varepsilon}d^{3+c}$, the application time is $O(\mathrm{nnz}(A))$.*

Proof. Let $\delta = \frac{1}{d}$ be the desired final failure probability. We run K independent repetitions, where K is a constant to be chosen later. For each $i \in [K]$, sample an independent CountSketch S_i and an independent fast JL transform T_i as in Lemma C.1, with failure parameter $\delta_1 = \delta^{1/K}$. Let x_i be the solution of the sketched regression problem:

$$x_i = \arg\min_x \|T_iS_i(Ax-b)\|_2^2 .$$

By Lemma C.1, each x_i is a $(1+\varepsilon)$ approximate regression solution with probability at least $1-\delta_1$. Say that x_i is good if

$$\|Ax_i - b\|_2^2 \le (1+\varepsilon)\min_x \|Ax-b\|_2^2 .$$

Then the probability that none of the x_i is good is at most $\delta_1^K = \delta$. We now need to select a good candidate among $x_1, \dots, x_K$. However, computing the residual $\|Ax_i - b\|_2$ naively is too slow for our purposes. To do this, we run the $(\varepsilon, \delta/K, \ell_2)$-estimation sketch from Lemma 3.3 in parallel on A and b. Let this sketch be denoted by Q. For each candidate x_i, we estimate the residual $\|Ax_i - b\|_2^2$ using $\|Q(Ax_i - b)\|_2^2$.

Conditioned on the candidates $x_1, \dots, x_K$, the sketch Q is independent of these residual vectors. Hence, with probability at least $1-\delta$, all K residual estimates are simultaneously correct up to a $(1 \pm \varepsilon)$ multiplicative factor. Now condition on this event and on the event that at least one candidate is good. Let j be an index such that x_j is good, and let

$$\hat{i} = \arg\min_{i \in [K]} \|Q(Ax_i - b)\|_2^2 .$$

By the definition of $\hat{i}$, $\|Q(Ax_{\hat{i}} - b)\|_2^2 \le \|Q(Ax_j - b)\|_2^2$. Using the correctness of the residual estimates, this implies

$$\|Ax_{\hat{i}} - b\|_2^2 \le (1 + O(\varepsilon))\,\|Ax_j - b\|_2^2 .$$

Since x_j is good, we obtain

$$\|Ax_{\hat{i}} - b\|_2^2 \le (1 + O(\varepsilon))\min_x \|Ax-b\|_2^2 .$$

Replacing ε by a sufficiently small constant multiple gives a $(1+\varepsilon)$ approximate regression solution. The overall failure probability is at most the probability that all regression sketches fail, plus the probability that one of the residual estimates fails, which is $O(\delta)$ by a union bound. Our overall sketching dimension is as follows, for $\varepsilon \ge \frac{1}{d}$, $\delta = \frac{1}{d}$, and constant K:

$$O\left(\frac{d}{\varepsilon}\log d + \frac{1}{\varepsilon^2}\log d\right) = O\left(\frac{d}{\varepsilon}\log d\right).$$

It remains to bound the time to apply the sketches. By Lemma C.1, applying all K regression sketches takes

$$O\left(K\,\mathrm{nnz}(A) + K\frac{d^3}{\varepsilon\delta_1}\log\frac{d}{\varepsilon\delta_1}\right).$$

The ℓ_2 residual-estimation sketch can be applied to A and b in time

$$O\left(C\,\mathrm{nnz}(A) + \frac{d}{\varepsilon^2}\left(\frac{\delta}{K}\right)^{-1/C}\log\frac{K}{\delta}\right),$$

where C is the boosting parameter in the ℓ_2 estimation sketch. Choosing K and C to be sufficiently large constants depending on c gives $O\left(\mathrm{nnz}(A) + \frac{d^{3+c}}{\varepsilon}\right)$ as desired. □

C.2 Low-Rank Approximation

In this section, we observe that ideas similar to those used to speed up our heavy-hitters sketch can also be applied to linear algebraic problems. We first recall a few preliminary definitions and results that will be helpful here.

Definition C.2. *Let $S \in \mathbb{R}^{m\times d}$ be a sketching matrix. S is said to satisfy the $(\varepsilon, \delta, 2)$-JL moment property if for all unit vectors $x \in \mathbb{R}^d$,*

$$\mathrm{E}\left|\|Sx\|_2^2 - 1\right|^2 \le \varepsilon^2\delta.$$

The following fact is standard; see, for example, Woodruff et al. (2014).

Lemma C.3. *The $(\varepsilon, \delta, 2)$-JL moment property is satisfied by a CountSketch with $O(\frac{1}{\varepsilon^2\delta})$ rows, as well as a fast JL sketch with $O(\frac{1}{\varepsilon^2}\log\frac{1}{\delta})$ rows.*

We also recall the notion of a projection-cost-preserving (PCP) sketch (Musco & Musco, 2020; Cohen et al., 2015), which is well-suited to low-rank approximation problems. One could use other combinations of sketching primitives here, but the explicit conditions given in Musco & Musco (2020) are convenient to check. We simplify the standard definition slightly for our context. In what follows, $A \in \mathbb{R}^{d\times n}$ should be thought of as very wide ($d \ll n$), with columns representing data points or feature vectors.

Definition C.4. *Let $A \in \mathbb{R}^{d\times n}$. The sketching matrix $S \in \mathbb{R}^{n\times m}$ is an (ε, k) projection-cost-preserving sketch for A if*

$$(1-\varepsilon)\left\|A - PA\right\|_F^2 \le \left\|AS - PAS\right\|_F^2 \le (1+\varepsilon)\left\|A - PA\right\|_F^2,$$

for all orthogonal projection matrices $P \in \mathbb{R}^{d\times d}$ of rank at most k.

Next we check that the composition of a CountSketch and a JL sketch has the PCP property, so that we can apply the same trick as for regression to get truly linear sketching times.

Lemma C.5. *Let $A \in \mathbb{R}^{d\times n}$ be fixed, and let $S = S_1 S_2 \in \mathbb{R}^{n\times m}$, where $S_1 \in \mathbb{R}^{n\times r}$ is a CountSketch with $r = O(\frac{k^2}{\varepsilon^2\delta})$ columns and $S_2 \in \mathbb{R}^{r\times m}$ is a fast JL transform with $m = O(\frac{k}{\varepsilon^2}\log\frac{1}{\delta})$ columns. Then, with probability at least $1 - O(\delta)$, S is an (ε, k)-PCP sketch for A.*

Proof. From the definition, the composition of (ε, k)-PCP sketches is an $(O(\varepsilon), k)$-PCP sketch when $\varepsilon < 0.5$. Thus, by Theorem 2 of Musco & Musco (2020), it suffices to check for $i = 1, 2$ that with probability at least $1 - \delta$,

1. S_i is an $(1 \pm \varepsilon)$ distortion oblivious subspace embedding for k-dimensional subspaces,
2. S_i has the $O(\frac{\varepsilon}{\sqrt{k}})$ approximate matrix multiplication property,
3. S_i preserves Frobenius norm to $(1 \pm \varepsilon)$ multiplicative error.

These are all standard facts: for the CountSketch S_1, the first property is from Lemma C.3. The second property follows from Lemma 45 of Woodruff et al. (2014), and the third property follows from the JL moment property (see, e.g., Musco & Musco, 2020). The sketching matrix S_2 satisfies the $(\varepsilon/3, \delta/9^k, 2)$-JL moment property for the stated size (Woodruff et al., 2014), and is immediately a PCP sketch by Musco & Musco (2020). □

Next, we give a version of Lemma 3.3 for estimating the Frobenius norm.

Lemma C.6. *For any $A \in \mathbb{R}^{d\times n}$, $\varepsilon, \delta \in (0, 1)$, and constant $C \ge 1$, there exists an oblivious sketching matrix $Q \in \mathbb{R}^{n\times m}$ with $m = O(\varepsilon^{-2}\log\frac{1}{\delta})$ such that, given AQ, one can estimate $\|A\|_F^2$ within a $(1 \pm \varepsilon)$ factor with probability at least $1 - \delta$. The sketched matrix AQ has $O(d\varepsilon^{-2}\log\frac{1}{\delta})$ entries and can be computed in time*

$$O\left(C\,\mathrm{nnz}(A) + d\varepsilon^{-2}\delta^{-1/C}\log\frac{1}{\delta}\right).$$

Proof. We use the same sketching matrices as in Lemma 3.3, now applied on the right to the rows of A. The only difference is that the sketch must preserve a Frobenius norm rather than the Euclidean norm of a single vector. As above, the composition of CountSketch and a JL sketch satisfies the $(\varepsilon, \delta, 2)$-JL moment property for the dimensions we choose. This property is preserved under direct sums, and therefore gives $\|AQ\|_F^2 = (1 \pm \varepsilon)\|A\|_F^2$ with probability at least $1 - \delta$[4]. The time bound follows by applying the same CountSketch-plus-JL decomposition row-wise: the CountSketch part costs $O(C\,\mathrm{nnz}(A))$, and the final JL part is applied to d reduced rows, giving the stated additional term. □

Theorem 3.12. *Fix any constant $c > 0$. There exists a sketch that, given AS for $A \in \mathbb{R}^{d\times n}$, recovers a rank-$k$ orthogonal projection Π satisfying $\|\Pi A - A\|_F^2 \le (1+\varepsilon)\|A_k - A\|_F^2$ with probability*

[4]One way to formalize this is to use the direct-sum argument from Ahle et al. (2020).

at least $1 - \frac{1}{d}$*, where* A_k *is the optimal rank-*k *approximation to* A*. The sketching dimension is* $O(\frac{kd}{\varepsilon^2} \log d)$*, and the sketch can be applied in time*

$$O_c\left(\operatorname{nnz}(A) + \frac{k^2}{\varepsilon^2} d^{1+c} \log \frac{dk}{\varepsilon}\right).$$

In particular, if $\operatorname{nnz}(A) = \widetilde{\Omega}(\frac{k^2}{\varepsilon^2} d^{1+c})$*, the application time is* $O_c(\operatorname{nnz}(A))$*.*

Proof. Set the target failure probability to $\delta = 1/d$. Let T be a sufficiently large constant depending only on c, and set $\delta_0 = \delta^{1/T}$. Draw independent sketches $R_1, \ldots, R_T$ from Lemma C.5 with failure probability parameter δ_0. Each R_i can be applied in time

$$O\left(\operatorname{nnz}(A) + \frac{dk^2}{\varepsilon^2 \delta_0} \log \frac{dk^2}{\varepsilon^2 \delta_0}\right).$$

With probability at least $1 - \delta$, at least one of $R_1, \ldots, R_T$ is an (ε, k)-PCP sketch for A. For each i, compute from AR_i the projection Π_i returned by the PCP recovery algorithm. If R_i is a valid PCP sketch, then

$$\|\Pi_i A - A\|_F^2 \le (1 + O(\varepsilon)) \|A_k - A\|_F^2,$$

after adjusting constants; see Musco & Musco (2020) for the standard PCP-to-low-rank guarantee.

It remains to select a good candidate among $\Pi_1, \ldots, \Pi_T$. We use an independent Frobenius-estimation sketch Q from Lemma C.6, with distortion parameter ε and failure probability δ/T. Given AQ, for each candidate projection we estimate $\|\Pi_i A - A\|_F^2$ by $\|\Pi_i AQ - AQ\|_F^2$. Conditioned on the candidate projections, Q is independent of the residual matrices $\Pi_i A - A$, so by a union bound all T residual estimates are correct up to a $(1 \pm \varepsilon)$ factor with probability at least $1 - \delta$. Let $\hat{i}$ be the index with the smallest sketched residual. If j is an index for which R_j is a valid PCP sketch, then the correctness of the residual estimates gives

$$\|\Pi_{\hat{i}} A - A\|_F^2 \le (1 + O(\varepsilon)) \|\Pi_j A - A\|_F^2 \le (1 + O(\varepsilon)) \|A_k - A\|_F^2.$$

Replacing ε by a sufficiently small constant multiple gives the desired approximation, and the total failure probability is $O(\delta)$. The total sketching dimension of the PCP sketches is $O_c\left(\frac{kd}{\varepsilon^2} \log d\right)$. The Frobenius-estimation sketch uses only $O(d\varepsilon^{-2} \log(T/\delta))$ additional entries, which is lower order for $k \ge 1$. The time to apply all PCP sketches is

$$O\left(T \operatorname{nnz}(A) + T \frac{dk^2}{\varepsilon^2 \delta_0} \log \frac{dk^2}{\varepsilon^2 \delta_0}\right) = O_c\left(\operatorname{nnz}(A) + \frac{k^2}{\varepsilon^2} d^{1+c} \log \frac{dk}{\varepsilon}\right),$$

where T is chosen so that $\delta_0^{-1} = d^{1/T} \le d^c$. The Frobenius-estimation sketch can be applied in time $O\left(C \operatorname{nnz}(A) + d\varepsilon^{-2} (T/\delta)^{1/C} \log \frac{T}{\delta}\right)$, and choosing C sufficiently large makes this lower order than the displayed PCP-sketching time. This proves the theorem. □