JMLR: Workshop and Conference Proceedings vol 30 (2013) 1–22

Subspace Embeddings and ℓ_p-Regression Using Exponential Random Variables

David P. Woodruff DPWOODRU@US.IBM.COM
IBM Research Almaden

Qin Zhang QINZHANG@CSE.UST.HK
IBM Research Almaden

Abstract

Oblivious low-distortion subspace embeddings are a crucial building block for numerical linear algebra problems. We show for any real $p, 1 \le p < \infty$, given a matrix $M \in \mathbb{R}^{n \times d}$ with $n \gg d$, with constant probability we can choose a matrix Π with $\max(1, n^{1-2/p})\text{poly}(d)$ rows and n columns so that simultaneously for all $x \in \mathbb{R}^d$, $\|Mx\|_p \le \|\Pi Mx\|_\infty \le \text{poly}(d)\|Mx\|_p$. Importantly, ΠM can be computed in the optimal $O(\text{nnz}(M))$ time, where $\text{nnz}(M)$ is the number of non-zero entries of M. This generalizes all previous oblivious subspace embeddings which required $p \in [1, 2]$ due to their use of p-stable random variables. Using our matrices Π, we also improve the best known distortion of oblivious subspace embeddings of ℓ_1 into ℓ_1 with $\tilde{O}(d)$ target dimension in $O(\text{nnz}(M))$ time from $\tilde{O}(d^3)$ to $\tilde{O}(d^2)$, which can further be improved to $\tilde{O}(d^{3/2}) \log^{1/2} n$ if $d = \Omega(\log n)$, answering a question of Meng and Mahoney (STOC, 2013).

We apply our results to ℓ_p-regression, obtaining a $(1+\epsilon)$-approximation in $O(\text{nnz}(M) \log n)+\text{poly}(d/\epsilon)$ time, improving the best known $\text{poly}(d/\epsilon)$ factors for every $p \in [1, \infty) \setminus \{2\}$. If one is just interested in a $\text{poly}(d)$ rather than a $(1 + \epsilon)$-approximation to ℓ_p-regression, a corollary of our results is that for all $p \in [1, \infty)$ we can solve the ℓ_p-regression problem without using general convex programming, that is, since our subspace embeds into ℓ_∞ it suffices to solve a linear programming problem. Finally, we give the first protocols for the distributed ℓ_p-regression problem for every $p \ge 1$ which are nearly optimal in communication and computation.

1. Introduction

An oblivious subspace embedding with distortion κ is a distribution over linear maps $\Pi : \mathbb{R}^n \to \mathbb{R}^t$ for which for any fixed d-dimensional subspace of $\mathbb{R}^n$, represented as the column space of an $n \times d$ matrix M, with constant probability, $\|Mx\|_p \le \|\Pi Mx\|_p \le \kappa\|Mx\|_p$ simultaneously for all vectors $x \in \mathbb{R}^d$. The goal is to minimize t, κ, and the time to compute $\Pi \cdot M$. For a vector v, $\|v\|_p = (\sum_{i=1}^n |v_i|^p)^{1/p}$ is its p-norm.

Oblivious subspace embeddings have proven to be an essential ingredient for quickly and approximately solving numerical linear algebra problems. One of the canonical problems is regression, which is well-studied in the learning community, see [13, 15, 16, 20] for some recent advances. Sárlos [28] observed that oblivious subspace embeddings could be used to approximately solve least squares regression and low rank approximation, and he used fast Johnson-Lindenstrauss transforms [2, 1] to obtain the fastest known algorithms for these problems at the time. Optimizations to this in the streaming model are in [10, 19].

As an example, in least squares regression, one is given an $n \times d$ matrix M which is usually overconstrained, i.e., $n \gg d$, as well as a vector $b \in \mathbb{R}^n$. The goal is to output $x^* = \text{argmin}_x\|Mx - b\|_2$, that is, to find the vector x^* so that Mx^* is the (Euclidean) projection of b onto the column space of M. This can be solved exactly in $O(nd^2)$ time. Using fast Johnson-Lindenstrauss transforms, Sárlos was able to find a vector x' with $\|Mx' - b\|_2 \le (1 + \epsilon)\|Mx^* - b\|_2$ in $O(nd \log d) + \text{poly}(d/\epsilon)$ time, providing a substantial improvement. The application of oblivious subspace embeddings (to the space spanned by the columns of M together with b) is immediate: given M and b, compute ΠM and Πb, and solve the problem $\min_x \|\Pi M x - \Pi b\|_2$. If $\kappa = (1+\epsilon)$ and $t \ll n$, one obtains a relative error approximation by solving a much smaller instance of regression.

Another line of work studied ℓ_p-regression for $p \neq 2$. One is given an $n \times d$ matrix M and an $n \times 1$ vector b, and one seeks $x^* = \text{argmin}_x \|Mx - b\|_p$. For $1 \leq p < 2$, this provides a more robust form of regression than least-squares, since the solution is less sensitive to outliers. For $2 < p \leq \infty$, this is even more sensitive to outliers, and can be used to remove outliers. While ℓ_p-regression can be solved in poly(n) time for every $1 \leq p \leq \infty$ using convex programming, this is not very satisfying if $n \gg d$. For $p = 1$ and $p = \infty$ one can use linear programming to solve these problems, though for $p = 1$ the complexity will still be superlinear in n. Clarkson [9] was the first to achieve an $n \cdot \text{poly}(d)$ time algorithm for ℓ_1-regression, which was then extended to ℓ_p-regression for every $1 \leq p \leq \infty$ with the same running time [14].

The bottleneck of these algorithms for ℓ_p-regression was a preprocessing step, in which one well-conditions the matrix M by choosing a different basis for its column space. Sohler and Woodruff [29] got around this for the important case of $p = 1$ by designing an oblivious subspace embedding Π for which $\|Mx\|_1 \leq \|\Pi M x\|_1 = O(d \log d)\|Mx\|_1$ in which Π has $O(d \log d)$ rows. Here, Π was chosen to be a matrix of Cauchy random variables. Instead of running the expensive conditioning step on M, it is run on ΠM, which is much smaller. One obtains a $d \times d$ change of basis matrix R^{-1}. Then one can show the matrix $\Pi M R^{-1}$ is well-conditioned. This reduced the running time for ℓ_1-regression to $nd^{\omega-1} + \text{poly}(d/\epsilon)$, where $\omega < 3$ is the exponent of matrix multiplication. The dominant term is the $nd^{\omega-1}$, which is the cost of computing ΠM when Π is a matrix of Cauchy random variables.

In [12], Clarkson et. al combined the ideas of Cauchy random variables and Fast Johnson Lindenstrauss transforms to obtain a more structured family of subspace embeddings, referred to as the FCT1 in their paper, thereby improving the running time for ℓ_1-regression to $O(nd \log n) + \text{poly}(d/\epsilon)$. An alternate construction, referred to as the FCT2 in their paper, gave a family of subspace embeddings that was obtained by partitioning the matrix M into $n/\text{poly}(d)$ blocks and applying a fast Johnson Lindenstrauss transform on each block. Using this approach, the authors were also able to obtain an $O(nd \log n) + \text{poly}(d/\epsilon)$ time algorithm for ℓ_p-regression for every $1 \leq p \leq \infty$.

While the above results are nearly optimal for dense matrices, one could hope to do better if the number of non-zero entries of M, denoted nnz(M), is much smaller than nd. Indeed, M is often a sparse matrix, and one could hope to achieve a running time of $O(\text{nnz}(M)) + \text{poly}(d/\epsilon)$. Clarkson and Woodruff [11] designed a family of sparse oblivious subspace embeddings Π with poly(d/ϵ) rows, for which $\|Mx\|_2 \leq \|\Pi M x\|_2 \leq (1+\epsilon)\|Mx\|_2$ for all x. Importantly, the time to compute ΠM is only nnz(M), that is, proportional to the sparsity of the input matrix. The poly(d/ϵ) factors were optimized by Meng and Mahoney [22], Nelson and Nguyen [25], and Miller and Peng [24]. Combining this idea with that in the FCT2, they achieved running time $O(\text{nnz}(M) \log n) + \text{poly}(d/\epsilon)$ for ℓ_p-regression for any constant p, $1 \leq p < \infty$.

Meng and Mahoney [22] gave an alternate subspace embedding family to solve the ℓ_p-regression problem in $O(\text{nnz}(M) \log n) + \text{poly}(d/\epsilon)$ time for $1 \leq p < 2$. One feature of their construction is that the number of rows in the subspace embedding matrix Π is only poly(d), while that of Clarkson and Woodruff [11] for $1 \leq p < 2$ is $n/\text{poly}(d)$. This feature is important in the distributed setting, for which there are multiple machines, each holding a subset of the rows of M, who wish to solve an ℓ_p-regression problem by communicating with a central server. The natural solution is to use shared randomness to agree upon an embedding matrix Π, then apply Π locally to each of their subsets of rows, then add up the sketches using the linearity of Π. The communication is proportional to the number of rows of Π. This makes the algorithm of Meng and Mahoney more communication-efficient, since they achieve poly(d/ϵ) communication. However, one drawback of the construction of Meng and Mahoney is that their solution only works for $1 \le p < 2$. This is inherent since they use p-stable random variables, which only exist for $p \le 2$.

1.1. Our Results

In this paper, we improve all previous low-distortion oblivious subspace embedding results for every $p \in [1, \infty)\backslash\{2\}$. We note that the case $p = 2$ is already resolved in light of [11, 22, 25]. All results hold with arbitrarily large constant probability. γ is an arbitrarily small constant. In all results ΠM can be computed in $O(\text{nnz}(M))$ time (for the third result, we assume that $\text{nnz}(M) \ge d^{2+\gamma}$).

- A matrix $\Pi \in \mathbb{R}^{O(n^{1-2/p}\log n(d\log d)^{1+2/p}+d^{5+4p})\times n}$ for $p > 2$ such that given $M \in \mathbb{R}^{n\times d}$, for $\forall x \in \mathbb{R}^d$,

$$\Omega(1/(d\log d)^{1/p}) \cdot \|Mx\|_p \le \|\Pi Mx\|_\infty \le O((d\log d)^{1/p}) \cdot \|Mx\|_p .$$

- A matrix $\Pi \in \mathbb{R}^{O(d^{1+\gamma})\times n}$ for $1 \le p < 2$ such that given $M \in \mathbb{R}^{n\times d}$, for $\forall x \in \mathbb{R}^d$,

$$\Omega\left(\max\left\{1/(d\log d\log n)^{\frac{1}{p}-\frac{1}{2}}, 1/(d\log d)^{1/p}\right\}\right)\cdot\|Mx\|_p \le \|\Pi Mx\|_2 \le O((d\log d)^{1/p})\cdot\|Mx\|_p .$$

  Note that since $\|\Pi Mx\|_\infty \le \|\Pi Mx\|_2 \le O(d^{(1+\gamma)/2})\,\|\Pi Mx\|_\infty$, we can always replace the 2-norm estimator by the ∞-norm estimator with the cost of another $d^{(1+\gamma)/2}$ factor in the distortion.

- A matrix $\Pi \in \mathbb{R}^{O(d\log^{O(1)} d)\times n}$ such that given $M \in \mathbb{R}^{n\times d}$, for $\forall x \in \mathbb{R}^d$,

$$\Omega\left(\max\left\{1/(d\log d), 1/\sqrt{d\log d\log n}\right\}\right)\cdot\|Mx\|_1 \le \|\Pi Mx\|_1 \le O(d\log^{O(1)} d)\cdot\|Mx\|_1 .$$

  In [22] the authors asked whether a distortion $\tilde{O}(d^3)$ [1] is optimal for $p = 1$ for mappings ΠM that can be computed in $O(\text{nnz}(M))$ time. Our result shows that the distortion can be further improved to $\tilde{O}(d^2)$, and if one also has $d > \log n$, even further to $\tilde{O}(d^{3/2})\log^{1/2} n$. Our embedding also improves the $\tilde{O}(d^{2+\gamma})$ distortion of the much slower [12]. In Table 1 we compare our result with previous results for ℓ_1 oblivious subspace embeddings. Our lower distortion embeddings for $p = 1$ can also be used in place of the $\tilde{O}(d^3)$ distortion embedding of [22] in the context of quantile regression [30].

Our oblivious subspace embeddings directly lead to improved $(1+\epsilon)$-approximation results for ℓ_p-regression for every $p \in [1,\infty)\backslash\{2\}$. We further implement our algorithms for ℓ_p-regression in a distributed setting, where we have k machines and a centralized server. The sites want to solve the regression problem via communication. We state both the communication and the time required of our distributed ℓ_p-regression algorithms. One can view the time complexity of a distributed algorithm as the sum of the time complexities of all sites including the centralized server (see Section 5 for details).

Given an ℓ_p-regression problem specified by $M \in \mathbb{R}^{n\times(d-1)}, b \in \mathbb{R}^n, \epsilon > 0$ and p, let $\bar{M} = [M, -b] \in \mathbb{R}^{n\times d}$. Let $\phi(t,d)$ be the time of solving ℓ_p-regression problem on t vectors in d dimensions.

- For $p > 2$, we obtain a distributed algorithm with communication $\tilde{O}\left(kn^{1-2/p}d^{2+2/p} + d^{4+2p}/\epsilon^2\right)$ and running time $\tilde{O}\left(\text{nnz}(\bar{M}) + (k+d^2)(n^{1-2/p}d^{2+2/p} + d^{6+4p}) + \phi(\tilde{O}(d^{3+2p}/\epsilon^2), d)\right)$.

- For $1 \le p < 2$, we obtain a distributed algorithm with communication $\tilde{O}\left(kd^{2+\gamma} + d^5 + d^{3+p}/\epsilon^2\right)$ and running time $\tilde{O}\left(\text{nnz}(\bar{M}) + kd^{2+\gamma} + d^{7-p/2} + \phi(\tilde{O}(d^{2+p}/\epsilon^2), d\right)$.

1. We use $\tilde{O}(f)$ to denote a function of the form $f\cdot\log^{O(1)}(f)$.

	Time	Distortion	Dimemsion
[29]	$nd^{\omega-1}$	$\tilde{O}(d)$	$\tilde{O}(d)$
[12]	$nd \log d$	$\tilde{O}(d^{2+\gamma})$	$\tilde{O}(d^5)$
[11] + [25]	$\text{nnz}(A) \log n$	$\tilde{O}\left(d^{(x+1)/2}\right)$ $(x \geq 1)$	$\tilde{O}(n/d^x)$
[11] + [12] + [25]	$\text{nnz}(A) \log n$	$\tilde{O}(d^3)$	$\tilde{O}(d)$
[11] + [29] + [25]	$\text{nnz}(A) \log n$	$\tilde{O}(d^{1+\omega/2})$	$\tilde{O}(d)$
[22]	$\text{nnz}(A)$	$\tilde{O}(d^3)$	$\tilde{O}(d^5)$
[22] + [25]	$\text{nnz}(A) + \tilde{O}(d^6)$	$\tilde{O}(d^3)$	$\tilde{O}(d)$
This paper	$\text{nnz}(A) + \tilde{O}(d^{2+\gamma})$	$\tilde{O}(d^2)$	$\tilde{O}(d)$
	$\text{nnz}(A) + \tilde{O}(d^{2+\gamma})$	$\tilde{O}(d^{3/2}) \log^{1/2} n$	$\tilde{O}(d)$

Table 1: Results for ℓ_1 oblivious subspace embeddings. $\omega < 3$ is the exponent of matrix multiplication. γ is an arbitrarily small constant.

We comment on several advantages of our algorithms over standard iterative methods for solving regression problems. We refer the reader to Section 4.5 of the survey [21] for more details.

- In our algorithm, there is no assumption on the input matrix M, i.e., we do not assume it is well-conditioned. Iterative methods are either much slower than our algorithms if the condition number of M is large, or would result in an additive ϵ approximation instead of the relative error ϵ approximation that we achieve.
- Our work can be used in conjunction with other ℓ_p-regression algorithms. Namely, since we find a well-conditioned basis, we can run iterative methods on our well-conditioned basis to speed them up.

1.2. Our Techniques

Meng and Mahoney [22] achieve $O(\text{nnz}(M) \log n) + \text{poly}(d)$ time for ℓ_p-regression with sketches of the form $S \cdot D \cdot M$, where S is a $t \times n$ hashing matrix for $t = \text{poly}(d)$, that is, a matrix for which in each column there is a single randomly positioned entry which is randomly either 1 or -1, and D is a diagonal matrix of p-stable random variables. The main issues with using p-stable random variables X are that they only exist for $1 \leq p \leq 2$, and that the random variable $|X|^p$ is heavy-tailed *in both directions*.

We replace the p-stable random variable with the reciprocal of an *exponential random variable*. Exponential random variables have stability properties with respect to the minimum operation, that is, if $u_1, \ldots, u_n$ are exponentially distributed and $\lambda_i > 0$ are scalars, then $\min\{u_1/\lambda_1, \ldots, u_n/\lambda_n\}$ is distributed as u/λ, where $\lambda = \sum_i \lambda_i$. This property was used to estimate the p-norm of a vector, $p > 2$, in an elegant work of Andoni [3]. In fact, by replacing the diagonal matrix D in the sketch of [22] with a diagonal matrix with entries $1/u_i^{1/p}$ for exponential random variables u_i, the sketch coincides with the sketch of Andoni, up to the setting of t. Importantly, this new setting of D has no restriction on $p \in [1, \infty)$. We note that while Andoni's analysis for vector norms requires the variance of $1/u_i^{1/p}$ to exist, which requires $p > 2$, in our setting this restriction can be removed. If $X \sim 1/u^{1/p}$, then X^p is only heavy-tailed *in one direction*, while the lower tail is exponentially decreasing. This results in a simpler analysis than [22] for $1 \leq p < 2$ and an improved distortion. The analysis of the expansion follows from the properties of a well-conditioned basis and is by now standard [29, 22, 12], while for the contraction by observing that S is an ℓ_2-subspace embedding, for any fixed x, $\|SDMx\|_1 \geq \|SDMx\|_2 \geq \frac{1}{2}\|DMx\|_2 \geq \frac{1}{2}\|DMx\|_\infty \sim \|Mx\|_1/(2u)$, where u is an exponential random variable. Given the exponential tail of u, the bound for all x follows from a standard net argument. While this already improves the distortion of [22], a more refined analysis gives a distortion of $\tilde{O}(d^{3/2})\log^{1/2} n$ provided $d > \log n$.

For $p > 2$, we need to embed our subspace into ℓ_∞. A feature is that it implies one can obtain a $\mathrm{poly}(d)$-approximation to ℓ_p-regression by solving an ℓ_∞-regression problem, in $O(\mathrm{nnz}(M)) + \mathrm{poly}(d)$ time. As ℓ_∞-regression can be solved with linear programming, this may result in significant practical savings over convex program solvers for general p. This is also why we use the ℓ_∞-estimator for vector p-norms rather than the estimators of previous works [18, 4, 6, 8] which were not norms, and therefore did not have efficient optimization procedures, such as finding a well-conditioned basis, in the sketch space. Our embedding is into $n^{1-2/p}\mathrm{poly}(d)$ dimensions, whereas previous work was into $n/\mathrm{poly}(d)$ dimensions. This translates into near-optimal communication and computation protocols for distributed ℓ_p-regression for every p. A parallel least squares regression solver LSRN was developed in [23], and the extension to $1 \le p < 2$ was a motivation of [22]. Our result gives the analogous result for every $2 < p < \infty$, which is near-optimal in light of an $\Omega(n^{1-2/p})$ sketching lower bound for estimating the p-norm of a vector over the reals [27].

2. Preliminaries

In this paper we only consider the real RAM model of computation, and state our running times in terms of the number of arithmetic operations.

Given a matrix $M \in \mathbb{R}^{n \times d}$, let $M_1, \ldots, M_d$ be the columns of M, and $M^1, \ldots, M^n$ be the rows of M. Define $\ell_i = \left\|M^i\right\|_p$ $(i = 1, \ldots, n)$, where the ℓ_i^p are known as the *leverage scores* of M. Let $\mathrm{range}(M) = \{y \mid y = Mx, x \in \mathbb{R}^d\}$. W.l.o.g., we constrain $\|x\|_1 = 1, x \in \mathbb{R}^d$; by scaling our results will hold for all $x \in \mathbb{R}^d$. Define $\|M\|_p$ to be the element-wise ℓ_p norm of M. That is, $\|M\|_p = (\sum_{i\in[d]} \|M_i\|_p^p)^{1/p} = (\sum_{j\in[n]} \left\|M^j\right\|_p^p)^{1/p}$.

Let $[n] = \{1, \ldots, n\}$. Let ω denote the exponent of matrix multiplication.

2.1. Well-Conditioning of A Matrix

We introduce two definitions on the well-conditioning of matrices.

Definition 1 **((α, β, p)-well-conditioning [14])** *Given a matrix $M \in \mathbb{R}^{n \times d}$ and $p \in [1, \infty)$, let q be the dual norm of p, that is, $1/p + 1/q = 1$. We say M is* (α, β, p)-well-conditioned *if (1) $\|x\|_q \le \beta \|Mx\|_p$ for any $x \in \mathbb{R}^d$, and (2) $\|M\|_p \le \alpha$. Define $\Delta'_p(M) = \alpha\beta$.*

It is well known that the Auerbach basis [5] (denoted by A throughout this paper) for a d-dimensional subspace $(\mathbb{R}^n, \|\cdot\|_p)$ is $(d^{1/p}, 1, p)$-well-conditioned. Thus by definition we have $\|x\|_q \le \|Ax\|_p$ for any $x \in \mathbb{R}^d$, and $\|A\|_p \le d^{1/p}$. In addition, the Auerbach basis also has the property that $\|A_i\|_p = 1$ for all $i \in [d]$.

Definition 2 **(ℓ_p-conditioning [12])** *Given a matrix $M \in \mathbb{R}^{n \times d}$ and $p \in [1, \infty)$, define $\zeta_p^{\max}(M) = \max_{\|x\|_2 \le 1} \|Mx\|_p$ and $\zeta_p^{\min}(M) = \min_{\|x\|_2 \ge 1} \|Mx\|_p$. Define $\Delta_p(M) = \zeta_p^{\max}(M)/\zeta_p^{\min}(M)$ to be the* ℓ_p-norm condition number *of M.*

The following lemma states the relationship between the two definitions.

Lemma 3 **([14])** *Given a matrix $M \in \mathbb{R}^{n \times d}$ and $p \in [1, \infty)$, we have*

$$d^{-|1/2-1/p|}\Delta_p(M) \le \Delta'_p(M) \le d^{\max\{1/2,1/p\}}\Delta_p(M).$$

2.2. Oblivious Subspace Embeddings

An oblivious subspace embedding (OSE) for the Euclidean norm, given a parameter d, is a distribution $\mathcal{D}$ over $m \times n$ matrices such that for any d-dimensional subspace $\mathcal{S} \subset \mathbb{R}^n$, with probability 0.99 over the choice of $\Pi \sim \mathcal{D}$, we have

$$1/2 \cdot \|x\|_2 \leq \|\Pi x\|_2 \leq 3/2 \cdot \|x\|_2 , \quad \forall x \in \mathcal{S}.$$

Note that OSE's only work for the 2-norm, while in this paper we get similar results for ℓ_p-norms for all $p \in [1, \infty) \backslash \{2\}$. Two important parameters that we want to minimize in the construction of OSE's are: (1) The number of rows of Π, that is, m. This is the dimension of the embedding. (2) The number of non-zero entries in the columns of Π, denoted by s. This affects the running time of the embedding.

In [25], buiding upon [11], several OSE constructions are given. In particular, they show that there exist OSE's with $(m, s) = \left(O(d^2), 1\right)$ and $(m, s) = \left(O(d^{1+\gamma}), O(1)\right)$ for any constant $\gamma > 0$ and $(m, s) = (\tilde{O}(d), \log^{O(1)} d)$.

2.3. Distributions

Given two random variables X, Y, we write $X \simeq Y$ if X and Y have the same distribution.

p-stable Distribution. We say a distribution $\mathcal{D}_p$ is p-stable, if for any vector $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$ and $X_1, \ldots, X_n \stackrel{i.i.d.}{\sim} \mathcal{D}_p$, we have $\sum_{i \in [n]} \alpha_i X_i \simeq \|\alpha\|_p X$, where $X \sim \mathcal{D}_p$. It is well-known that p-stable distribution exists if and only if $p \in [1, 2]$ (see. e.g., [17]). For $p = 2$ it is the Gaussian distribution and for $p = 1$ it is the Cauchy distribution. We say a random variable X is p-stable if X is chosen from a p-stable distribution.

Exponential Distribution. An exponential distribution has support $x \in [0, \infty)$, probability density function (PDF) $f(x) = e^{-x}$ and cumulative distribution function (CDF) $F(x) = 1 - e^{-x}$. We say a random variable X is exponential if X is chosen from the exponential distribution.

Property 1 *The exponential distribution has the following properties.*

1. *(**max stability**) If $u_1, \ldots, u_n$ are exponentially distributed, and $\alpha_i > 0$ $(i = 1, \ldots, n)$ are real numbers, then $\max\{\alpha_1/u_1, \ldots, \alpha_n/u_n\} \simeq \left(\sum_{i \in [n]} \alpha_i\right)/u$, where u is exponential.*
2. *(**lower tail bound**) For any X that is exponential, there exist absolute constants c_e, c'_e such that, $\min\{0.5, c'_e t\} \leq \mathbf{Pr}[X \leq t] \leq c_e t, \ \forall t \geq 0$.*

The second property holds since the median of the exponential distribution is the constant $\ln 2$ (that is, $\mathbf{Pr}[x \leq \ln 2] = 50\%$), and the PDFs on $x = 0, x = \ln 2$ are $f(0) = 1, f(\ln 2) = 1/2$, differing by a factor of 2. Here we use that the PDF is monotone decreasing.

Given two random variables X, Y chosen from two probability distributions, we say $X \succeq Y$ if for $\forall t \in \mathbb{R}$ we have $\mathbf{Pr}[X \geq t] \geq \mathbf{Pr}[Y \geq t]$. The following lemma shows a relationship between the p-stable distribution and the exponential distribution. The proof can be found in Appendix A.1.

Lemma 4 *For any $p \in [1, 2)$, there exists a constant κ_p such that $|X_p| \succeq \kappa_p \cdot 1/U^{1/p}$, where X_p is p-stable and U is an exponential.*

The following lemma characterizes the sum of inverse exponentials. See Appendix A.2 for the proof.

Lemma 5 *Let $u_1, \dots, u_d$ be d exponentials. Let $X = \sum_{i \in [d]} 1/u_i$. Then, for any $t > 1$.*

$$\mathbf{Pr}[X > td/\kappa_1] \le (1 + o(1)) \log(td)/t,$$

where κ_1 is defined in Lemma 4.

Conventions. In the paper we will define several events $\mathcal{E}_0, \mathcal{E}_1, \dots$ in the early analysis, which we will condition on in the later analysis. Each of these events holds with probability 0.99, and there will be no more than ten of them. Thus by a union bound all of them hold simultaneously with probability 0.9. Therefore these conditions will not affect our overall error probability by more than 0.1.

Global Parameters. We set a few parameters which will be used throughout the paper: $\rho = c_1 d \log d$; $\iota = 1/(2\rho^{1/p})$; $\eta = c_2 d \log d \log n$; $\tau = \iota/(d\eta)$.

3. p-norm with $p > 2$

3.1. Algorithm

We set the subspace embedding matrix $\Pi = SD$, where $D \in \mathbb{R}^{n \times n}$ is a diagonal matrix with $1/u_1^{1/p}, \dots, 1/u_n^{1/p}$ on the diagonal such that all u_i $(i = 1, 2, \dots, n)$ are i.i.d. exponentials. And S is an (m, s)-OSE with $(m, s) = \left(6n^{1-2/p}\eta/\iota^2 + d^{5+4p}, 1\right)$. More precisely, we pick random hash functions $h : [n] \to [m]$ and $\sigma : [n] \to \{-1, 1\}$. For each $i \in [n]$, we set $S_{h(i),i} = \sigma(i)$. Since $m = \omega(d^2)$, by [25] such an S is an OSE.

3.2. Analysis

In this section we prove the following Theorem.

Theorem 6 *Let $A \in \mathbb{R}^{d \times n}$ be an Auerbach basis of a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_p)$. Given the above choices of $\Pi \in \mathbb{R}^{(6n^{1-2/p}\eta/\iota^2 + d^{5+4p}) \times n}$, for any $p > 2$ we have*

$$\Omega(1/(d \log d)^{1/p}) \cdot \|Ax\|_p \le \|\Pi A x\|_\infty \le O((d \log d)^{1/p}) \cdot \|Ax\|_p\,, \quad \forall x \in \mathbb{R}^d.$$

Remark 7 *Note that since the inequality holds for all $x \in \mathbb{R}^d$, this theorem also holds if we replace the Auerbach basis A by any matrix M whose column space is a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_p)$.*

Property 2 *Let $A \in \mathbb{R}^{d \times n}$ be a $(d^{1/p}, 1, p)$-well-conditioned Auerbach basis. For an $x \in \mathbb{R}^d$, let $y = Ax \in$ range$(A) \subseteq \mathbb{R}^n$. Each such y has the following properties. Recall that we can assume $\|x\|_1 = 1$.*

1. $\|y\|_p \le \sum_{i \in d} \|A_i\|_p \cdot |x_i| = \|x\|_1 = 1.$
2. $\|y\|_p = \|Ax\|_p \ge \|x\|_q \ge \|x\|_1 / d^{1-1/q} = 1/d^{1/p}.$
3. *For all $i \in [n]$,* $|y_i| = \left|(A^i)^T x\right| \le \left\|A^i\right\|_1 \cdot \|x\|_\infty \le d^{1-1/p} \left\|A^i\right\|_p \cdot \|x\|_1 = d^{1-1/p} \ell_i.$

Let H be the set of indices $i \in [n]$ such that $\ell_i / u_i^{1/p} \ge \tau$. Let $L = [n] \backslash H$. Then

$$\begin{aligned}
\mathbf{E}[|H|] &= \textstyle\sum_{i \in [n]} \mathbf{Pr}[\ell_i / u_i^{1/p} \ge \tau] \\
&= \textstyle\sum_{i \in [n]} \mathbf{Pr}[u_i \le \ell_i^p / \tau^p] \\
&\le \textstyle\sum_{i \in [n]} c_e \ell_i^p / \tau^p \quad \text{(Property 1)} \\
&\le c_e d / \tau^p. \quad (\textstyle\sum_{i \in [n]} \ell_i^p = \|A\|_p^p \le d)
\end{aligned}$$

Therefore with probability 0.99, we have $|H| \leq 100c_e d/\tau^p$. Let $\mathcal{E}_0$ denote this event, which we will condition on in the rest of the proof.

For a $y \in \text{range}(A)$, let $w_i = 1/u_i^{1/p} \cdot y_i$. For all $i \in L$, we have

$$|w_i| = 1/u_i^{1/p} \cdot |y_i| \leq d^{1-1/p}\ell_i/u_i^{1/p} < d^{1-1/p}\tau \leq d^{1-1/p}\tau \cdot d^{1/p} \|y\|_p = d\tau \|y\|_p .$$

In the first and third inequalities we use Property 2, and the second inequality follows from the definition of L. For $j \in [m]$, let

$$z_j(y) = \sum_{i:(i \in L) \wedge (h(i)=j)} \sigma(j) \cdot w_i.$$

Define $\mathcal{E}_1$ to be the event that for all $i, j \in H$, we have $h(i) \neq h(j)$. The rest of the proof conditions on $\mathcal{E}_1$. The following lemma is implicit in [3]. See Section B.1 for a sketch of the proof.

Lemma 8 ([3]) *1. Assuming that $\mathcal{E}_0$ holds, $\mathcal{E}_1$ holds with probability at least* 0.99.

2. For any $\iota > 0$, for all $j \in [m]$,

$$\mathbf{Pr}[|z_j(y)| \geq \iota \|y\|_p] \leq \exp\left[-\frac{\iota^2/2}{n^{1-2/p}/m + \iota d\tau/3}\right] = e^{-\eta}.$$

3.2.1. No Overestimation

By Lemma 8 we have that with probability $(1 - m \cdot d \cdot e^{-\eta}) \geq 0.99$, $\max_{j \in [m]} z_j(A_i) \leq \iota \|A_i\|_p = \iota$ for all $i \in [d]$. Let $\mathcal{E}_2$ denote this event, which we condition on. Note that $A_i \in \text{range}(A)$ for all $i \in [d]$. Thus,

$$\begin{aligned}
\|SDAx\|_\infty &\leq \textstyle\sum_{i \in [d]} \|SDA_i\|_\infty \cdot |x_i| \\
&\leq \textstyle\sum_{i \in [d]} \left(\|DA_i\|_\infty + \max_{j \in [m]} z_j(A_i)\right) \cdot |x_i| \quad \text{(conditioned on } \mathcal{E}_1) \\
&\leq \textstyle\sum_{i \in [d]} (\|DA_i\|_\infty \cdot |x_i|) + \iota \cdot \|x\|_1 , \quad \text{(conditioned on } \mathcal{E}_2)
\end{aligned} \tag{1}$$

Let $v_i = \|DA_i\|_\infty$ and $v = \{v_1, \ldots, v_d\}$. By Hölder's inequality, we have

$$\textstyle\sum_{i \in [d]} (\|DA_i\|_\infty \cdot |x_i|) = \sum_{i \in [d]} (v_i \cdot |x_i|) \leq \|v\|_p \|x\|_q .$$

We next bound $\|v\|_p$:

$$\|v\|_p^p = \textstyle\sum_{i \in [d]} \|DA_i\|_\infty^p \sim \sum_{i \in [d]} \|A_i\|_p^p / u_i = \sum_{i \in [d]} 1/u_i,$$

where each u_i ($i \in [d]$) is an exponential. By Lemma 5 we know that with probability 0.99, $\sum_{i \in [d]} 1/u_i \leq 200/\kappa_1 \cdot d \log d$, thus $\|v\|_p \leq (200/\kappa_1 \cdot d \log d)^{1/p}$. Denote this event by $\mathcal{E}_3$ which we condition on. Thus,

$$\begin{aligned}
(1) &\leq \|v\|_p \|x\|_q + \iota \|x\|_1 \\
&\leq (200/\kappa_1 \cdot d \log d)^{1/p} \|x\|_q + \iota d^{1-1/q} \|x\|_q \quad \text{(conditioned on } \mathcal{E}_3) \\
&\leq 2(200/\kappa_1 \cdot d \log d)^{1/p} \|x\|_q \quad (\iota < 1/d^{1/p}) \\
&\leq 2(200/\kappa_1 \cdot d \log d)^{1/p} \cdot \|Ax\|_p . \quad (A \text{ is } (d^{1/p}, 1, p)\text{-well-conditioned})
\end{aligned} \tag{2}$$

3.2.2. No Underestimation

In this section we lower bound $\|SDAx\|_\infty$, or $\|SDy\|_\infty$, for all $y \in \text{range}(A)$. For a fixed $y \in \text{range}(A)$, by the triangle inequality

$$\|SDy\|_\infty \quad \geq \quad \|Dy\|_\infty - \max_{j\in[m]} z_j(y).$$

By Lemma 8 we have that with probability $(1 - m \cdot e^{-\eta})$, $z_j(y) \leq \iota \|y\|_p$ for all $j \in [m]$. We next bound $\|Dy\|_\infty$. By Property 1, it holds that $\|Dy\|_\infty \sim \|y\|_p / v^{1/p}$, where v is an exponential. Since $\mathbf{Pr}[v \geq \rho] \leq e^{-\rho}$ for an exponential v, with probability $(1 - e^{-\rho})$ we have

$$\|Dy\|_\infty \quad \geq \quad 1/\rho^{1/p} \cdot \|y\|_p , \quad \forall y \in \text{range}(A). \tag{3}$$

Therefore, with probability $(1 - m \cdot e^{-\eta} - e^{-\rho}) \geq (1 - 2e^{-\rho})$,

$$\|SDy\|_\infty \quad \geq \quad \|Dy\|_\infty - \iota \|y\|_p \geq 1/(2\rho^{1/p}) \cdot \|y\|_p . \tag{4}$$

Given the above "for each" result (for each y, the bound holds with probability $1 - 2e^{-\rho}$), we next use a standard net-argument to show

$$\|SDy\|_\infty \geq \Omega\left(1/\rho^{1/p} \cdot \|y\|_p\right), \quad \forall y \in \text{range}(A). \tag{5}$$

Due to space constraints, we leave the arguments to Appendix B.2.

Finally, Theorem 6 follows from inequalities (2), (5), and our choice of ρ.

4. p-norm with $1 \leq p \leq 2$

4.1. Algorithm

Our construction of the subspace embedding matrix Π is similar to that for p-norms with $p > 2$: We again set $\Pi = SD$, where D is an $n \times n$ diagonal matrix with $1/u_1^{1/p}, \ldots, 1/u_n^{1/p}$ on the diagonal, where u_i $(i = 1, \ldots, n)$ are i.i.d. exponentials. The difference is that this time we choose S to be an (m, s)-OSE with $(m, s) = \big(O(d^{1+\gamma}), O(1)\big)$ from [25] (γ is an arbitrary small constant). More precisely, we first pick random hash functions $h : [n] \times [s] \to [m/s], \sigma : [n] \times [s] \to \{-1, 1\}$. For each $(i, j) \in [n] \times [s]$, we set $S_{(j-1)s+h(i,j),i} = \sigma(i, j)/\sqrt{s}$, where $\sqrt{s}$ is just a normalization factor.

4.2. Analysis

In this section we prove the following theorem.

Theorem 9 *Let A be an Auerbach basis of a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_p)$. Given the above choices of $\Pi \in \mathbb{R}^{O(d^{1+\gamma})\times n}$. For any $1 \leq p < 2$ we have*

$$\Omega\left(\max\left\{1/(d\log d\log n)^{\frac{1}{p}-\frac{1}{2}}, 1/(d\log d)^{1/p}\right\}\right)\cdot\|Ax\|_p \leq \|\Pi Ax\|_2 \leq O((d\log d)^{1/p})\cdot\|Ax\|_p , \ \forall x \in \mathbb{R}^d.$$

Again, since the inequality holds for all $x \in \mathbb{R}^d$, the theorem holds if we replace the Auerbach basis A by any matrix M whose column space is a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_p)$.

Remark 10 *Using the inter-norm inequality $\|\Pi Ax\|_2 \leq \|\Pi Ax\|_p \leq d^{(1+\gamma)(1/p-1/2)} \|\Pi Ax\|_2 , \ \forall p \in [1, 2)$, we can replace the 2-norm estimator by the p-norm estimator in Theorem 9 by introducing another $d^{(1+\gamma)(1/p-1/2)}$ factor in the distortion. We will remove this extra factor for $p = 1$ below.*

In the rest of the section we prove Theorem 9. Define $\mathcal{E}_5$ to be the event that $\|SDAx\|_2 = (1 \pm 1/2) \|DAx\|_2$ which we condition on. Since S is an OSE, $\mathcal{E}_5$ holds with probability 0.99.

4.2.1. No Overestimation

We can write $S = \frac{1}{\sqrt{s}}(S_1, \ldots, S_s)^T$, where each $S_i \in \mathbb{R}^{(m/s)\times n}$ with one ± 1 on each column. For any $x \in \mathbb{R}^d$, let $y = Ax \in \mathbb{R}^n$. Let $D' \in \mathbb{R}^{n\times n}$ be a diagonal matrix with i.i.d. p-stable random variables on the diagonal. Let $\mathcal{E}_6$ be the event that for all $i \in [s]$, $\|S_i D' y\|_p \leq c_4 (d \log d)^{1/p} \cdot \|y\|_p$ for all $y \in \text{range}(A)$, where c_4 is some constant. Since $s = O(1)$ and $S_1, \ldots, S_s$ are independent, we know by [22] (Sec. A.2 in [22]) that $\mathcal{E}_6$ holds with probability 0.99. The rest of the proof conditions on $\mathcal{E}_6$. We have

$$\begin{aligned}
\|SDy\|_2 &\leq 3/2 \cdot \|Dy\|_2 \quad \text{(conditioned on } \mathcal{E}_5) \\
&\preceq 3/2 \cdot \kappa_p \left\|D'y\right\|_2 \quad \text{(Lemma 4)} \\
&\leq 3 \cdot \kappa_p \left\|SD'y\right\|_2 \quad \text{(conditioned on } \mathcal{E}_5) \\
&\leq 3 \cdot \kappa_p \left\|SD'y\right\|_p \\
&\leq 3 \cdot \kappa_p \cdot \tfrac{1}{\sqrt{s}} \textstyle\sum_{i\in[s]} \|S_i D'y\|_p \quad \text{(triangle inequality)} \\
&\leq 3 \cdot \kappa_p \cdot \tfrac{1}{\sqrt{s}} \cdot s \cdot c_4 (d \log d)^{1/p} \cdot \|y\|_p \quad \text{(conditioned on } \mathcal{E}_6) \\
&\leq c_5 (d \log d)^{1/p} \cdot \|y\|_p\,, \quad (s = O(1), \kappa_p = O(1), c_5 \text{ sufficiently large})
\end{aligned} \tag{6}$$

4.2.2. No Underestimation

For any $x \in \mathbb{R}^d$, let $y = Ax \in \mathbb{R}^n$.

$$\begin{aligned}
\|SDy\|_2 &\geq 1/2 \cdot \|Dy\|_2 \quad \text{(conditioned on } \mathcal{E}_5) \\
&\geq 1/2 \cdot \|Dy\|_\infty \sim 1/2 \cdot \|y\|_p / u \quad (u \text{ is exponential}) \\
&\geq 1/2 \cdot 1/\rho^{1/p} \cdot \|y\|_p\,. \quad \text{(By (3), holds w.pr. } (1 - e^{-\rho}))
\end{aligned} \tag{7}$$

Given this "for each" result, we again use a net-argument to show

$$\|SDy\|_2 \geq \Omega\left(1/\rho^{1/p} \cdot \|y\|_p\right) = \Omega\left(1/(d \log d)^{1/p}\right) \cdot \|y\|_p\,, \quad \forall y \in \text{range}(A). \tag{8}$$

Due to space constraints, we leave it to Appendix C.1.

In the case when $d \geq \log^{2/p-1} n$, using a finer analysis we can show that

$$\|SDy\|_2 \geq \Omega\left(1 \Big/ (d \log d \log n)^{\frac{1}{p}-\frac{1}{2}}\right) \cdot \|y\|_p\,, \quad \forall y \in \text{range}(A).$$

Due to the space constraints, we leave the improved analysis to Section C.2.

Finally, Theorem 9 follows from (6), (8) and our choices of ρ.

4.3. Improved Analysis for ℓ_1 Subspace Embeddings

We can further improve the distortion for ℓ_1 using the 1-norm estimator in Remark 10. Let $S' \in \mathbb{R}^{\tilde{O}(d)\times O(d^{1+\gamma})}$ be an $(\tilde{O}(d), \log^{O(1)} d)$-OSE from [25]. We have

$$\begin{aligned}
\left\|S'SDAx\right\|_1 &\leq \log^{O(1)}(d) \cdot \|SDAx\|_1 \leq \log^{O(1)}(d) \, \|DAx\|_1 \\
&\preceq \log^{O(1)}(d) \cdot \|CAx\|_1 \quad (C \in \mathbb{R}^{n\times n} \text{ be a diagonal matrix with i.i.d. Cauchy}) \\
&\leq \log^{O(1)}(d) \cdot \textstyle\sum_{i\in[n]} \|CAe_i\|_1 \cdot \|x\|_\infty \\
&\leq d \log^{O(1)} d \cdot \|x\|_\infty \quad \text{(Lemma 2.3 in [12])} \\
&\leq d \log^{O(1)} d \cdot \|Ax\|_1\,.
\end{aligned}$$

The first two inequalities follow from the fact that each column of S' and S only have $\log^{O(1)}(d)$ of ± 1's, and therefore the mappings S and S' contract ℓ_1-norms, up to a $\log^{O(1)}(d)$ factor.

The lower bounds in Section 4.2.2 still holds since $\|S'SDAx\|_1 \geq \|S'SDAx\|_2 \geq 1/2 \cdot \|SDAx\|_2$.

We state the following theorem in terms of a general matrix whose column space is a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_1)$. In Section C.3 we show that our analysis is tight up to a polylog factor.

Theorem 11 *Let M be a full-rank matrix in a d-dimensional subspace of $(\mathbb{R}^n, \|\cdot\|_1)$. Given the above choices of S, S' and D, let $\Pi = S'SD \in \mathbb{R}^{\tilde{O}(d)\times n}$. We have*

$$\Omega\left(\max\left\{1/(d\log d), 1/\sqrt{d\log d\log n}\right\}\right) \cdot \|Mx\|_1 \leq \|\Pi Mx\|_1 \leq O(d\log^{O(1)} d) \cdot \|Mx\|_1, \quad \forall x \in \mathbb{R}^d.$$

The embedding ΠM can be computed in time $O(nnz(M) + d^{2+\gamma}\log^{O(1)} d)$.

5. Regression

We need the following lemmas for ℓ_p regression.

Lemma 12 ([12]) *Given a matrix $M \in \mathbb{R}^{n\times d}$ with full column rank and $p \in [1, \infty)$, it takes at most $O(nd^3\log n)$ time to find a matrix $R \in \mathbb{R}^{d\times d}$ such that MR^{-1} is (α, β, p)-well-conditioned with $\alpha\beta \leq 2d^{1+\max\{1/2,1/p\}}$.*

Lemma 13 ([12]) *Given a matrix $M \in \mathbb{R}^{n\times d}, p \in [1, \infty), \epsilon > 0$, and a matrix $R \in \mathbb{R}^{d\times d}$ such that MR^{-1} is (α, β, p)-well-conditioned, it takes $O(nnz(M) \cdot \log n)$ time to compute a sampling matrix $\Pi \in \mathbb{R}^{t\times n}$ such that with probability 0.99, $(1-\epsilon)\|Mx\|_p \leq \|\Pi Mx\|_p \leq (1+\epsilon)\|Mx\|_p$, $\forall x \in \mathbb{R}^d$. The value t is $O\left((\alpha\beta)^p d\log(1/\epsilon)/\epsilon^2\right)$ for $1 \leq p < 2$ and $O\left((\alpha\beta)^p d^{p/2}\log(1/\epsilon)/\epsilon^2\right)$ for $p > 2$.*

Lemma 14 ([12]) *Given an ℓ_p-regression problem specified by $M \in \mathbb{R}^{n\times(d-1)}, b \in \mathbb{R}^n$, and $p \in [1, \infty)$, let Π be a $(1\pm\epsilon)$-distortion embedding matrix of the subspace spanned by M's columns and b from Lemma 13, and let $\hat{x}$ be an optimal solution to the sub-sampled problem $\min_{x\in\mathbb{R}^d}\|\Pi Mx - \Pi b\|_p$. Then $\hat{x}$ is a $\frac{1+\epsilon}{1-\epsilon}$-approximation solution to the original problem.*

5.1. Regression for p-norm with $p > 2$

Lemma 15 *Let $\Pi \in \mathbb{R}^{m\times n}$ be a subspace embedding matrix of the d-dimensional normed space spanned by the columns of matrix $M \in \mathbb{R}^{n\times d}$ such that $\mu_1\|Mx\|_p \leq \|\Pi Mx\|_\infty \leq \mu_2\|Mx\|_p$ for $\forall x \in \mathbb{R}^d$. If R is a matrix such that ΠMR^{-1} is (α, β, ∞)-well-conditioned, then MR^{-1} is $(\beta\mu_2, d^{1/p}\alpha/\mu_1, p)$-well-conditioned for any $p \in (2, \infty)$.*

Proof According to Definition 1, we only need to prove

$$\begin{aligned} \|x\|_q &\leq \|x\|_1 \leq \beta\left\|\Pi MR^{-1}x\right\|_\infty \quad (\Pi MR^{-1} \text{ is } (\alpha,\beta,\infty)\text{-well-conditioned}) \\ &\leq \beta\cdot\mu_2\left\|MR^{-1}x\right\|_p. \quad (\text{property of } \Pi) \end{aligned}$$

And,

$$\begin{aligned} \left\|MR^{-1}\right\|_p^p &= \sum_{i\in[d]}\left\|MR^{-1}e_i\right\|_p^p \quad (e_i \text{ is the standard basis in } \mathbb{R}^d) \\ &\leq 1/\mu_1^p\sum_{i\in[d]}\left\|\Pi MR^{-1}e_i\right\|_\infty^p \quad (\text{property of } \Pi) \\ &\leq 1/\mu_1^p\cdot d\alpha^p. \quad (\Pi MR^{-1} \text{ is } (\alpha,\beta,\infty)\text{-well-conditioned}) \end{aligned}$$

■

Theorem 16 *There exists an algorithm that given an ℓ_p-regression problem specified by $M \in \mathbb{R}^{n\times(d-1)}$, $b \in \mathbb{R}^n$ and $p \in (2,\infty)$, with constant probability computes a $(1+\epsilon)$-approximation to an ℓ_p-regression problem in time $\tilde{O}\left(nnz(\bar{M}) + n^{1-2/p}d^{4+2/p} + d^{8+4p} + \phi(\tilde{O}(d^{3+2p}/\epsilon^2), d)\right)$, where $\bar{M} = [M, -b]$ and $\phi(t,d)$ is the time to solve ℓ_p-regression problem on t vectors in d dimensions.*

Proof Our algorithm is similar to those ℓ_p-regression algorithms described in [14, 12, 22]. For completeness we sketch it here. Let Π be the subspace embedding matrix in Section 3 for $p > 2$. By Theorem 6, we have $(\mu_1, \mu_2) = \left(\Omega(1/(d\log d)^{1/p}), O((d\log d)^{1/p})\right)$.

Algorithm: ℓ_p **regression for** $p > 2$

1. Compute $\Pi\bar{M}$.
2. Use Lemma 12 to compute a matrix $R \in \mathbb{R}^{d\times d}$ such that $\Pi\bar{M}R^{-1}$ is (α,β,∞)-well-conditioned with $\alpha\beta \leq 2d^{3/2}$. By Lemma 15, $\bar{M}R^{-1}$ is $(\beta\mu_2, d^{1/p}\alpha/\mu_1, p)$-well-conditioned.
3. Given R, use Lemma 13 to find a sampling matrix Π^1 such that $(1-\epsilon)\cdot\left\|\bar{M}x\right\|_p \leq \left\|\Pi^1\bar{M}x\right\|_p \leq (1+\epsilon)\cdot\left\|\bar{M}x\right\|_p, \quad \forall x \in \mathbb{R}^d$.
4. Compute $\hat{x}$ which is the optimal solution to the sub-sampled problem $\min_{x\in\mathbb{R}^d}\left\|\Pi^1 Mx - \Pi^1 b\right\|_p$.

Analysis. The correctness of the algorithm is guaranteed by Lemma 14. Now we analyze the running time. Step 1 costs time $O(\text{nnz}(\bar{M}))$, by our choice of Π. Step 2 costs time $O(md^3\log m)$ by Lemma 12, where $m = O(n^{1-2/p}\log n(d\log d)^{1+2/p} + d^{5+4p})$. Step 3 costs time $O(\text{nnz}(\bar{M})\log n)$ by Lemma 13, giving a sampling matrix $\Pi^1 \in \mathbb{R}^{t\times n}$ with $t = O(d^{3+2p}\log^2 d\log(1/\epsilon)/\epsilon^2)$. Step 4 costs time $\phi(t,d)$, which is the time to solve ℓ_p-regression problem on t vectors in d dimensions. To sum up, the total running time is $O\left(\text{nnz}(\bar{M})\log n + n^{1-2/p}d^{4+2/p}\log^2 n\log^{1+2/p} d + d^{8+4p}\log n + \phi(O(d^{3+2p}\log^2 d\log(1/\epsilon)/\epsilon^2), d)\right)$. ■

5.2. Regression for p-norm with $1 \leq p < 2$

Theorem 17 *There exists an algorithm that given an ℓ_p regression problem specified by $M \in \mathbb{R}^{n\times(d-1)}$, $b \in \mathbb{R}^n$ and $p \in [1,2)$, with constant probability computes a $(1+\epsilon)$-approximation to an ℓ_p-regression problem in time $\tilde{O}\left(nnz(\bar{M}) + d^{7-p/2} + \phi(\tilde{O}(d^{2+p}/\epsilon^2), d)\right)$, where $\bar{M} = [M, -b]$ and $\phi(t,d)$ is the time to solve ℓ_p-regression problem on t vectors in d dimensions.*

Proof The regression algorithm for $1 \leq p < 2$ is similar but slightly more complicated than that for $p > 2$, since we try to optimize the dependence on d in the running time. Due to space constraints, we leave this proof to Appendix D.1. ■

Remark 18 *In [22] an algorithm together with several variants for ℓ_1-regression are proposed, all with running time of the form $\tilde{O}\left(nnz(\bar{M}) + poly(d) + \phi(\tilde{O}(poly(d)/\epsilon^2), d)\right)$. Among all these variants, the power of d in poly(d) (ignoring log factors) in the second term is at least 7, and the power of d in poly(d) in the third term is at least 3.5. In our algorithm both terms are improved.*

Application to ℓ_1 Subspace Approximation. Given a matrix $M \in \mathbb{R}^{n\times d}$ and a parameter k, the ℓ_1-subspace approximation is to compute a matrix $\hat{M}$ of rank $k \in [d-1]$ such that $\left\|M - \hat{M}\right\|_1$ is minimized. When $k = d-1$, $\hat{M}$ is a hyperplane, and the problem is called ℓ_1 *best hyperplane fitting*. In [12] it is shown that this problem is equivalent to solving the regression problem $\min_{W\in\mathcal{C}} \|AW\|_1$, where the constraint set is $\mathcal{C} = \{W \in \mathbb{R}^{d\times d} : W_{ii} = -1\}$. Therefore, our ℓ_1-regression result directly implies an improved algorithm for ℓ_1 best hyperplane fitting. Formally, we have

Theorem 19 *Given $M \in \mathbb{R}^{n\times d}$, there exists an algorithm that computes a $(1+\epsilon)$-approximation to the ℓ_1 best hyperplane fitting problem with probability 0.9, using time $O\left(nnz(M)\log n + \frac{1}{\epsilon^2}poly(d, \log\frac{d}{\epsilon})\right)$.*

The poly(d) factor in our algorithm is better than those by using the regression results in [11, 12, 22].

6. Regression in the Distributed Setting

Due to space constraints, we leave this section to Appendix E.

References

[1] Nir Ailon and Bernard Chazelle. The fast johnson–lindenstrauss transform and approximate nearest neighbors. *SIAM J. Comput.*, 39(1):302–322, 2009.

[2] Nir Ailon and Edo Liberty. Fast dimension reduction using rademacher series on dual bch codes. In *SODA*, pages 1–9, 2008.

[3] Alexandr Andoni. High frequency moment via max stability. Available at http://web.mit.edu/andoni/www/papers/fkStable.pdf, 2012.

[4] Alexandr Andoni, Robert Krauthgamer, and Krzysztof Onak. Streaming algorithms via precision sampling. In *FOCS*, pages 363–372, 2011.

[5] H. Auerbach. *On the area of convex curves with conjugate diameters*. PhD thesis, PhD thesis, University of Lwów, 1930.

[6] Lakshminath Bhuvanagiri, Sumit Ganguly, Deepanjan Kesh, and Chandan Saha. Simpler algorithm for estimating frequency moments of data streams. In *SODA*, pages 708–713, 2006.

[7] J. Bourgain, J. Lindenstrauss, and V. Milman. Approximation of zonoids by zonotopes. *Acta mathematica*, 162(1):73–141, 1989.

[8] Vladimir Braverman and Rafail Ostrovsky. Recursive sketching for frequency moments. *CoRR*, abs/1011.2571, 2010.

[9] Kenneth L. Clarkson. Subgradient and sampling algorithms for ℓ_1 regression. In *In Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 257–266, 2005.

[10] Kenneth L. Clarkson and David P. Woodruff. Numerical linear algebra in the streaming model. In *STOC*, pages 205–214, 2009.

[11] Kenneth L. Clarkson and David P. Woodruff. Low rank approximation and regression in input sparsity time. *CoRR*, abs/1207.6365, 2012. To appear in STOC, 2013.

[12] Kenneth L. Clarkson, Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, Xiangrui Meng, and David P. Woodruff. The fast cauchy transform: with applications to basis construction, regression, and subspace approximation in l1. *CoRR*, abs/1207.4684, 2012. Also in SODA 2013.

[13] Laëtitia Comminges and Arnak S. Dalalyan. Tight conditions for consistent variable selection in high dimensional nonparametric regression. *Journal of Machine Learning Research - Proceedings Track (COLT)*, 19:187–206, 2011.

[14] A. Dasgupta, P. Drineas, B. Harb, R. Kumar, and M.W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. *SIAM Journal on Computing*, 38(5):2060–2078, 2009.

[15] Sébastien Gerchinovitz. Sparsity regret bounds for individual sequences in online linear regression. *Journal of Machine Learning Research - Proceedings Track (COLT)*, 19:377–396, 2011.

[16] Daniel Hsu, Sham M. Kakade, and Tong Zhang. Random design analysis of ridge regression. *Journal of Machine Learning Research - Proceedings Track (COLT)*, 23:9.1–9.24, 2012.

[17] Piotr Indyk. Stable distributions, pseudorandom generators, embeddings, and data stream computation. *J. ACM*, 53(3):307–323, May 2006.

[18] Piotr Indyk and David P. Woodruff. Optimal approximations of the frequency moments of data streams. In *STOC*, pages 202–208, 2005.

[19] Daniel M. Kane and Jelani Nelson. Sparser johnson-lindenstrauss transforms. In *SODA*, pages 1195–1206, 2012.

[20] Daniel Kifer, Adam D. Smith, and Abhradeep Thakurta. Private convex optimization for empirical risk minimization with applications to high-dimensional regression. *Journal of Machine Learning Research - Proceedings Track (COLT)*, 23:25.1–25.40, 2012.

[21] Michael W. Mahoney. Randomized algorithms for matrices and data. *CoRR*, abs/1104.5557, 2011.

[22] Xiangrui Meng and Michael W. Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. *CoRR*, abs/1210.3135, 2012. To appear in STOC, 2013.

[23] Xiangrui Meng, Michael A. Saunders, and Michael W. Mahoney. Lsrn: A parallel iterative solver for strongly over- or under-determined systems. *CoRR*, abs/1109.5981, 2011.

[24] Gary Miller and Richard Peng. An iterative approach to row sampling. Unpublished manuscript, October 2012.

[25] Jelani Nelson and Huy L. Nguyen. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. *CoRR*, abs/1211.1002, 2012.

[26] J. P. Nolan. *Stable Distributions - Models for Heavy Tailed Data*. Birkhauser, Boston, 2013. In progress, Chapter 1 online at academic2.american.edu/$\sim$jpnolan.

[27] Eric Price and David P. Woodruff. Applications of the shannon-hartley theorem to data streams and sparse recovery. In *ISIT*, pages 2446–2450, 2012.

[28] Tamás Sarlós. Improved approximation algorithms for large matrices via random projections. In *FOCS*, pages 143–152, 2006.

[29] Christian Sohler and David P. Woodruff. Subspace embeddings for the l1-norm with applications. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, STOC '11, pages 755–764, 2011.

[30] Jiyan Yang, Xiangrui Meng, and Michael Mahoney. Quantile regression for large-scale applications. *CoRR*, abs/1305.0087, 2013.

Appendix A. Missing Proofs in Section 2

A.1. Proof for Lemma 4

Proof By Nolan ([26], Theorem 1.12), if X_p is p-stable with $p \in [1, 2)$, then

$$\mathbf{Pr}[X > x] \sim c_p x^{-p},$$

for some constant c_p when $x \to \infty$. By Property 1 we know that if U is exponential, then

$$\mathbf{Pr}[1/U^{1/p} > x] = \mathbf{Pr}[U < 1/x^p] \leq c_e x^{-p},$$

for some constant c_e. Therefore there exists a constant κ_p such that $|X_p| \succeq \kappa_p \cdot 1/U^{1/p}$. ■

A.2. Proof for Lemma 5

Proof By Lemma 4 we know $|C| \succeq \kappa_1 \cdot 1/u_i$ for a 1-stable (i.e., Cauchy) C and an exponential u_i. Given d Cauchy random variables $C_1, \ldots, C_d$, let $Y = \sum_{i \in [d]} |C_i|$. By Lemma 2.3 in [12] we have for any $t > 1$,

$$\mathbf{Pr}[Y > td] \leq (1 + o(1)) \log(td)/t.$$

The lemma follows from the fact that $Y \succeq \kappa_1 X$. ■

Appendix B. Missing Proofs in Section 3

B.1. Proof for Lemma 8

Proof (sketch, and we refer readers to [3] for the full proof). The first item simply follows from the birthday paradox; note that by our choice of m we have $\sqrt{m} = \omega(d/\tau^p)$. For the second item, we use Bernstein's inequality to show that for each $j \in [m]$, $z_j(y)$ is tightly concentrated around its mean, which is 0. ■

B.2. The Net-argument

Let the ball $B = \{y \in \mathbb{R}^n \mid y = Ax, \|x\|_1 = 1\}$. By Property 2 we have $\|y\|_p \leq 1$ for all $y \in B$. Call $B_\epsilon \subseteq B$ an ϵ-net of B if for any $y \in B$, we can find a $y' \in B_\epsilon$ such that $\|y - y'\|_p \leq \epsilon$. It is well-known that B has an ϵ-net of size at most $(3/\epsilon)^d$ [7]. We choose $\epsilon = 1/(8(200/\kappa_1 \cdot \rho d^2 \log d)^{1/p}$, then with probability

$$\begin{aligned} 1 - 2e^{-\rho} \cdot (3/\epsilon)^d &= 1 - 2e^{-c_1 d \log d} \cdot \left(24(200/\kappa_1 \cdot c_1 d \log d \cdot d^2 \log d)^{1/p}\right)^d \\ &\geq 0.99, \quad (c_1 \text{ sufficiently large}) \end{aligned}$$

$\|SDy'\|_\infty \geq 1/(2\rho^{1/p}) \cdot \|y'\|_p$ holds for all $y' \in B_\epsilon$. Let $\mathcal{E}_4$ denote this event which we condition on.

Now we consider $\{y \mid y \in B \backslash B_\epsilon\}$. Given any $y \in B\backslash B_\epsilon$, let $y' \in B_\epsilon$ such that $\|y - y'\|_p \leq \epsilon$. By the triangle inequality we have

$$\|SDy\|_\infty \geq \left\|SDy'\right\|_\infty - \left\|SD(y-y')\right\|_\infty. \tag{9}$$

Let x' be such that $Ax' = y'$. Let $\tilde{x} = x - x'$. Let $\tilde{y} = A\tilde{x} = y - y'$. Thus $\|\tilde{y}\|_p = \|A\tilde{x}\|_p \leq \epsilon$.

$$\begin{aligned}
\left\|SD(y-y')\right\|_\infty &= \|SDA\tilde{x}\|_\infty \\
&\leq 2(200/\kappa_1 \cdot d\log d)^{1/p} \cdot \|A\tilde{x}\|_p \quad \text{(by (2))} \\
&\leq 2(200/\kappa_1 \cdot d\log d)^{1/p} \cdot \epsilon. \\
&\leq 2(200/\kappa_1 \cdot d\log d)^{1/p} \cdot \epsilon \cdot d^{1/p} \cdot \|y\|_p \quad \text{(by Property 2)} \\
&= 1/(4\rho^{1/p}) \cdot \|y\|_p \quad (\epsilon = 1/(8(200/\kappa_1 \cdot \rho d^2 \log d)^{1/p}) \qquad (10)
\end{aligned}$$

By (4), (9) , (10), conditioned on $\mathcal{E}_4$, we have for all $y \in \text{range}(A)$, it holds that

$$\|SDy\|_\infty \geq 1/(2\rho^{1/p}) \cdot \|y\|_p - 1/(4\rho^{1/p}) \cdot \|y\|_p \geq 1/(4\rho^{1/p}) \cdot \|y\|_p.$$

Appendix C. Missing Proofs in Section 4

C.1. The Net-argument

Let the ball $B = \{y \in \mathbb{R}^n \mid y = Ax, \|y\|_p \leq 1\}$. Let $B_\epsilon \subseteq B$ be an ϵ-net of B with size at most $(3/\epsilon)^d$. We choose $\epsilon = 1/(4c_5(\rho d^2 \log d)^{1/p})$. Then with probability $1 - e^{-\rho} \cdot (3/\epsilon)^d \geq 0.99$, $\|SDy'\|_2 \geq 1/(2\rho^{1/p}) \cdot \|y'\|_p$ holds for all $y' \in B_\epsilon$. Let $\mathcal{E}_7$ denote this event which we condition on. For $y \in B\backslash B_\epsilon$, let $y' \in B_\epsilon$ such that $\|y - y'\|_p \leq \epsilon$. By the triangle inequality,

$$\|SDy\|_2 \geq \left\|SDy'\right\|_2 - \left\|SD(y-y')\right\|_2. \tag{11}$$

By (6) we have

$$\begin{aligned}
\left\|SD(y-y')\right\|_2 &\leq c_5(d\log d)^{1/p} \cdot \left\|y-y'\right\|_p \\
&\leq c_5(d\log d)^{1/p} \cdot \epsilon \\
&\leq c_5(d\log d)^{1/p} \cdot \epsilon \cdot d^{1/p} \|y\|_p \\
&= 1/(4\rho^{1/p}) \cdot \|y\|_p. \qquad (12)
\end{aligned}$$

By (7) (11) and (12), conditioned on $\mathcal{E}_7$, we have for all $y \in \text{range}(A)$, it holds that

$$\|SDy\|_2 \geq 1/(2\rho^{1/p}) \cdot \|y\|_p - 1/(4\rho^{1/p}) \cdot \|y\|_p \geq 1/(4\rho^{1/p}) \cdot \|y\|_p.$$

C.2. An Improved Analysis for ℓ_p ($p \in [1,2)$) Subspace Embeddings with $d \geq \log^{2/p-1} n$

The analysis for the upper bound is the same as that in Section 4.2.2. Now we give an improved analysis for the lower bound assuming that $d \geq \log^{2/p-1} n$.

Given a y, let y_X ($X \subseteq [n]$) be a vector such that $(y_X)_i = y_i$ if $i \in X$ and 0 if $i \in [n]\backslash X$. For convenience, we assume that the coordinates of y are sorted, that is, $y_1 \geq y_2 \geq \ldots \geq y_n$. Of course this order is unknown and not used by our algorithms.

We partition the n coordinates of y into $L = \log n + 2$ groups $W_1, \ldots, W_L$ such that $W_\ell = \{i \mid \|y\|_p / 2^\ell < y_i \le \|y\|_p / 2^{\ell-1}\}$. Let $w_\ell = |W_\ell|$ $(\ell \in [L])$ and let $W = \bigcup_{\ell \in [L]} W_\ell$. Thus

$$\|y_W\|_p^p \ge \|y\|_p^p - n \cdot \|y\|_p^p / (2^{L-1})^p \ge \|y\|_p^p / 2.$$

Let $K = c_K d \log d$ for a sufficiently large constant c_K. Define $T = \{1, \ldots, K\}$ and $B = W \backslash T$. Obviously, $W_1 \cup \ldots \cup W_{\log K - 1} \subseteq T$. Let $\lambda = 1/(10 d^p K)$ be a threshold parameter.

As before (Section 4.2.2), we have $\|SDy\|_2 \ge 1/2 \cdot \|Dy\|_2$. Now we analyze $\|Dy\|_2$ by two cases.

Case 1: $\|y_T\|_p^p \ge \|y\|_p^p / 4$. Let $H = \{i \mid (i \in [n]) \wedge (\ell_i^p \ge \lambda)\}$, where ℓ_i^p is the i-th leverage score of A. Since $\sum_{i \in [n]} \ell_i^p = d$, it holds that $|H| \le d/\lambda$.

We next claim that $\|y_{T \cap H}\|_p^p \ge \|y\|_p^p / 8$. To see this, recall that for each y_i $(i \in [n])$ we have $|y_i^p| \le d^{p-1} \ell_i^p$ (Property 2). Suppose that $\|y_{T \cap H}\|_p^p \le \|y\|_p^p / 8$, let $y_{i_{\max}}$ be the coordinate in $y_{T \backslash H}$ with maximum absolute value, then

$$\begin{aligned} \left|y_{i_{\max}}^p\right| &\ge \|y\|_p^p / (8K) \\ &\ge (1/d)/(8K) \quad \text{(by Property 2)} \\ &> d^{p-1} \lambda \\ &> d^{p-1} \ell_{i_{\max}}^p. \quad (i_{\max} \notin H) \end{aligned}$$

This is a contradiction.

Now we consider $\{u_i \mid i \in H\}$. Since the CDF of an exponential u is $(1 - e^{-x})$, we have with probability $(1 - d^{-10})$ that $1/u \ge 1/(10 \log d)$. By a union bound, with probability $(1 - d^{-10} |H|) \ge (1 - d^{-10} \cdot 10 d^{p+1} K) \ge 0.99$, it holds that $1/u_i \ge 1/(10 \log d)$ for all $i \in H$. Let $\mathcal{E}_7$ be this event which we condition on. Then for any y such that $\|y_T\|_p^p \ge \|y\|_p^p / 4$, we have $\sum_{i \in T \cap H} |y_i^p| / u_i \ge \|y\|_p^p / (80 \log d)$, and consequently,

$$\|Dy\|_2 \ge \frac{\|Dy\|_p}{K^{1/p - 1/2}} \ge \frac{\|y\|_p}{(80 \log d)^{1/p} \cdot K^{1/p - 1/2}}.$$

Case 2: $\|y_B\|_p^p \ge \|y\|_p^p / 4$. Let $W'_\ell = B \cap W_\ell$ $(\ell \in [L])$ and $w'_\ell = |W'_\ell|$. Let $F = \{\ell \mid w'_\ell \ge K/32\}$ and let $W' = \bigcup_{\ell \in F} W_\ell$. We have

$$\begin{aligned} \|y_{W'}\|_p^p &\ge \|y\|_p^p / 4 - \sum_{\ell = \log K}^{L} \left(K/32 \cdot (\|y\|_p / 2^{\ell - 1})^p \right) \\ &\ge \|y\|_p^p / 4 - \|y\|_p^p \cdot K/32 \cdot \sum_{\ell = \log K}^{L} \left(1/2^{\ell - 1} \right) \\ &\ge \|y\|_p^p / 8. \end{aligned}$$

For each $\ell \in F$, let $\alpha_\ell = w'_\ell / (2^\ell)^p$. We have

$$\|y\|_p^p / 8 \le \|y_{W'}\|_p^p = \sum_{\ell \in F} \left(w'_\ell \cdot \left(\|y\|_p / 2^{\ell - 1} \right)^p \right) \le \sum_{\ell \in F} \left(\alpha_\ell \cdot 4 \|y\|_p^p \right).$$

Thus $\sum_{\ell \in F} \alpha_\ell \ge 1/32$.

Now for each $\ell \in F$, we consider $\sum_{i \in W_\ell} \left(y_i / u_i^{1/p}\right)^2$. By Property 1, for an exponential u we have $\mathbf{Pr}[1/u \geq w'_\ell / K] \geq c'_e \cdot K / w'_\ell$ ($c'_e = \Theta(1)$). By a Chernoff bound, with probability $(1 - e^{-\Omega(K)})$, there are at least $\Omega(K)$ of $i \in W_\ell$ such that $1/u_i \geq w'_\ell / K$. Thus with probability at least $(1 - e^{-\Omega(K)})$, we have

$$\sum_{i \in W_\ell} \left(y_i / u_i^{1/p}\right)^2 \geq \Omega(K) \cdot \left(\frac{\|y\|_p}{2^\ell} \cdot \frac{{w'_\ell}^{1/p}}{K^{1/p}} \right)^2 \geq \Omega\left(\frac{\alpha_\ell^{2/p} \|y\|_p^2}{K^{2/p-1}} \right).$$

Therefore with probability $(1 - L \cdot e^{-\Omega(K)}) \geq (1 - e^{-\Omega(d \log d)})$, we have

$$\begin{aligned} \|Dy\|_2^2 &\geq \sum_{\ell \in F} \sum_{i \in W_\ell} \left(y_i / u_i^{1/p}\right)^2 \\ &\geq \Omega\left(\frac{\|y\|_p^2}{K^{2/p-1}} \cdot \sum_{\ell \in F} \alpha_\ell^{2/p} \right) \\ &\geq \Omega\left(\frac{\|y\|_p^2}{(K \log n)^{2/p-1}} \right) \quad \left(\textstyle\sum_{\ell \in F} \alpha_\ell \geq 1/32 \text{ and } |F| \leq \log n\right) \end{aligned} \tag{13}$$

Since the success probability is as high as $(1 - e^{-\Omega(d \log d)})$, we can further show that (13) holds for all $y \in \text{range}(A)$ using a net-argument as in previous sections.

To sum up the two cases, we have that for $\forall y \in \text{range}(A)$ and $p \in [1, 2)$, $\|Dy\|_2 \geq \Omega\left(\frac{\|y\|_p}{(d \log d \log n)^{\frac{1}{p} - \frac{1}{2}}} \right)$.

C.3. A Tight Example

We have the following example showing that given our embedding matrix $S'SD$, the distortion we get for $p = 1$ is tight up to a polylog factor. The worst case M is the same as the "bad" example given in [22], that is, $M = (I_d, \mathbf{0})^T$ where I_d is the $d \times d$ identity matrix. Suppose that the top d rows of M get perfectly hashed by S' and S, then $\|S'SDMx\|_2 = \left(\sum_{i \in [d]} (x_i / u_i)^2\right)^{1/2}$, where u_i are i.i.d. exponentials. Let $i^* = \arg\max_{i \in [d]} 1/u_i$. We know from Property 1 that with constant probability, $1/u_{i^*} = \Omega(d)$. Now if we choose x such that $x_{i^*} = 1$ and $x_i = 0$ for all $i \neq i^*$, then $\|S'SDMx\|_2 = d$. On the other hand, we know that with constant probability, for $\Omega(d)$ of $i \in [d]$ we have $1/u_i = \Theta(1)$. Let K ($|K| = \Omega(d)$) denote this set of indices. Now if we choose x such that $x_i = 1/|K|$ for all $i \in K$ and $x_i = 0$ for all $i \in [d] \backslash |K|$, then $\|S'SDMx\|_2 = 1/\sqrt{|K|} = O(1/\sqrt{d})$. Therefore the distortion is at least $\Omega(d^{3/2})$.

Appendix D. Missing Proofs in Section 5

Lemma 20 ([29, 22]) *Given $M \in \mathbb{R}^{n \times d}$ with full column rank, $p \in [1, 2)$, and $\Pi \in \mathbb{R}^{m \times n}$ whose entries are i.i.d. p-stables, if $m = cd \log d$ for a sufficiently large constant c, then with probability* 0.99*, we have*

$$\Omega(1) \cdot \|Mx\|_p \leq \|\Pi Mx\|_p \leq O((d \log d)^{1/p}) \cdot \|Mx\|_p, \quad \forall x \in \mathbb{R}^d.$$

In addition, ΠM can be computed in time $O(nd^{\omega-1})$ where ω is the exponent of matrix multiplication.

Lemma 21 *Let $\Pi \in \mathbb{R}^{m\times n}$ be a subspace embedding matrix of the d-dimensional normed space spanned by the columns of matrix $M \in \mathbb{R}^{n\times d}$ such that*

$$\mu_1 \cdot \|Mx\|_p \leq \|\Pi M x\|_2 \leq \mu_2 \cdot \|Mx\|_p\,, \quad \forall x \in \mathbb{R}^d. \tag{14}$$

If R is the "R" matrix in the QR-decomposition of ΠM, then MR^{-1} is (α, β, p)-well-conditioned with $\alpha\beta \leq d^{1/p}\mu_2/\mu_1$ for any $p \in [1, 2)$.

Proof We first analyze $\Delta_p(MR^{-1}) = \mu_2/\mu_1$ (Definition 2).

$$\begin{aligned}\left\|MR^{-1}x\right\|_p &\leq 1/\mu_1 \cdot \left\|\Pi M R^{-1} x\right\|_2 \quad \text{(by (14))}\\ &= 1/\mu_1 \cdot \|Qx\|_2 \quad (\Pi M R^{-1} = QRR^{-1} = Q)\\ &= 1/\mu_1 \cdot \|x\|_2 \quad (Q \text{ has orthonormal columns})\end{aligned}$$

And

$$\begin{aligned}\left\|MR^{-1}x\right\|_p &\geq 1/\mu_2 \cdot \left\|\Pi M R^{-1} x\right\|_2 \quad \text{(by (14))}\\ &= 1/\mu_2 \cdot \|Qx\|_2\\ &= 1/\mu_2 \cdot \|x\|_2\end{aligned}$$

Then by Lemma 3 it holds that

$$\alpha\beta = \Delta'_p(MR^{-1}) \leq d^{\max\{1/2, 1/p\}} \Delta_p(MR^{-1}) = d^{1/p}\mu_2/\mu_1.$$

■

D.1. Proof for Theorem 17

Proof The regression algorithm for $1 \leq p < 2$ is similar but slightly more complicated than that for $p > 2$, since we are trying to optimize the dependence on d in the running time. Let Π be the subspace embedding matrix in Section 4 for $1 \leq p < 2$. By theorem 9, we have $(\mu_1, \mu_2) = (\Omega(1/(d \log d)^{1/p}), O((d \log d)^{1/p}))$ (we can also use $(\Omega(1/(d \log d \log n)^{\frac{1}{p}-\frac{1}{2}}), O((d \log d)^{1/p}))$ which will give the same result).

Algorithm: ℓ_p-Regression for $1 \leq p < 2$

1. Compute $\Pi\bar{M}$.
2. Compute the QR-decomposition of $\Pi\bar{M}$. Let $R \in \mathbb{R}^{d\times d}$ be the "R" in the QR-decomposition.
3. Given R, use Lemma 13 to find a sampling matrix $\Pi^1 \in \mathbb{R}^{t_1\times n}$ such that
$$(1 - 1/2) \cdot \left\|\bar{M}x\right\|_p \leq \left\|\Pi^1 \bar{M} x\right\|_p \leq (1 + 1/2) \cdot \left\|\bar{M}x\right\|_p\,, \quad \forall x \in \mathbb{R}^d. \tag{15}$$
4. Use Lemma 20 to compute a matrix $\Pi^2 \in \mathbb{R}^{t_2\times t_1}$ for $\Pi^1\bar{M}$ such that
$$\Omega(1) \cdot \left\|\Pi^1 \bar{M} x\right\|_p \leq \left\|\Pi^2\Pi^1 \bar{M} x\right\|_p \leq O((d \log d)^{1/p}) \cdot \left\|\Pi^1 \bar{M}x\right\|_p\,, \quad \forall x \in \mathbb{R}^d.$$
Let $\Pi^3 = \Pi^2\Pi^1 \in \mathbb{R}^{t_2\times n}$. By (15) and $\|z\|_2 \leq \|z\|_p \leq m^{1/p-1/2}\,\|z\|_2$ for any $z \in \mathbb{R}^m$, we have
$$\Omega(1/{t_2}^{1/p-1/2}) \cdot \left\|\bar{M}x\right\|_p \leq \left\|\Pi^3 \bar{M} x\right\|_2 \leq O((d \log d)^{1/p}) \cdot \left\|\bar{M}x\right\|_p\,, \quad \forall x \in \mathbb{R}^d.$$

5. Compute the QR-decomposition of $\Pi^3\bar{M}$. Let $R_1 \in \mathbb{R}^{d\times d}$ be the "R" in the QR-decomposition.
6. Given R_1, use Lemma 13 again to find a sampling matrix $\Pi^4 \in \mathbb{R}^{t_3\times n}$ such that Π^4 is a $(1 \pm 1/2)$-distortion embedding matrix of the subspace spanned by $\bar{M}$.
7. Use Lemma 12 to compute a matrix $R_2 \in \mathbb{R}^{d\times d}$ such that $\Pi^4\bar{M}{R_2}^{-1}$ is (α, β, p)-well-conditioned with $\alpha\beta \le 2d^{1+1/p}$.
8. Given R_2, use Lemma 13 again to find a sampling matrix $\Pi^5 \in \mathbb{R}^{t_4\times n}$ such that Π^5 is a $(1 \pm \epsilon)$-distortion embedding matrix of the subspace spanned by $\bar{M}$.
9. Compute $\hat{x}$ which is the optimal solution to the sub-sampled problem $\min_{x\in\mathbb{R}^d} \left\|\Pi^5 Mx - \Pi^5 b\right\|_p$.

Analysis. The correctness of the algorithm is guaranteed by Lemma 14. Now we analyze the running time. Step 1 costs time $O(\text{nnz}(\bar{M}))$, by our choice of Π. Step 2 costs time $O(md^2) = O(d^{3+\gamma})$ using standard QR-decomposition, where γ is an arbitrarily small constant. Step 3 costs time $O(\text{nnz}(\bar{M}) \log n)$ by Lemma 13, giving a sampling matrix $\Pi^1 \in \mathbb{R}^{t_1\times n}$ with $t_1 = O(d^4 \log^2 d)$. Step 4 costs time $O(t_1 d^{\omega-1}) = O(d^{3+\omega} \log^2 d)$ where ω is the exponent of matrix multiplication, giving a matrix $\Pi^3 \in \mathbb{R}^{t_2\times n}$ with $t_2 = O(d \log d)$. Step 5 costs time $O(t_2 d^2) = O(d^3 \log d)$. Step 6 costs time $O(\text{nnz}(\bar{M}) \log n)$ by Lemma 13, giving a sampling matrix $\Pi^4 \in \mathbb{R}^{t_3\times n}$ with $t_3 = O(d^{4-p/2} \log^{2-p/2} d)$. Step 7 costs time $O(t_3 d^3 \log t_3) = O(d^{7-p/2} \log^{3-p/2} d)$. Step 8 costs time $O(\text{nnz}(\bar{M}) \log n)$ by Lemma 13, giving a sampling matrix $\Pi^5 \in \mathbb{R}^{t_4\times n}$ with $t_4 = O(d^{2+p} \log(1/\epsilon)/\epsilon^2)$. Step 9 costs time $\phi(t_4, d)$, which is the time to solve ℓ_p-regression problem on t_4 vectors in d dimensions. To sum up, the total running time is

$$O\left(\text{nnz}(\bar{M}) \log n + d^{7-p/2} \log^{3-p/2} d + \phi(O(d^{2+p} \log(1/\epsilon)/\epsilon^2), d)\right).$$

■

Appendix E. Regression in the Distributed Setting

In this section we consider the ℓ_p-regression problem in the distributed setting, where we have k machines $P_1, \ldots, P_k$ and one central server. Each machine has a disjoint subset of the rows of $M \in \mathbb{R}^{n\times(d-1)}$ and $b \in \mathbb{R}^d$. The server has a 2-way communication channel with each machine, and the server wants to communicate with the k machines to solve the ℓ_p-regression problem specified by M, b and p. Our goal is to minimize the overall communication of the system, as well as the total running time.

Let $\bar{M} = [M, -b]$. Let $I_1, \ldots, I_k$ be the sets of rows that $P_1, \ldots, P_k$ have, respectively. Let $\bar{M}_i$ $(i \in [k])$ be the matrix by setting all rows $j \in [n]\backslash I_i$ in $\bar{M}$ to 0. We use Π to denote the subspace embedding matrix proposed in Section 3 for $p > 2$ and Section 4 for $1 \le p < 2$, respectively. We assume that both the server and the k machines agree on such a Π at the beginning of the distributed algorithms using, for example, shared randomness.

E.1. Distributed ℓ_p-regression for $p > 2$

The distributed algorithm for ℓ_p regression with $p > 2$ is just a distributed implementation of Algorithm 5.1.

Algorithm: Distributed ℓ_p-regression for $p > 2$

1. Each machine computes and sends $\|\bar{M}_i\|_p$ to the server. And then the server computes $\|\bar{M}\|_p = \left(\sum_{i\in[k]} \|\bar{M}_i\|_p^p\right)^{1/p}$ and sends to each site. $\|\bar{M}\|_p$ is needed for Lemma 13 which we will use later.

2. Each machine P_i computes and sends $\Pi\bar{M}_i$ to the server.

3. The server computes $\Pi\bar{M}$ by summing up $\Pi\bar{M}_i$ $(i = 1, \ldots, k)$. Next, the server uses Lemma 12 to compute a matrix $R \in \mathbb{R}^{d\times d}$ such that $\Pi\bar{M}R^{-1}$ is (α, β, ∞)-well-conditioned with $\alpha\beta \le 2d^{3/2}$, and sends R to each of the k machines.

4. Given R and $\|\bar{M}\|_p$, each machine uses Lemma 13 to compute a sampling matrix Π_i^1 such that Π_i^1 is a $(1 \pm \epsilon)$-distortion embedding matrix of the subspace spanned by $\bar{M}_i$, and then sends the sampled rows of $\Pi_i^1\bar{M}_i$ that are in I_i to the server.

5. The server constructs a global matrix $\Pi^1\bar{M}$ such that the j-th row of $\Pi^1\bar{M}$ is just the j-th row of $\Pi_i^1\bar{M}_i$ if $(j \in I_i) \wedge (j \text{ get sampled})$, and 0 otherwise. Next, the server computes $\hat{x}$ which is the optimal solution to the sub-sampled problem $\min_{x\in\mathbb{R}^d} \|\Pi^1 Mx - \Pi^1 b\|_p$.

Analysis. Step 1 costs communication $O(k)$. Step 2 costs communication $O(kmd)$ where $m = O(n^{1-2/p}\log n(d\log d)^{1+2/p} + d^{5+4p})$. Step 3 costs communication $O(kd^2)$. Step 4 costs communication $O(td + k)$ where $t = O(d^{3+2p}\log^2 d\log(1/\epsilon)/\epsilon^2)$, that is, the total number of rows get sampled in rows $I_1 \cup I_2 \cup \cdots \cup I_k$. Therefore the total communication cost is

$$O\left(kn^{1-2/p}d^{2+2/p}\log n\log^{1+2/p} d + kd^{6+4p} + d^{4+2p}\log^2 d\log(1/\epsilon)/\epsilon^2\right).$$

The total running time of the system, which is essentially the running time of the centralized algorithm (Theorem 16) plus the communication cost, is

$$O\left(\text{nnz}(\bar{M})\log n + (k + d^2\log n)(n^{1-2/p}d^{2+2/p}\log n\log^{1+2/p} d + d^{6+4p}) + \phi(O(d^{3+2p}\log^2 d\log(1/\epsilon)/\epsilon^2), d)\right).$$

E.2. Distributed ℓ_p-regression for $1 \le p < 2$

The distributed algorithm for ℓ_p-regression with $1 \le p < 2$ is a distributed implementation of Algorithm D.1.

Algorithm: Distributed ℓ_p-regression for $1 \le p < 2$

1. Each machine computes and sends $\|\bar{M}_i\|_p$ to the server. And then the server computes $\|\bar{M}\|_p = \left(\sum_{i\in[k]} \|\bar{M}_i\|_p^p\right)^{1/p}$ and sends to each site.

2. Each machine P_i computes and sends $\Pi\bar{M}_i$ to the server.

3. The server computes $\Pi\bar{M}$ by summing up $\Pi\bar{M}_i$ $(i = 1, \ldots, k)$. Next, the server computes a QR-decomposition of $\Pi\bar{M}$, and sends R (the "R" in QR-decomposition) to each of the k machines.

4. Given R and $\|\bar{M}\|_p$, each machine P_i uses Lemma 13 to compute a sampling matrix $\Pi_i^1 \in \mathbb{R}^{t_1\times n}$ such that Π_i^1 is a $(1 \pm 1/2)$-distortion embedding matrix of the subspace spanned by $\bar{M}_i$, and then sends the sampled rows of $\Pi_i^1\bar{M}_i$ that are in I_i to the server.

5. The server constructs a global matrix $\Pi^1 \bar{M}$ such that the j-th row of $\Pi^1 \bar{M}$ is just the j-th row of $\Pi_i^1 \bar{M}_i$ if $(j \in I_i) \wedge (j$ get sampled), and 0 otherwise. After that, the server uses Lemma 20 to compute a matrix $\Pi^2 \in \mathbb{R}^{t_2 \times t_1}$ for $\Pi^1 \bar{M}$. Next, the server computes a QR-decomposition of $\Pi^2 \Pi^1 \bar{M}$, and sends R_1 (the "R" in QR-decomposition) to each of the k machines.

6. Given R_1 and $\|\bar{M}\|_p$, each machine P_i uses Lemma 13 again to compute a sampling matrix $\Pi_i^4 \in \mathbb{R}^{t_3 \times n}$ such that Π_i^4 is a $(1 \pm 1/2)$-distortion embedding matrix of the subspace spanned by $\bar{M}_i$, and then sends the sampled rows of $\Pi_i^4 \bar{M}_i$ that are in I_i to the server.

7. The server constructs a global matrix $\Pi^4 \bar{M}$ such that the j-th row of $\Pi^4 \bar{M}$ is just the j-th row of $\Pi_i^4 \bar{M}_i$ if $(j \in I_i) \wedge (j$ get sampled), and 0 otherwise. Next, the server uses Lemma 12 to compute a matrix $R_2 \in \mathbb{R}^{d \times d}$ such that $\Pi \bar{M} R_2{}^{-1}$ is (α, β, p)-well-conditioned with $\alpha\beta \leq 2d^{1+1/p}$, and sends R_2 to each of the k machines.

8. Given R_2 and $\|\bar{M}\|_p$, each machine P_i uses Lemma 13 again to compute a sampling matrix $\Pi_i^5 \in \mathbb{R}^{t_4 \times n}$ such that Π_i^5 is a $(1 \pm \epsilon)$-distortion embedding matrix of the subspace spanned by $\bar{M}_i$, and then sends the sampled rows of $\Pi_i^5 \bar{M}_i$ that are in I_i to the server.

9. The server constructs a global matrix $\Pi^5 \bar{M}$ such that the j-th row of $\Pi^5 \bar{M}$ is just the j-th row of $\Pi_i^5 \bar{M}_i$ if $(j \in I_i) \wedge (j$ get sampled), and 0 otherwise. Next, the server computes $\hat{x}$ which is the optimal solution to the sub-sampled problem $\min_{x \in \mathbb{R}^d} \|\Pi^5 M x - \Pi^5 b\|_p$.

Communication and running time. Step 1 costs communication $O(k)$. Step 2 costs communication $O(kmd)$ where $m = O(d^{1+\gamma})$ for some arbitrarily small γ. Step 3 costs communication $O(kd^2)$. Step 4 costs communication $O(t_1 d + k)$ where $t_1 = O(d^4 \log^2 d)$. Step 5 costs communication $O(kd^2)$. Step 6 costs communication $O(t_3 d + k)$ where $t_3 = O(d \log d)$. Step 7 costs communication $O(kd^2)$. Step 8 costs communication $O(t_4 d + k)$ where $t_4 = O(d^{2+p} \log(1/\epsilon)/\epsilon^2)$. Therefore the total communication cost is

$$O\left(kd^{2+\gamma} + d^5 \log^2 d + d^{3+p} \log(1/\epsilon)/\epsilon^2\right).$$

The total running time of the system, which is essentially the running time of the centralized algorithm (Theorem 17) plus the communication cost, is

$$O\left(\text{nnz}(\bar{M}) \log n + kd^{2+\gamma} + d^{7-p/2} \log^{3-p/2} d + \phi(O(d^{2+p} \log(1/\epsilon)/\epsilon^2), d)\right).$$

Remark 22 *It is interesting to note that the work done by the server C is just poly(d), while the majority of the work at Step $2, 4, 6, 8$, which costs $O(nnz(\bar{M}) \cdot \log n)$ time, is done by the k machines. This feature makes the algorithm fully scalable.*