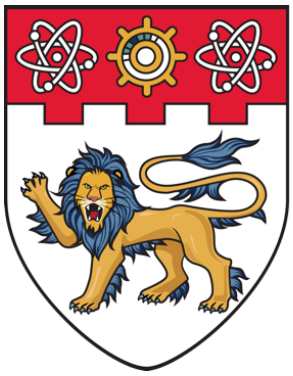


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

ALGORITHMS FOR LARGE-SCALE NUMERICAL LINEAR ALGEBRA

SUN YIMING

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

2024

ALGORITHMS FOR LARGE-SCALE NUMERICAL LINEAR ALGEBRA

SUN YIMING

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

A thesis submitted to the Nanyang Technological University in partial fulfilment of the requirement for the degree of Doctor of Philosophy

2024

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research done by me except where otherwise stated in this thesis. The thesis work has not been submitted for a degree or professional qualification to any other university or institution. I declare that this thesis is written by myself and is free of plagiarism and of sufficient grammatical clarity to be examined. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

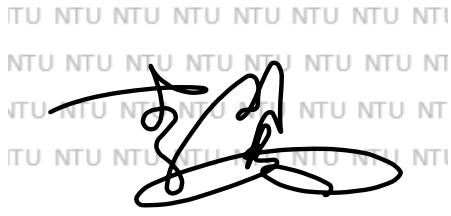
15/09/2024

Date

Sun Yiming

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it of sufficient grammatical clarity to be examined. To the best of my knowledge, the thesis is free of plagiarism and the research and writing are those of the candidate's except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.



15 September 2024

. . . . . .

Date

. . . . . .

Li Yi

Authorship Attribution Statement

This thesis contains material from 3 papers accepted at conferences in which I am listed as an author.

Chapter 2 is published as Yifei Jiang, Yi Li, Yiming Sun, Jiaxin Wang, and David P. Woodruff. Single Pass Entrywise-Transformed Low Rank Approximation. Proceedings of the 38th International Conference on Machine Learning, pp. 4982–4991. PMLR, 2021.

The contributions of the co-authors are as follows:

- Yi Li and David P. Woodruff proposed a COUNT-SKETCH-like data structure and completed the main part of the paper.
- I applied the main theorem to linear regression and gave corresponding space complexity for the regression problem.
- Yi Li conducted the experiment.

Chapter 3 is published as Cheng Chen, Yi Li, and Yiming Sun. Online Active Regression. Proceedings of the 39th International Conference on Machine Learning, pp. 3320–3335. PMLR, 2022.

The contributions of the co-authors are as follows:

- Yi Li suggested the topic.
- I wrote the regression proof part and algorithms of the manuscript. The manuscript was revised collaboratively with Yi Li and Cheng Chen.
- Yi Li proved that the compression technique can be used to estimate ℓ_p-Lewis weights where $p \in [1, 2]$ in online learning.
- I conducted all experiments using synthetic and real-world data.
- Cheng Chen wrote the proof of the upper bound on the sum of online Lewis weights
- All authors contributed equally.

Chapter 4 is published as Sheng-Jun Huang, Yi Li, Yiming Sun, and Ying-Peng Tang. One-shot Active Learning Based on Lewis Weight Sampling for Multiple Deep Models. Proceedings of the 12th International Conference on Learning Representations, 2024.

The contributions of the co-authors are as follows:

- Ying-Peng Tang initiated the topic and Yi Li proposed the theoretical direction.
- Ying-Peng Tang conducted all the experiments.
- I wrote the theory part and Ying-Peng Tang wrote the remainder of the paper.
- Yi Li revised the whole paper.
- All authors contributed equally.

15/09/2024
Date

Sun Yiming

Abstract

Numerical linear algebra is fundamental to many computational models used in various disciplines, from science to engineering. In the era of big data, the scale of numerical linear algebraic problems often becomes prohibitively large, rendering classical algorithms inefficient. To address this challenge, two common approaches are sampling and sketching, which extract and retain succinct information of the data, enabling efficient algorithms to produce approximate solutions that are acceptable for the original problem. A typical model for handling massive data is the turnstile streaming model, where data entries are updated incrementally, one at a time. Linear sketching, a widely used technique used in turnstile streaming algorithms and encompassing matrix row sampling as a special case, is the focus of this thesis.

This thesis studies three problems in randomized numerical linear algebra: (i) Entrywise-Transformed Low-Rank Approximation. A matrix $A = (a_{i,j})$ is given in the turnstile streaming model while the task is to find a low-rank approximation to the entrywise-transformed matrix $f(A) = (f(a_{i,j}))$. The nonlinearity of f poses challenges, rendering previous methods ineffective. We design a Count-Sketch-like structure to solve the problem. (ii) Online Active Regression. The problem asks to solve the regression task $\min_x \|Ax-b\|_p$, where A is given row-by-row in an online fashion and access to the entries of b should be minimized. We show that this problem can be solved using ℓ_p Lewis weight sampling and a compression technique for $p \in [1, 2]$. (iii) Active Regression with Shared Labels. The objective is to simultaneously solve k regression problems $\min_{x\in\mathbb{R}^d} \|f(A^j x) - b\|_p$ for $j = 1, 2, \ldots, k$ while minimizing the number of queries made to the entries of b, where f is a Lipschitz function and $f(x) = (f(x_i))$ is the entrywise transformation of x. We solve the problem using a shared sampling matrix S for all matrices A^j, based on Lewis weight sampling.

Acknowledgements

First of all, I would like to express my greatest gratitude to my supervisor, Prof. Li Yi, for his invaluable and consistent guidance and support throughout my Ph.D. journey. He guided me into the field of randomized numerical linear algebra. His mentorship and expertise have been essential to my research life. I am very grateful to have him as my supervisor.

I am also grateful to my parents, partner and friends for their priceless support and unconditional love, without which I would not be who I am today.

Contents

List of Figures

List of Algorithms

Chapter 1

Introduction

Numerical linear algebra forms the backbone of many computational techniques used across various scientific and engineering disciplines. From natural language processing to deep learning, matrices arising in these scenarios can be extremely large, making the classical methods infeasible. This prompts the need for innovative techniques that reduce the algorithmic complexity in both space and time. To address this challenge, sampling and sketching have emerged as two prevalent approaches, which extract essential information from large inputs, reducing the problem's size while allowing for the computation of acceptable approximation solutions. Typically, these methods give randomized algorithms, using randomness as part of their internal logic without assuming a probability distribution over possible inputs. Indeed, randomness and approximation are often necessary for achieving efficiency, as many problems otherwise would require, provably, storing essentially the entire input, which could be prohibitive for large matrices. In the field of numerical linear algebra, this use of randomization has given rise to a research direction known as *randomized numerical linear algebra*, which has now become a well-established field in theoretical computer science.

In the following sections, we shall review the fundamental concepts related to sketching and sampling algorithms. We shall also introduce definitions and notations used throughout this thesis, followed by a discussion of the foundational techniques such as subspace embeddings and Lewis weight sampling. We then present the randomized numerical linear algebraic problems considered in this thesis. Finally, we provide an overview of the content of each chapter.

1.1 Sketching and Sampling

Turnstile Streams. The *turnstile streaming model* is one of the earliest streaming models studied in theoretical computer science and has been a prevailing model in the research of streaming algorithms. In this model, the input corresponds to an underlying vector $x \in \mathbb{R}^n$ and the data stream consists of updates of the form $(i, \Delta) \in \{1, \dots, n\} \times \mathbb{R}$, where each update modifies the coordinate x_i as $x_i \leftarrow x_i + \Delta$. The goal is to avoid storing the entire vector while still being able to compute with an acceptable error a function $f(x)$ for the final vector x. The space usage should be $o(n)$ and ideally only $\text{poly}(\log n)$.

In the case where the underlying vector has only integer entries, i.e. $x \in \mathbb{Z}^n$, the increments Δ will also be integers. In the case where the input corresponds to a matrix $A \in \mathbb{R}^{m \times n}$, which can be viewed as an (mn)-dimensional vector, the stream entries perform matrix updates of the form $A_{ij} \leftarrow A_{ij} + \Delta$.

Sketching. A powerful approach to handle massive inputs that correspond to a vector $x \in \mathbb{R}^n$ is to use a *linear sketch*. Instead of storing the entire vector $x \in \mathbb{R}^n$, one stores a much shorter vector Sx, where $S \in \mathbb{R}^{m \times n}$ with $m \ll n$ is called the sketching matrix. This approach is particularly useful for turnstile streaming problems. When x receives an update $x_i \leftarrow x_i + \Delta$, the corresponding sketch Sx is also updated to $S(x + \Delta e_i) = Sx + \Delta(Se_i)$, where $e_1, \ldots, e_n$ are the canonical basis vectors in $\mathbb{R}^n$. This means that the sketch is updated by $\Delta(Se_i)$, which is the i-th column of S scaled by Δ. Therefore, a linear sketch is easy to maintain throughout the streaming process. The linearity of the sketch is also advantageous in distributed computing, where the input data stream is divided across multiple servers. If the servers use the same sketching matrix S, each server can independently sketch its portion of the data stream and the final sketch for the entire problem will simply be the sum of the sketches maintained by each server.

When the input stream corresponds to a matrix $A \in \mathbb{R}^{m \times n}$, one can view A as a vector of length mn and, consequently, a general sketching matrix S would have dimensions $s \times (mn)$. More commonly, a sketching algorithm for matrix inputs maintains a matrix product of the form SA or SAT, where $S \in \mathbb{R}^{s \times m}$ and $T \in \mathbb{R}^{m \times t}$ are also called sketching matrices. These types of sketches are usually referred to as one-sided and two-sided (or bilinear) sketches, respectively.

As said earlier, algorithms for massive inputs are usually randomized, which means that the sketching matrix S is random. If the distribution of S does not depend on the input, we say S is an oblivious sketch; otherwise we say S is non-oblivious. Typical examples of oblivious sketches include Count-Sketch [11] and Count-Min [19] for heavy hitter problems, Johnson-Lindenstrauss Transform matrices [42, 44] for dimension reduction of vectors and OSNAP [62] for subspace embeddings in the ℓ_2-norm. A typical example of non-oblivious sketches is the subspace embedding matrix in the ℓ_p-norm for $p \neq 2$, see Section 1.2.3 for more discussion.

Sampling. Sampling is a conventional technique for processing large datasets and is widely used in statistical tasks. The idea is to select representative data items and use these representatives to estimate the statistics of the original dataset. For matrix inputs, one can sample either the entries or the rows of the matrix. For example, entry sampling can be used to solve the matrix sparsification problem and row sampling the spectral sparsification (see, e.g. [75] for details).

This thesis focuses on row sampling of matrices, a specific case of linear sketches. We shall consider two sampling schemes, referred to as the first and second kind throughout this thesis. The first kind independently retains or discards each row, while the second kind ensures a fixed number of sampled rows, with each sample drawn from the same distribution. Formally, suppose that a matrix A has n rows.

- First kind of row sampling: Let $p_1, \ldots, p_n \in [0, 1]$ be the sampling probabilities. The i-th row of A is retained with probability p_i and discarded with probability $1 - p_i$. In other words, let S be an $n \times n$ diagonal matrix, where the diagonal entries S_{ii} are independent Bernoulli variables with $\mathbb{E}\, S_{ii} = p_i$. This S is called the sampling matrix and the sampled matrix is $S \cdot A$ with zero rows removed.

- Second kind of row sampling: Let $p_1, \ldots, p_n \geq 0$ with $\sum_i p_i = 1$, and let m be the prescribed sample size. We independently sample m rows of A, where each sample is the i-th row of A with probability p_i. In other words, the sampling matrix S is

an $m \times n$ matrix of i.i.d. rows and $\Pr\{S_{j,*} = e_i^\top\} = p_i$, where $S_{j,*}$ denotes the j-th row of S.

1.2 Preliminaries

Notation. We use $[n]$ to denote the integer set $\{1, 2, \ldots, n\}$ and $|F|$ the cardinality of a set F.

We use $\mathrm{Ber}(p)$ to denote the Bernoulli distribution with parameter p, i.e. for $X \sim \mathrm{Ber}(p)$, $\Pr\{X = 1\} = p$ and $\Pr\{X = 0\} = 1 - p$.

For a vector $x \in \mathbb{R}^n$, the weighted ℓ_p-norm of x is defined to be $\|x\|_{w,p} = (\sum_{i=1}^n w_i |x_i|^p)^{\frac{1}{p}}$, where $w_1, w_2, \ldots, w_n \geq 0$ are the weights such that $w_1 + \cdots + w_n = 1$. The ℓ_p-norm of x is defined to be $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{\frac{1}{p}}$.

For a matrix A, we denote by $A^\dagger$ its Moore–Penrose inverse, $A^\top$ its transpose, $\sigma_i(A)$ its i-th singular value, sorted in the descending order, $A_{i,*}$ its i-th row of A and $A_{*,j}$ its j-th column. We define the operator norm of A, denoted by $\|A\|_2$, to be $\|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2$. We denote $\mathrm{nnz}(A)$ the number of nonzero entries in matrix A. For two symmetric matrices A and B, we write that $A \preceq B$ if $B - A$ is positive semi-definite.

Suppose that a matrix A has the singular value decomposition $A = \sum_{i=1}^r \sigma_i(A) u_i v_i^\top$, where r is the rank of A and u_i, v_i are the singular vectors. By the Eckart-Young Theorem, for every rotationally invariant norm $\|\cdot\|$, the matrix $\sum_{i=1}^k \sigma_i(A) u_i v_i^\top$ is the minimizer for $\min_{\mathrm{rank}(A')\leq k} \|A - A'\|$. Therefore, we call $\sum_{i=1}^k \sigma_i(A) u_i v_i^\top$ the best rank-k approximation to A, denoted by $[A]_k$.

We write $a = (1 \pm \epsilon)b$ if $(1-\epsilon)b \leq a \leq (1+\epsilon)b$ and $a \lesssim_{t_1,t_2,\ldots} b$ if there exists a constant C depending only on $t_1, t_2, \ldots$ such that $a \leq Cb$. We also write $a \sim_{t_1,t_2,\ldots} b$ if $a \lesssim_{t_1,t_2,\ldots} b$ and $b \lesssim_{t_1,t_2,\ldots} a$.

1.2.1 Concentration Inequalities

Two useful inequalities are Bernstein inequalities of scalars and of matrices. Both are taken from [75].

Lemma 1.2.1 (Scalar Bernstein)**.** *Let $X_1, \ldots, X_s$ be i.i.d. random variables such that $\mathbb{E}\, X_i = 0$, $\mathbb{E}\, X_i^2 \leq \sigma^2$ and $|X_i| \leq \rho$ almost surely for some constant M. For all $\epsilon > 0$, it holds that*

$$\Pr\left\{\left|\frac{1}{s}\sum_{i=1}^s X_i\right| > \epsilon\right\} \leq 2\exp\left(-\frac{s\epsilon^2/2}{\sigma^2 + \rho\epsilon/3}\right).$$

Lemma 1.2.2 (Matrix Bernstein)**.** *Let X_1, …, X_s be s independent copies of a symmetric random matrix $X \in \mathbb{R}^{d\times d}$ with $\mathbb{E}\, X = 0$, $\|X\|_2 \leq \rho$ and $\|\mathbb{E}\, X^2\|_2 \leq \sigma^2$. For all $\epsilon > 0$, it holds that*

$$\Pr\left\{\left\|\frac{1}{s}\sum_{i=1}^s X_i\right\|_2 > \epsilon\right\} \leq 2d \cdot \exp\left(-\frac{s\epsilon^2/2}{\sigma^2 + \rho\epsilon/3}\right).$$

We shall also need Freedman's inequality for martingales [29]. The form we cite in the following lemma is taken from [74].

Lemma 1.2.3 (Freedman's inequality)**.** *Let $Y_0, Y_1, \ldots, Y_n$ be a martingale with the difference sequence $X_1, \ldots, X_n$. Assume that $|X_i| \le \rho$ almost surely for all $i = 1, \ldots, n$. Define the predictable quadratic variation difference of the martingale*

$$W_k = \sum_{i=1}^{k} \mathbb{E}(X_i^2 | X_1, \ldots, X_{i-1}), \quad k = 1, \ldots, n.$$

Then for all $t > 0$ and $\sigma^2 > 0$,

$$\Pr\left\{Y_n \ge t \text{ and } W_n \le \sigma^2\right\} \le \exp\left(-\frac{t^2/2}{\sigma^2 + \rho t/3}\right).$$

From Chapter 3 onwards, we shall repeatedly need the tail bound of Dudley's integral. The following form is cited from [77, Problem 8.1.7].

Lemma 1.2.4 (Dudley's integral, tail bound)**.** *Let X_t be a zero-mean stochastic process that is subgaussian w.r.t. a pseudo-metric d on the indexing set T. Then it holds for all $u \ge 0$ that*

$$\Pr\left\{\sup_{t,s\in T} |X_t - X_s| > C\left(\int_0^\infty \sqrt{\log N(T, d, \epsilon)}d\epsilon + u\,\mathrm{diam}(T)\right)\right\} \le 2\exp(-u^2),$$

where C is an absolute constant.

The preceding tail bound provides a subgaussian tail, which can be converted into a moment bound through integration. The following proposition states this basic fact.

Proposition 1.2.5. *Let X be a subgaussian variable such that $\Pr\{|X| > t\} \le Ce^{-c^2}$ for some constants $C, c > 0$ and every $t > 0$. Then $\left(\mathbb{E}\,|X|^\ell\right)^{1/\ell} \le K\sqrt{\ell}$ for all $\ell \ge 1$, where K is a constant that depends on C and c only.*

1.2.2 Subspace Embeddings

Originally introduced as a purely mathematical problem in geometric functional analysis, *subspace embedding* now serves as a fundamental component in many efficient algorithms for solving large-scale linear algebraic problems. The following is a formal definition, described in the language of linear algebra.

Definition 1.2.6 (Subspace embeddings)**.** Given $p \ge 1$, a distortion parameter $\epsilon \in [0, 1)$ and a matrix $A \in \mathbb{R}^{n\times d}$, we say a matrix $S \in \mathbb{R}^{m\times n}$ is a $(1+\epsilon)$-subspace-embedding for A in the ℓ_p-norm if it holds

$$\frac{1}{1+\epsilon}\|SAx\|_p \le \|Ax\|_p \le (1+\epsilon)\|SAx\|_p \tag{1.1}$$

for all $x \in \mathbb{R}^d$.

Note that S can be viewed as a linear map from $\mathbb{R}^n$ to $\mathbb{R}^m$ and it preserves the ℓ_p-norm of all vectors in the column space of A, which is a subspace of $\mathbb{R}^n$ of dimension at most d, up to a factor of $1+\epsilon$. In other words, S embeds the column space of A from $\mathbb{R}^n$ to $\mathbb{R}^m$ with a small distortion of $(1+\epsilon)/\frac{1}{1+\epsilon} = 1 + \Theta(\epsilon)$ for small ϵ, hence the term 'subspace embedding'.

Since $1/(1+\epsilon) = 1 - \Theta(\epsilon)$ for small ϵ, it is common to use

$$(1-\epsilon)\|SAx\|_p \le \|Ax\|_p \le (1+\epsilon)\|SAx\|_p \tag{1.2}$$

instead of (1.1) in the definition of a subspace embedding.

The key question regarding subspace embedding is determining the value of m, which is the number of rows in S and represents the target dimension of the embedding.

Problem. *Given $p \ge 1$, a dimension parameter d and a small distortion parameter ϵ, what is the smallest $N = N_p(d,\epsilon)$ such that for every matrix $A \in \mathbb{R}^{n\times d}$, there exists a $(1+\epsilon)$-subspace embedding $S \in \mathbb{R}^{N\times n}$ for A?*

For a comprehensive survey of this problem, we refer readers to [43]. Below, we briefly review the existing results.

The case of $p = 2$ is immediate, where one can take the QR decomposition of $A = QR$ with $Q \in \mathbb{R}^{n\times d}$ having orthonormal columns and $R \in \mathbb{R}^{d\times d}$ being upper triagonal, Taking $S = Q^\top$ gives an isometric embedding, i.e. $\epsilon = 0$. In fact, when p is an even integer, there always exists an isometric embedding with $n = \binom{d+p-1}{p} - 1$ (see, e.g. [46]). In general, the best existing upper bounds on $N_p(d,\epsilon)$ are as follows.

Lemma 1.2.7. *The following bounds on $N_p(d,\epsilon)$ hold.*

$$N_p(d,\epsilon) \le \begin{cases} C\epsilon^{-2} d\log d & p = 1 \\ C\epsilon^{-2} d(\log \epsilon^{-2} d)(\log\log \epsilon^{-2} d + \log(1/\epsilon))^2 & p \in (1,2) \\ C_p \epsilon^{-2} d^{p/2} \log^2 d \log(d/\epsilon) & p \in (2,\infty)\setminus 2\mathbb{Z} \\ C\epsilon^{-2}(10d/p)^{p/2} & p \in 2\mathbb{Z}, \end{cases} \tag{1.3}$$

where $C > 0$ is an absolute constant and $C_p > 0$ is a constant that depends only on p.

The result for $p = 1$ and $p \in (1,2)$ are due to Talagrand and can be found in [71] and [72], respectively. The result for $p > 2$ is presented in the famous monograph by Ledoux and Talagrand [47, Section 15], which also includes a simpler proof for the case $p \le 2$, albeit with slightly worse logarithmic factors. The result for even integers p is due to Schechtman [68].

We remark that the dependence on d in the results of Lemma 1.2.7 is tight for $p \in 2\mathbb{Z}$ and tight up to logarithmic factors for other values of p. The right dependence on ϵ has been a long-standing open problem; see [43, p845] for a discussion.

When d is a small constant, it is known that the ϵ-dependence can be better than $1/\epsilon^2$. The case of $p = 1$ has been near perfectly solved, as it is now known that

$$\epsilon^{-2(d-1)/(d+2)} \lesssim_d N_1(d,\epsilon) \lesssim_d \begin{cases} (\epsilon^{-2}\log(1/\epsilon))^{-(d-1)/(d+2)}, & \text{if } d = 3,4; \\ \epsilon^{-2(d-1)/(d+2)}, & \text{if } d = 2 \text{ or } d \ge 5. \end{cases} \tag{1.4}$$

The lower bound is due to Bourgain et al. [6], the upper bound for $d \ge 5$ to Matoušek [56] and the upper bound for $d = 3, 4$ to [5, 56]. The proof of the lower bound has been generalized recently by Li et al. [52] to show that the lower bound

$$N_p(d,\epsilon) \gtrsim_d \epsilon^{-2(d-1)/(d+2p)}.$$

holds for all $p \in [1,\infty)\setminus(2\mathbb{Z})$ and a near-tight upper bound

$$N_p(d,\epsilon) \lesssim_d (\epsilon^{-2}\log(\epsilon^{-1}))^{(d-1)/(d+2p)}$$

holds for all odd integers p.

For larger $d \gtrsim \log(1/\epsilon)$, another recent work by Li et al. [50] shows that $N_p(d, \epsilon) \gtrsim \epsilon^{-2}/\operatorname{poly}(\log(1/\varepsilon))$ for all $p \geq [1, \infty) \setminus (2\mathbb{Z})$, which is optimal up to logarithmic factors. However, a lower bound of $\tilde{\Omega}(d/\epsilon^2)$ or even $\tilde{\Omega}(d/\epsilon)$ remains an open question.

1.2.3 Lewis Weight Sampling

For general p, the upper bounds in Lemma 1.2.7 are traditionally established by a technique known as Lewis's "change of density" and an iterative argument [43]. Let $E \subseteq \mathbb{R}^n$ denote the column space of A. Consider $p > 2$ for example. Roughly speaking, one can find weights $w_1, \ldots, w_n$ for E such that $\|\cdot\|_{w,2} \leq d^{1/2-1/p}\|\cdot\|_{w,p}$. This property does not generally hold for the standard ℓ_p norm because the underlying space is n-dimensional. These weights, now called ℓ_p-Lewis weights in the literature of computer science, facilitate a proof, via a chaining argument, that, with high probability, randomly subsampling coordinates (using a sampling distribution based on these weights) induces a low-distortion isomorphism between E and its restriction on the sampled coordinates. Repeating this process progressively reduces the dimension while increasing the distortion until the distortion meets the desired threshold. We remark that a non-iterative argument for $p > 2$ is given by Bourgain et al. [6], but the upper bound on N is $\tilde{O}(d^{p/2}/\epsilon^5)$, which has a much worse dependence on ϵ.

While the iterative argument satisfactorily resolves the subspace embedding problem from the mathematical perspective, it falls short from a computational one. First, computing the Lewis weight involves maximizing the volume of non-Euclidean ellipsoids and is thus not computationally efficient. Second, the process gives only an implicit subspace embedding due to its iterative nature, making it challenging to construct the subspace embedding S explicitly. A breakthrough occurred about a decade ago when Cohen and Peng [16] showed that a simple iteration calculates the ℓ_p-Lewis weights up to a constant factor for $p \leq 4$. They also showed that sampling the rows of A according to these approximate Lewis weights is sufficient to obtain a subspace embedding. More recently, Fazel et al. [28] designed an efficient algorithm for calculating the Lewis weights when $p > 4$. The Lewis weights for all values of $p \geq 1$ can now be approximated up to a constant factor in near-input-sparsity time, i.e. $\tilde{O}(\operatorname{nnz}(A) + \operatorname{poly}(d))$.

Below we give a brief review of Lewis weights sampling. We first define the Lewis weights. Here we take the definition from [16]. A more systematic study of Lewis weights can be found in mathematical texts on Banach spaces, e.g., [79].

Definition 1.2.8 (Lewis weights)**.** Suppose that $A \in \mathbb{R}^{n\times d}$ and $p \geq 1$. The ℓ_p Lewis weights of A, denoted by $w_1(A), \ldots, w_n(A)$, are the unique real numbers such that

$$w_i(A) = (a_i^\top (A^\top W^{1-2/p} A)^\dagger a_i)^{p/2},$$

where W is the diagonal matrix with diagonal elements $w_1(A), \ldots, w_n(A)$ and $a_i \in \mathbb{R}^d$ is the i-th row of A (viewed as a column vector).

Note that all Lewis weights $w_i(A) \in [0, 1]$. The ℓ_2 Lewis weight is also called the leverage score. The following lemma summarizes a few useful properties of Lewis weights. See, e.g., [79] for a proof.

Lemma 1.2.9. *Suppose that $A \in \mathbb{R}^{n\times d}$ have full column rank and $p > 2$. Let $W = \operatorname{diag}\{w_1(A), \ldots, w_n(A)\}$. It holds that*

(1) $\sum_i w_i = d$;

(2) $\|W^{\frac{1}{2}-\frac{1}{p}}Ax\|_2 \leq d^{\frac{1}{2}-\frac{1}{p}}\|W^{-\frac{1}{p}}Ax\|_{w,p}$;

(3) $\|W^{-\frac{1}{p}}Ax\|_\infty \leq \sqrt{d}\|W^{-\frac{1}{p}}Ax\|_{w,p}$.

Below we define rescaled sampling matrices for both the first and the second kind of sampling scheme.

Definition 1.2.10 (Rescaled Sampling Matrix of First Kind)**.** Suppose that $p_1, \ldots, p_n \geq 0$ such that $p_1 + p_2 + \cdots + p_n = 1$. A diagonal matrix $S \in \mathbb{R}^{n\times n}$ is called a rescaled sampling matrix of the first kind if $S_{ii} = p_i^{-1/p}$ with probability p_i and $S_{ii} = 0$ with probability $1 - p_i$. The number m of nonzero rows in S is called the sample size.

Definition 1.2.11 (Rescaled Sampling Matrix of Second Kind)**.** Given a sample size m and $p_1, \ldots, p_n \geq 0$ such that $p_1 + p_2 + \cdots + p_n = 1$. A matrix $S \in \mathbb{R}^{m\times n}$ is called a rescaled sampling matrix of the second kind if the rows of S are i.i.d. copies of random vector X, where $X = (mp_j)^{-1/p}e_j^\top$ with probability p_j, $j = 1, \ldots, n$.

Note that a rescaled sampling matrix gives an unbiased estimator, i.e. satisfying that $\mathbb{E}\,\|SAx\|_p^p = \|Ax\|_p^p$ for all x. The Lewis weight sampling scheme used by Cohen and Peng [16] is of the second kind. We restate their result below.

Theorem 1.2.12 (Lewis weight sampling, [16, Theorem 7.1])**.** *Given $A \in \mathbb{R}^{n\times d}$. Suppose that $t_i \geq \beta w_i$ for all $i \in [n]$, where*

$$\beta \gtrsim_p \begin{cases} \frac{1}{\epsilon^2}(\log\frac{d}{\epsilon}\log^2(\log\frac{d}{\epsilon}) + \log\frac{1}{\delta}), & 1 \leq p < 2, p \neq 1 \\ \frac{1}{\epsilon^2}\log\frac{d}{\delta}, & p = 1, 2 \\ \frac{d^{\frac{p}{2}-1}}{\epsilon^5}(\log\frac{d}{\epsilon} + \log\frac{1}{\delta}) & 2 < p < \infty \end{cases}$$

is the oversampling parameter. Let $m = \sum_{i=1}^n t_i$. If $S \in \mathbb{R}^{m\times n}$ is a rescaled sampling matrix of the second kind with sampling probability $p_i = \frac{t_i}{m}$ for all i, then S is an $(1+\epsilon)$-subspace-embedding matrix for A in the ℓ_p-norm with probability at least $1 - \delta$.

Remark. Theorem 1.2.12 also holds when S is a rescaled sampling matrix of the first kind. Specifically, let β be the same as in Theorem 1.2.12 and suppose that $p_1, \ldots, p_n$ satisfy that $\min\{\beta w_i(A), 1\} \leq p_i \leq 1$ for each i. Let S be a rescaled sampling matrix of the first kind with probabilities $p_1, \ldots, p_n$. Then, S is an $(1+\epsilon)$-subspace-embedding matrix for A in the ℓ_p-norm with probability at least $1 - \delta$.

The case of $p \leq 2$ is proved by [16], based on the bounds in Lemma 1.2.7, while the case of $p > 2$ is claimed to be implicit in [6]. Note that when $p > 2$, the resulting sample size is $O(d^{p/2}/\epsilon^5 \cdot \log d \log(1/\epsilon))$ for constant δ, as opposed to the improved bound of $\tilde{O}(d^{p/2}/\epsilon^2)$ mentioned in Lemma 1.2.7. Woodruff and Yasuda [82] improved the sample complexity to $O(d^{p/2}/\epsilon^2 \cdot \log^2 d \cdot \log n)$, achieving a good dependence on ϵ but introducing an undesirable $\log n$ factor.

1.3 Randomized Numerical Linear Algebra

In this section, we give a brief review of linear regression and low-rank approximation problems, which are two important problems in randomized numerical linear algebra and

the focus of this thesis. The rapid growth of machine learning has lead to a huge increase in the demand for computational resources. Machine learning methods are particularly well-suited to the probabilistic and approximate techniques found in randomized numerical linear algebra. Because randomization is both cheap and highly effective, it has gained significant attention and will continue to play an important role as we look for ways to meet the growing need for computational resources [58]. For comprehensive and systematic studies of randomized numerical linear algebra, we refer readers to monographs such as [81] and [22].

Linear Regression Given $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^d$, where each row $A_{i,*}$ in A represents a data point with d features and the corresponding coordinate b_i of b represents the associated label, linear regression in the ℓ_p-norm is to solve $\min_{x\in\mathbb{R}^d} \|Ax-b\|_p$. We assume $n \gg d$ and the problem is over-constrained. When $p = 1$, this is a linear programme; when $p = 2$, classical methods such as QR decomposition take $O(nd^2)$ runtime; for other values of p, convex programme solvers are typically used, but they run in time $O((nd)^c)$ for some $c > 1$ and are thus prohibitive for large n. A natural idea is to reduce n, thereby reducing the original linear regression problem to a smaller one where classical methods can be applied efficiently. When $p = 1$, the ℓ_1 regression provides robustness to outliers; when $p < 2$, the regression is less sensitive to outliers because the loss function are not squared and when $p > 2$, the regression is even more sensitive to outliers. The ℓ_p regression for $p > 1$ has found various applications in machine learning, such as data clustering [27] and graph based semi-supervised learning [1].

Notice that $Ax - b$ always lies in the column space of the concatenated matrix $A' = (A \quad b)$. If S is a $(1+\epsilon)$-subspace embedding matrix for A' in the ℓ_p-norm, then $\|SAx - Sb\|_p = (1 \pm \epsilon)\|Ax - b\|_p$ for all x. This allows one to solve the sketched version $\tilde{x} = \arg\min_{x\in\mathbb{R}^d} \|SAx - Sb\|_p$, which has significantly fewer rows. To see that $\tilde{x}$ is a good solution to the original regression problem, observe that

$$\begin{aligned}\|A\tilde{x} - b\|_p \le \frac{1}{1-\epsilon}\|SA\tilde{x} - Sb\|_p \le \frac{1}{1-\epsilon}\|SAx^* - Sb\|_p \\ \le \frac{1+\epsilon}{1-\epsilon}\|Ax^* - b\|_p \le (1+3\epsilon)\|Ax^* - b\|_p\end{aligned}$$

for $\epsilon \in (0, 1/3)$. In other words, $\tilde{x}$ is a good solution in the sense that the objective function $\|A\tilde{x} - b\|_p$ is very close to the optimal value.

Earlier works primarily focused on the case of $p = 2$. The idea of applying subspace embeddings was first employed by Sarlos [67] in his pioneering work on numerical randomized linear algebra, where he used a Johnson-Lindenstrauss Transform matrix S, which is an oblivious sketch (the distribution of S does not depend on A and b). Later, more subspace embeddings for $p = 2$ were constructed. Non-oblivious sketches such as leverage score sampling matrices [23] and oblivious ones like OSNAP [62] and Count-Sketch [15] were developed. Dasgupta et al. [20] used two-stage sampling algorithm for general ℓ_p regression when $p > 1$. Research into the general ℓ_p Lewis weight row sampling began with the groundbreaking work of Cohen and Peng [16]. Beyond ℓ_p loss functions, several studies have generalized row sampling techniques to non-norm functions such as Huber losses, quantile losses and semi-norms [61, 49, 38, 39], extending the applicability of these techniques to a broader range of problems.

Low-Rank Approximation Low-rank approximation is at the core of many data modelling problems; see, for example, [55] for a comprehensive study. In its simplest form, given a matrix $A \in \mathbb{R}^{n\times n}$, the goal is to find a rank-k matrix $[A]_k$ such that

$$[A]_k = \underset{\text{rank}(A')\leq k}{\arg\min} \|A - A'\|.$$

where A' is an $n \times n$ matrix and $\|\cdot\|$ is a matrix norm that is typically unitarily invariant. When n is large, directly outputting an $n \times n$ matrix A' is already computationally expensive. Therefore, it is preferable to seek a factorization $A' = BC$, where $B \in \mathbb{R}^{n\times k}$ and $C \in \mathbb{R}^{k\times n}$, and output the factors B and C.

The best known exact solution to rank-k approximation is via singular value decomposition, which takes $O(n^\omega)$ time, where $\omega < 2.38$ is the matrix multiplication exponent [2]. Therefore, it is desirable to have a runtime proportional to n^2, the size of the input matrix A, for which one would only ask for a $(1+\epsilon)$-approximate rank-k approximation $B \cdot C$ such that

$$\|A - BC\| \leq (1+\epsilon)\|A - [A]_k\|.$$

A significant amount of research has been done for the case where the matrix norm $\|\cdot\|$ is the Frobenius norm. In this case, a common approach is to find first a subspace $E \subset \mathbb{R}^n$ (dependent on A) with $\dim(E) = \text{poly}(k/\epsilon)$ such that a good rank-k approximation can be found within E. Specifically, there exists a rank-k matrix X such that every row of X lies in E and satisfies $\|X - A\|_F \leq (1+\epsilon)\|A - [A]_k\|_F$. This subspace E is usually obtained via a linear sketch, meaning that E is the row space of SA for some sketching matrix S. The construction of S largely overlaps with classical subspace embeddings; oblivious constructions include fast Johnson-Lindenstrauss Transform matrices [67] and Count-Sketch [15], while non-oblivious examples include the leverage score sampling matrix [25]. Randomized low-rank approximation is used in support vector machine to efficiently compute the kernel matrix and in principle component analysis to extract principal components, enabling the simplification of high-dimensional data analysis [36].

As mentioned previously, linear sketches are well-suited when A is given in the turnstile streaming model. However, for low-rank approximation of $f(A)$, where A is still given in the turnstile streaming model and f is a non-linear function applied entrywise to A, it becomes difficult to update the entries of $f(A)$ in small space, which poses a significant challenge.

1.4 Thesis Organization and Overview

Chapters 2 to 4 are devoted to three numerical linear algebraic problems. Chapter 5 discusses ℓ_p-subspace embeddings for $p > 2$. Below is an overview of each chapter.

Chapter 2: Entrywise-Transformed Low-Rank Approximation Given a large $n \times d$ matrix $A = (a_{ij})$, one would like to compute a low-rank approximation of $f(A) = (f(a_{ij}))$, the resulting matrix of a function f applied entrywise to A. An important special case in applications is the likelihood function $f(x) = \log(|x| + 1)$. The straightforward solution of applying f to each entry of A and then computing the low-rank approximation requires storing all of A as well as making multiple passes over its entries. The work of Liang et al. [53] shows how to find a rank-k factorization to $f(A)$ for an $n \times n$ matrix A using only $n \cdot \text{poly}(\epsilon^{-1}k \log n)$ words of memory, with an overall error of

$10\|f(A) - [f(A)]_k\|_F^2 + \text{poly}(\epsilon/k)\|f(A)\|_{1,2}^2$, where $[f(A)]_k$ is the optimal rank-k matrix approximating $f(A)$ in Frobenius norm and $\|f(A)\|_{1,2} = \sum_i \|(f(A))_{*,i}\|_2$. Their algorithm uses three passes over the entries of A. In this chapter, we present the first one-pass algorithm for this problem, for the same class of functions f studied by Liang et al. [53]. Moreover, our error is $\|f(A) - [f(A)]_k\|_F^2 + \text{poly}(\epsilon/k)\|f(A)\|_F^2$, which is significantly smaller, as it removes the factor of 10 and also $\|f(A)\|_F^2 \leq \|f(A)\|_{1,2}^2$. We also give an algorithm for regression, pointing out an error in previous work.

Chapter 3: Online Active Regression Active regression considers a linear regression problem $\min_x \|Ax - b\|_p$, where the matrix A is fully accessible, but the vector b can only be queried by its coordinates. The goal is to minimize the number of queries to b. In this chapter, we consider an online extension of the active regression problem: the matrix A is revealed row by row and a decision must be made immediately on whether to query the corresponding entry of b. We propose novel algorithms for this problem under ℓ_p loss, where $p \in [1, 2]$. Given an accuracy parameter ϵ, our proposed algorithms only require $\tilde{\mathcal{O}}(\epsilon^{-1} d \log(n\kappa))$ queries of labels, where n is the number of data points and κ is a quantity, called the condition number, of the data points. A technical contribution is that we prove one can compress a fraction of rows in a matrix by sampling these rows according to their Lewis weights while preserving the Lewis weights of the uncompressed rows (see Lemma 3.4.3 for a formal statement).

Chapter 4: Active Regression with Shared Labels Active learning for multiple target models aims to reduce the amount of labelled data querying while effectively training multiple models concurrently. One application in deep learning with multiple models is to consider different representations of the same dataset using distinct network backbones and to learn the linear prediction layer on each representation. Formulated as linear regression problems, this means simultaneously solving k instances of ℓ_p-regression problems $\min_x \|f(A^j x) - b\|_p$ for $j = 1, \ldots, k$, where f is a Lipschitz function. The data matrices $A^1, \ldots, A^k \in \mathbb{R}^{n \times d}$ are fully accessible, while the label vector $b \in \mathbb{R}^n$, shared by all data matrices, can only be queried by coordinates. The goal is to minimize the number of queries to b. We solve this problem using Lewis weight sampling and build on the previous work of [31], improving their query complexity by removing a factor of d and extending their result from the case of $p = 2$ to a general p.

Chapter 5: ℓ_p-Subspace Embeddings for $p > 2$ Recall the Lewis weight sampling lemma (Theorem 1.2.12) given by Cohen and Peng [16]. However, their paper [16] does not provide a clear proof for the case of $p > 2$. For completeness, we include a detailed proof in this chapter, based on the work of Bourgain et al. [6]. This proof gives a target dimension of $\tilde{O}(d^{p/2}/\epsilon^5)$. We then improve this bound to $\tilde{O}(d^{p/2}/\epsilon^2)$, matching the target dimension achieved by recursive sampling and removing an undesirable $\log n$ factor present in the result of Woodruff and Yasuda [82]. Our analysis is based upon Theorem 1.2.12 for constant-factor approximation (i.e., ϵ is a constant) and is simpler than the analysis of [82].

Chapter 2

Entrywise-Transformed Low-Rank Approximation

2.1 Introduction

In numerous applications with matrices, such as machine learning, image clustering and recommendation systems, a common goal is to compute a low-rank approximation to a large matrix $A \in \mathbb{R}^{n\times d}$. If the rank of the low-rank approximation is k, then one can approximate A as $U \cdot V$, where $U \in \mathbb{R}^{n\times k}$ and $V \in \mathbb{R}^{k\times d}$. This results in a parameter reduction, as U and V only have $(n+d)k$ parameters in total, as compared to the nd parameters required of A. Since $k \ll \min(n,d)$, this parameter reduction is significant. Not only does it result in much smaller storage, when multiplying A by a vector x, it also now only takes $O((n+d)k)$ time instead of $O(nd)$ time, since one can first compute $V \cdot x$ and then $U \cdot (V \cdot x)$.

A challenge in the above applications is that often wants to compute a low-rank approximation not to A, but to an *entrywise* transformation to A by a function f. Namely, if $A = (a_{i,j})$, then we define $f(A) = f(a_{i,j})$ where we apply the function f to each entry of A. Common functions f include $f(x) = \log_2^c(|x|+1)$ or $f(x) = |x|^\alpha$ for $0 \leq \alpha \leq 2$. Indeed, for word embeddings in natural language processing (NLP), an essential subroutine is to perform a low-rank approximation of a matrix after applying the log-likelihood function to each entry, which corresponds to $f(x) = \log_2(|x|+1)$.. Note that in NLP the input matrices are often word co-occurrence count matrices, which can be created, e.g., from the entire Wikipedia database. Thus, such matrices are huge, with millions of rows and columns, and hard to store in memory. This necessitates models such as the streaming model for processing such data.

We can indeed capture the above scenarios formally with the turnstile streaming model. Recall that in this model, there is a large underlying matrix A, and we see a long sequence of updates to its entries. Each pass over the data stream is very expensive, and thus one would like to minimize the number of such passes. Also, one would like to use as little memory as possible to compute a low-rank approximation of the transformed matrix $f(A)$ in this model. In this chapter we will consider approximately optimal low-rank approximations, meaning factorizations $U \cdot V$ for which $\|U \cdot V - f(A)\|_F^2 \leq \|[f(A)]_k - f(A)\|_F^2 + \mathrm{poly}(\epsilon/k)\|f(A)\|_F^2$, where $[f(A)]_k$ is the optimal rank-k matrix approximating $f(A)$ in Frobenius norm. Recall the Frobenius norm $\|B\|_F$ of a matrix B is defined to be $(\sum_{i,j} B_{i.j}^2)^{1/2}$, which is the entrywise Euclidean norm of B.

Although there is a body of work in the streaming model computing low-rank approxi-

mations of matrices [7, 14, 24, 76, 80], such methods no longer apply in our setting due to the non-linearity of the function f. Indeed, a number of existing methods are based on sketching, whereby one stores $S \cdot A$ for a random matrix S with a small number of rows. Given an entrywise transformation f, which may be nonlinear, it is not clear how to maintain $S \cdot f(A)$ in a stream.

A natural question is: *can we compute a low-rank approximation to $f(A)$ in the streaming model with a small number of passes, ideally one, and a small amount of memory, ideally $n \cdot poly(k/\epsilon)$ memory?*

Motivated by the applications above, this question was asked by Liang et al. [53]; see also earlier work which studies entrywise low-rank approximation in the distributed model [83]. The work of [53] studies the function $f(x) = \log_2(|x| + 1)$ and gives a three-pass algorithm for $n \times n$ matrices A achieving $n \cdot \text{poly}(\epsilon^{-1} k \log n)$ memory and outputting factors U and V with the following error guarantee:

$$\|U \cdot V - f(A)\|_F^2 \leq 10 \, \|[f(A)]_k - f(A)\|_F^2 + \text{poly}(\epsilon/k) \, \|f(A)\|_{1,2}^2 \,,$$

where for a matrix B, $\|B\|_{1,2} = \sum_i \|B_{*,i}\|_2$. We note that this error guarantee is considerably weaker than what we would like, as there is a multiplicative factor 10 and an additive error that depends on $\|f(A)\|_{1,2}$. Using the relationship between the 1-norm and the 2-norm, we have that $\|f(A)\|_{1,2}$ could be as large as $\sqrt{n}\|f(A)\|_F$, and so their additive error can be a $\sqrt{n}$ factor larger than what is desired. Also, although the memory is of the desired order, the fact that the algorithm requires 3 passes can significantly slow it down. Moreover, when data is crawled from the internet, e.g. in applications of network traffic, it may be impractical to store the entire data set [26]. Therefore, in these settings it is impossible to make more than one pass over the data. Liang et al. [53] say "Whether there exists a one-pass algorithm is still an open problem, and is left for future work."

2.1.1 Our Contributions

In this chapter, we resolve this main open question of [53], obtaining a *one*-pass algorithm achieving $(n + d) \cdot \text{poly}(\epsilon^{-1} k \log n)$ memory for outputting a low-rank approximation for the function $f(x) = \log_2(|x| + 1)$, and achieving the stronger error guarantee:

$$\|U \cdot V - f(A)\|_F^2 \leq \|[f(A)]_k - f(A)\|_F^2 + \text{poly}(\epsilon/k) \, \|f(A)\|_F^2 \,.$$

We note that the $\text{poly}(\epsilon/k)$ factor in both the algorithm of [53] and our algorithm can be made arbitrarily small by increasing the memory by a $\text{poly}(k/\epsilon)$ factor, and thus it suffices to consider error of the form $\|[f(A)]_k - f(A)\|_F^2 + \epsilon\|f(A)\|_F^2$. We also note that our algorithm can be trivially adapted to rectangular matrices, so for ease of notation, we focus on the case $n = d$.

At a conceptual level, the algorithm of [53] uses one pass to obtain so-called approximate leverage scores of $f(A)$, then a second pass to sample columns of $f(A)$ according to these, and finally a third pass to do so-called adaptive sampling. In contrast, we observe that one can just do squared column norm sampling of $f(A)$ to obtain the above error guarantee, which is a common method for low-rank approximation to A. However, in one pass it is not possible to sample actual columns of A or of $f(A)$ according to these probabilities, so we build a data structure to sample noisy columns by approximations to their squared norms in a single pass. This is related to block ℓ_2-sampling in a stream, see, e.g., [54]. However, the situation here is complicated by the fact that we must sample according to

the sum of squares of f values of entries in a column of A, rather than the squared length of the column of A itself. The transformation function f's nonlinearity makes many of the techniques considered in [54] inapplicable. To this end we build new hashing and sub-sampling data structures, generalizing data structures for length or squared length sampling from [3, 4, 48], and we give a novel analysis for sampling noisy columns of A proportional to the sum of squares of f values to their entries.

Finally, we apply our new sampling techniques to the regression problem, showing that our techniques are more broadly applicable.

2.2 Preliminaries

Notation. For a vector $x \in \mathbb{R}^n$, we denote by $|x|$ the vector whose i-th entry is $|x_i|$. For a function f, let $f(A)$ denote the entrywise-transformed matrix $(f(A))_{ij} = f(A_{ij})$.

Useful Inequalities. Below we list a few useful inequalities regarding the function $f(x) = \log(1 + |x|)$.

Proposition 2.2.1. *For $x > 0$, it holds that $\ln(1 + x) > x/(1 + 2x)$.*

Proof. Let $h(x) = (1+2x)\ln(1+x) - x$. Since $h(0) = 0$, it suffices to show that $h'(x) > 0$. We calculate that

$$h'(x) = \frac{x}{1+x} + 2\ln(1+x).$$

Since $h'(0) = 0$, it suffices to show that $h''(x) > 0$. This can be readily verified by calculating that

$$h''(x) = \frac{3+2x}{(1+x)^2} > 0. \qquad \square$$

Proposition 2.2.2. *It holds for all $x, y \in \mathbb{R}$ and all $a \geq 0$ that $f(x + y) \leq f(x) + f(y)$ and $f(ax) \leq af(x)$. As a consequence, for $x, y \in \mathbb{R}^n$ it holds that $\|f(x + y)\|_2^2 \leq (\|f(x)\|_2 + \|f(y)\|_2)^2$.*

Proposition 2.2.3. *It holds for all $x, y \geq 0$ that $f(\sqrt{x^2 + y^2})^2 \leq f(x)^2 + f(y)^2$.*

Proof. Let $f(x, y) = \ln^2(1 + x) + \ln^2(1 + y) - \ln^2(1 + \sqrt{x^2 + y^2})$. It suffices to show that $f(x, y) \geq 0$. The inequality is clearly true when $x = 0$ or $y = 0$. Note that

$$\frac{\partial f}{\partial x} = 2\left(\frac{\log(1+x)}{1+x} - \frac{x\ln(1+\sqrt{x^2+y^2})}{x^2+y^2+\sqrt{x^2+y^2}}\right)$$
$$\frac{\partial f}{\partial y} = 2\left(\frac{\log(1+y)}{1+y} - \frac{y\ln(1+\sqrt{x^2+y^2})}{x^2+y^2+\sqrt{x^2+y^2}}\right)$$

Assuming $x, y > 0$, $\partial f/\partial x = \partial f/\partial y = 0$ implies that

$$\frac{\log(1+x)}{x(1+x)} = \frac{\log(1+x)}{y(1+y)}.$$

It is easy to verify that $\log(1+x))/(x(1+x))$ is decreasing w.r.t. x (checking the derivative and using Proposition 2.2.3), so we must have $x = y$. Now, let

$$h(x) = \frac{\partial f}{\partial x}(x, x) = \frac{2\ln(1+x)}{1+x} - \frac{\sqrt{2}\ln(1+\sqrt{2}x)}{1+\sqrt{2}x}.$$

We shall show that $h(x) > 0$ for all $x > 0$. This will imply that $f(x, y)$ has no local minimum or maximum when $x, y > 0$ and so it is easy to see that $f(x, y)$ attains the minimum at its boundary $x = 0$ or $y = 0$, yielding that $f(x, y) \geq 0$ for all $x, y \geq 0$.

To see that $h(x) > 0$, let

$$g(a) = \frac{\ln(1 + ax)}{a(1 + ax)}.$$

We calculate

$$g'(a) = \frac{ax - (1 + 2ax)\ln(1 + ax)}{a^2(1 + ax)^2}.$$

It follows from Proposition 2.2.1 that $g'(a) < 0$. Hence $g(a)$ is decreasing w.r.t. a and $g(\sqrt{2}) < g(1)$, which is exactly $\frac{1}{\sqrt{2}}h(x) > 0$. □

Lemma 2.2.4. *Let $a_1, \ldots, a_m$ be real numbers and $\epsilon_1, \ldots, \epsilon_m$ be 4-wise independent random variables on the set $\{-1, 1\}$ (i.e., Rademacher random variables). It holds that*

$$\mathbb{E}\, f\left(\sum_i \epsilon_i a_i\right)^2 \leq C \sum_i f(a_i)^2.$$

where $C > 0$ is an absolute constant.

Proof. It is clear that the base of the logarithm does not matter and we assume that the base is e. Let $Z = \sum_i \epsilon_i a_i$ and $\sigma^2 = \sum a_i^2$. Then $\mathbb{E}\, Z^2 = \sigma^2$ and $\mathbb{E}\,|Z| \leq (\mathbb{E}\,|Z|^2)^{1/2} = \sigma$. Let $g(x) = \ln(1 + x)$ and

$$Z_1 = \begin{cases} |Z|, & |Z| \geq e - 1; \\ 0, & \text{otherwise}, \end{cases} \qquad Z_2 = \begin{cases} 0, & |Z| \geq e - 1; \\ |Z|, & \text{otherwise}. \end{cases}$$

Then $|Z| = Z_1 + Z_2$ and

$$\mathbb{E}\, g(|Z|)^2 = \mathbb{E}(g(Z_1 + Z_2))^2 \leq \mathbb{E}(g(Z_1) + g(Z_2))^2 \leq \mathbb{E}\, 2(g(Z_1)^2 + g(Z_2)^2),$$

where the first inequality follows from Proposition 2.2.2. For the first term, we define $h(x) = g(x) \cdot \mathbf{1}_{\{x \geq e-1\}}$. Then $h(x)^2$ is concave on $[0, \infty)$. Hence

$$\mathbb{E}\, g(Z_1)^2 = \mathbb{E}\, h(Z_1)^2 = \mathbb{E}\, h(|Z|)^2 \leq h(\mathbb{E}\,|Z|)^2 \leq h(\sigma)^2 \leq g(\sigma)^2.$$

Next we upper bound the second term. The first case is $\sigma \leq e - 1$. Since $\mathbb{E}\, Z^4 \leq 3\sigma^4$, it holds that $\Pr\{Z_2 \geq t\sigma\} \leq \Pr\{|Z| \geq t\sigma\} \leq 3/t^4$. Then

$$\begin{aligned}
\mathbb{E}\, g(Z_2)^2 &\leq \mathbb{E}\, g(e-1)g(Z_2) \\
&= \mathbb{E}\, g(Z_2) \\
&= \int_0^{e-1} g(x) \Pr\{Z_2 \geq x\} dx \\
&= \sigma \int_0^{(e-1)/\sigma} g(t\sigma) \Pr\{Z_2 \geq t\sigma\} dt \\
&= \sigma^2 \int_0^{(e-1)/\sigma} g(t) \Pr\{Z_2 \geq t\sigma\} dt \quad \text{(by Proposition 2.2.2)} \\
&\leq \sigma^2 \left(\int_0^1 g(t) dt + 3 \int_1^{(e-1)/\sigma} \frac{g(t)}{t^4} dt \right) \\
&\leq C_1 \sigma^2 \\
&\leq C_1 (e-1)^2 g(\sigma)^2,
\end{aligned}$$

where $C_1 > 0$ is an absolute constant and the last inequality follows from the fact that $g(x) \geq x/(e-1)$ on $[0, e-1]$. The second case is $\sigma > e-1$. In this case,

$$\mathbb{E}\, g(Z_2)^2 \leq 1 \leq g(\sigma)^2.$$

Therefore, we conclude that

$$\mathbb{E}\, g(|Z|)^2 \leq 2(1 + C_1(e-1)^2) g(\sigma)^2 = C_2 g\left(\sqrt{\sum_i a_i^2}\right)^2 \leq C_2 \sum_i g(|a_i|)^2,$$

where the last inequality follows from Proposition 2.2.3. □

Lemma 2.2.5. *For arbitrary vectors $y, z \in \mathbb{R}^n$ such that $\|f(y)\|_2^2 \geq \xi^{-2}\|f(z)\|_2^2$ for some $\xi \in (0,1)$, it holds that $(1 - 3\xi^{2/3})\|f(y)\|_2^2 \leq \|f(y+z)\|_2^2 \leq (1+3\xi)\|f(y)\|_2^2$.*

Proof. We first prove the upper bound.

$$\begin{aligned}
\|f(y+z)\|_2^2 &= \sum_i f(y_i + z_i)^2 \\
&\leq \sum_i [f(y_i) + f(z_i)]^2 \quad \text{(Proposition 2.2.2)} \\
&= \sum_i f(y_i)^2 + \sum_i f(z_i)^2 + \sum_i 2 f(y_i) f(z_i) \\
&\leq \|f(y)\|_2^2 + \xi^2 \|f(y)\|_2^2 + 2\|f(y)\|_2 \|f(z)\|_2 \quad \text{(Cauchy-Schwarz)} \\
&\leq \left(\xi^2 + 2\xi + 1\right) \|f(y)\|_2^2 \\
&\leq (1 + 3\xi) \|f(y)\|_2^2 . \quad \text{(since } \xi < 1)
\end{aligned}$$

Next we prove the lower bound. Let $I = \{i : y_i z_i \leq 0\}$, $J_1 = \{i \in I : |y_i| \leq |z_i|\}$ and $J_2 = \{i \in I : |z_i| < |y_i| \leq \zeta^{-1}|z_i|\}$ for some $\zeta < 1$ to be determined. Then

$$\begin{aligned}
\|f(y+z)\|_2^2 &= \sum_{i \in J_1} f(y_i + z_i)^2 + \sum_{i \in J_2} f(y_i + z_i)^2 + \sum_{i \in I \setminus (J_1 \cup J_2)} f(y_i + z_i)^2 + \sum_{i \notin I} f(y_i + z_i)^2 \\
&\geq \sum_{i \in I \setminus (J_1 \cup J_2)} f(y_i + z_i)^2 + \sum_{i \notin I} f(y_i)^2 .
\end{aligned}$$

When $i \in I \setminus (J_1 \cup J_2)$, we have $|z_i| \leq \zeta |y_i|$. It then follows that

$$\log(|y_i + z_i| + 1) \geq \log((1-\zeta)|y_i| + 1) \geq (1-\zeta)\log(|y_i| + 1),$$

where, for the last inequality, one can easily verify that $h_\epsilon(x) = \frac{\log(1+(1-\epsilon)x)}{\log(1+x)}$ is increasing on $[0, \infty)$ and $\lim_{x \to 0^+} h_\epsilon(x) = 1 - \epsilon$. Hence

$$\sum_i f(y_i + z_i)^2 \geq (1-\zeta)^2 \sum_{i \in I \setminus (J_1 \cup J_2)} f(y_i)^2 + \sum_{i \notin I} f(y_i)^2 \geq (1-\zeta)^2 \sum_{i \notin J_1 \cup J_2} f(y_i)^2 .$$

Now, note that

$$\sum_{i \in J_1} f(y_i)^2 \leq \sum_{i \in J_1} f(z_i)^2 \leq \|f(z)\|_2^2 \leq \xi^2 \|f(y)\|_2^2$$

and (using Proposition 2.2.2)

$$\sum_{i\in J_2} f(y_i)^2 \le \zeta^{-2}\sum_{i\in J_1} f(z_i)^2 \le \zeta^{-2}\|f(z)\|_2^2 \le (\zeta^{-1}\xi)^2\,\|f(y)\|_2^2\,.$$

It follows that

$$\begin{aligned}\sum_i f(y_i+z_i)^2 &\ge (1-\zeta)^2\left(\|f(y)\|_2^2 - \xi^2\,\|f(y)\|_2^2 - (\zeta^{-1}\xi)^2\,\|f(y)\|_2^2\right)\\ &= (1-\zeta)^2(1-\xi^2-(\zeta^{-1}\xi)^2)\,\|f(y)\|_2^2\,.\end{aligned}$$

Choosing $\zeta = (\xi^2/(1-\xi^2))^{1/3}$ maximizes the right-hand side, yielding

$$\|f(y+z)\|_2^2 \ge (1-3\xi^{2/3})\,\|f(y)\|_2^2\,. \qquad \square$$

2.3 Algorithm

Our algorithm uses two important subroutines: a subsampling data structure called an H-Sketch, and a sketch for approximating the inner product of a transformed vector and a raw vector called LogSum. The former is inspired from a subsampling algorithm of [48] and is meant to sample a noisy approximation to a column from a distribution which is close to the desired distribution. In fact, one can show that it is impossible to sample the actual columns in a single pass, hence, we have to resort to noisy approximations and show they suffice. The latter LogSum sketch is the same as in [53], which approximates the inner product $\langle f(x), y\rangle$ for vectors x, y. Executing these sketches in parallel is highly non-trivial since the subsampling algorithm of [48] samples columns of A according to their ℓ_2 norms, but here we must sample them according to the squares of their ℓ_2 norms after applying f to each entry.

Roughly speaking, the above combination gives us a small set of poly(k/ϵ) noisy columns of $f(A)$, sampled approximately from the squared ℓ_2 norm of each column of $f(A)$, after which we can appeal to squared column-norm sampling results for low-rank approximation in [30], which argue that if you then compute the top-k left singular vectors of $f(A)$, forming the columns of the $n \times k$ matrix L, then $LL^\top f(A)$ is a good rank-k approximation to $f(A)$. The final output of the low-rank approximation will be two factors, L and $L^\top f(A)$. The algorithm in [53] first computes L by an involved algorithm in three passes, and then computes $L^\top f(A)$ in another pass using LogSum sketches. Our algorithm follows the same outline but we shall demonstrate how to compute L in only one pass, which is our sole focus for low-rank approximation in this chapter. Note that our ultimate goal, which we only achieve approximately, is to sample columns of $f(A)$ proportional to their squared ℓ_2 norms. This is a fundamentally different sampling scheme from that of [53], which performs leverage score sampling followed by adaptive sampling, which are not amenable to a one-pass implementation.

2.3.1 H-Sketch

We first present a basic heavy hitter structure in Algorithm 2.1, and a complete heavy hitter structure in Algorithm 2.2 by repeating the basic structure R times. The complete heavy hitter structure supports a query function. Below we analyse the guarantee of this heavy hitter data structure.

Algorithm 2.1 Basic heavy hitter substructure

Input: $A \in \mathbb{R}^{n\times n}$, ν, ϕ
Output: a data structure H

$w \leftarrow O(1/(\phi^2\nu^3))$
Prepare a pairwise independent hash function $h : [n] \to [w]$
Prepare 4-wise independent random signs $\{\epsilon_i\}_{i=1}^n$
Prepare a hash table H with w buckets, where each bucket stores a vector in $\mathbb{R}^n$.
for each $v \in [w]$ **do**
    $H_v \leftarrow \sum_{i\in h^{-1}(v)} \epsilon_i A_i$
end for
return H

Algorithm 2.2 Complete H-Sketch

Input: $A \in \mathbb{R}^{n\times n}$, ν, ϕ, δ
Output: a data structure H

$R \leftarrow O(\log(n/\delta))$
for each $r \in [R]$ **do**
    Initialize a basic substructure $H^{(r)}$ (Algorithm 2.1) with parameters ν and ϕ
end for

function QUERY(i)
    for each $r \in [R]$ **do**
        $v_r \leftarrow H^{(r)}_{h_r(i)}$ ▷ h_r is the hash function in $H^{(r)}$
    end for
    $r^* \leftarrow$ index r of the median of $\{\|f(v_r)\|_2\}_{r\in[R]}$
    return v_{r^*}
end function

Let $M = \|f(A)\|_F^2$. We define $I_\epsilon = \{i \in [n] : \|f(A_i)\|_2^2 \geq \epsilon M\}$, the set of the indices of the ϵ-heavy columns. Let α be a small constant to be determined later.

Lemma 2.3.1. *With probability at least* 0.9*, each column in $I_{\alpha\phi}$ is mapped to a distinct bucket by h.*

Proof. Note that $|I_{\alpha\phi}| \leq 1/(\alpha\phi)$. Thus, there exists a collision with probability at most

$$\frac{1}{w}\binom{1/(\alpha\phi)}{2} \leq \frac{1}{2w\alpha^2\phi^2} \leq 0.1,$$

provided that $w \geq 1/(0.2 \cdot \alpha^2\phi^2) = 5/(\alpha^2\phi^2)$. □

Lemma 2.3.2. *For each $u \in [n]$, it holds with probability at least* 2/3 *that*

$$\left\| f\left(\sum_{i\notin(I_{\alpha\phi}\cup\{u\})} \mathbf{1}_{\{h(i)=h(u)\}}\epsilon_i A_i \right) \right\|_2^2 \leq 3C\frac{M}{w},$$

where $C > 0$ is an absolute constant.

Algorithm 2.3 Sampling using H-Sketch

Require: (i) An estimate $\widehat{M}$ such that $M \le \widehat{M} \le KM$; (ii) a complete heavy hitter structure $\mathcal{D}_0$ of parameters $(O(1), O(\epsilon^3/(KL^3)), 1/(\hat{L}+1))$; (iii) $\hat{L}$ complete heavy hitter structures (see Algorithm 2.1), denoted by $\mathcal{D}_1, \dots, \mathcal{D}_{\hat{L}}$, where $\mathcal{D}_j$ ($j \in [\hat{L}]$) has parameters $(O(1), O(\epsilon^3/L^3), 1/(\hat{L}+1))$ and is applied to the columns of A downsampled at rate 2^{-j};

$L \leftarrow \log(Kn/\epsilon)$, $\hat{L} \leftarrow \log n$
$\zeta \leftarrow$ a random variable uniformly distributed in $[1/2, 1]$
for $j = 0, \dots, \hat{L}$ **do**
    $\Lambda_j \leftarrow$ top $\Theta(L^3/\epsilon^3)$ heavy hitters from $\mathcal{D}_j$
end for
$j_0 \leftarrow \log(4K\epsilon^{-3}L^3)$
$\zeta \leftarrow$ uniform variable in $[1/2, 1]$
for $j = 1, \dots, j_0$ **do**
    Let $\lambda_1^{(j)}, \dots, \lambda_s^{(j)}$ be the elements in Λ_0 contained in $[(1+\epsilon)\zeta\frac{\widehat{M}}{2^j}, (2-\epsilon)\zeta\frac{\widehat{M}}{2^j}]$
    $\widetilde{M}_j \leftarrow |\lambda_1^{(j)}| + \cdots + |\lambda_s^{(j)}|$
end for
for $j = j_0 + 1, \dots, L$ **do**
    Find the largest ℓ for which Λ_ℓ contains s_j elements $\lambda_1^{(j)}, \dots, \lambda_{s_j}^{(j)}$ in $[(1+\epsilon)\zeta\frac{\widehat{M}}{2^j}, (2-\epsilon)\zeta\frac{\widehat{M}}{2^j}]$ for $(1-\sqrt{20}\epsilon)L^2/\epsilon^2 \le s_j \le 2(1+\sqrt{20}\epsilon)L^2/\epsilon^2$
    if such an ℓ exists **then**
        $\widetilde{M}_j \leftarrow (|\lambda_1^{(j)}| + \cdots + |\lambda_{s_j}^{(j)}|)2^\ell$
        $W_j \leftarrow \Lambda_\ell$
    else
        $\widetilde{M}_j \leftarrow 0$
    end if
end for
$j^* \leftarrow$ sample from $[L]$ according to pdf $\Pr(j^* = j) = \widetilde{M}_j / \sum_j \widetilde{M}_j$
$i^* \leftarrow$ sample from W_{j^*} according to pdf $\Pr(i^* = i) = |\lambda_i^{(j)}|/\widetilde{M}_j$
$v_{j^*,i^*} \leftarrow$ vector returned by QUERY(i^*) on $\mathcal{D}_{j^*}$
return v_{j^*,i^*}

Proof. Let $v = h(u)$. Since h is pairwise independent, $\Pr\{h(i) = v\} = 1/w$ for all $i \neq w$. Let

$$Z_v = \sum_{i \notin (I_{\alpha\phi} \cup \{u\})} \mathbf{1}_{\{h(i)=v\}} \|f(A_i)\|_2^2.$$

then

$$\mathbb{E}\, Z_v \le \sum_{i \notin I_{\alpha\phi}} \mathbb{E}\, \mathbf{1}_{\{h(i)=v\}} \|f(A_i)\|_2^2 \le \frac{M}{w}.$$

It follows from Lemma 2.2.4 that

$$\mathop{\mathbb{E}}_{\{\epsilon_i\},h} \left\| f\left(\sum_{i \notin I_{\alpha\phi}} \mathbf{1}_{\{h(i)=v\}} \epsilon_i A_i \right) \right\|_2^2 \le \mathop{\mathbb{E}}_{h} C \sum_{i \notin I_{\alpha\phi}} \left\| f\left(\mathbf{1}_{\{h(i)=v\}} A_i \right) \right\|_2^2 = C \mathop{\mathbb{E}}_{h} Z_v \le C\frac{M}{w},$$

where we used the fact that $f(0) = 0$ and $\mathbf{1}_{\{h(i)=v\}} \in \{0, 1\}$ in the second step (the equality). The result follows from Markov's inequality. □

Lemma 2.3.3. *Suppose that $\nu \in (0, 0.05]$ and $\alpha = 0.3/C > \beta$, where C is the absolute constant in Lemma 2.3.2. With probability at least $1 - \delta$, for all $i \in [n]$, the output v_{r^*} of Algorithm 2.2 satisfies that*

(a) $(1-\nu)\|f(A_i)\|_2^2 \le \|f(v_{r^*})\|_2^2 \le (1+\nu)\|f(A_i)\|_2^2$ *for all* $i \in I_\phi$;

(b) $\|f(v_{r^*})\|_2^2 \le 0.92\phi M$ *for all* $i \notin I_{\alpha\phi}$;

(c) $\|f(v_{r^*})\|_2^2 \le (1+\nu^{3/2})^2\phi M$.

Proof. Fix $u \in [n]$. With probability at least $0.9 - 1/3 > 0.5$, the events in Lemmas 2.3.1 and 2.3.2 happen. Condition on those events.

From the proof of Lemma 2.3.2, we know that $i \in I_\phi$ do not collide with other elements in $I_{\alpha\phi}$. Hence, it follows from Lemma 2.2.5 (where $\xi^2 \le 3C/(\phi w) \le (\nu/3)^3$) that

$$(1-\nu)\left\|f(A_i)\right\|_2^2 \le \left\|f(H_{h(i)})\right\|_2^2 \le (1+\nu)\left\|f(A_i)\right\|_2^2,$$

provided that $w \ge 3^4 C/(\phi\nu^3)$.

When $i \notin I_{\alpha\phi}$, we have

$$\left\|f(H_{h(i)})\right\|_2^2 \le 3C\left(\alpha\phi + \frac{1}{w}\right)M \le (0.9 + \nu^{3/2})\phi M \le 0.92\phi M.$$

When $i \in I_{\alpha\phi} \setminus I_\phi$, we have that H_i contains only i and columns from $[n] \setminus I_{\alpha\phi}$. Hence by Proposition 2.2.2,

$$\begin{aligned}\left\|f(H_{h(i)})\right\|_2^2 &\le \left(\|f(A_i)\|_2 + \sqrt{\frac{3C}{w}M}\right)^2 \\ &\le \left(\sqrt{\phi M} + \sqrt{\nu^3\phi M}\right)^2 \\ &\le (1+\nu^{3/2})^2\phi M,\end{aligned}$$

provided that $w \ge 3C/(\phi\nu^3)$.

Finally, repeating $O(\log(n/\delta))$ times and taking the median and a union bound over all n columns gives the claimed result. □

Next we analyze the sampling algorithm, presented in Algorithm 2.3, which simulates sampling a column from A according to the column norms. The following theorem is our guarantee.

Theorem 2.3.4. *Let $\epsilon > 0$ be a constant small enough. With probability at least 0.9, Algorithm 2.3 outputs v_{j^*,i^*} which satisfies that, there exists $u \in [n]$ such that*

$$(1 - O(\epsilon))\left\|f(A_u)\right\|_2^2 \le \left\|f(v_{j^*,i^*})\right\|_2^2 \le (1 + O(\epsilon))\left\|f(A_u)\right\|_2^2.$$

Furthermore, there exists an absolute constant $c \in (0, 1/2]$ such that

$$\Pr\{u = i\} \ge c\frac{\|f(A_i)\|_2^2}{\|f(A)\|_F^2}$$

for all i belonging to some set $I \subseteq [n]$ such that $\sum_{i\in I}\|f(A_i)\|_2^2 \ge (1-6\epsilon)M$, provided that ϵ further satisfies that $\epsilon \le c/C$ for some absolute constant $C > 0$.

Proof. The analysis of the algorithm is largely classical, for which we define the following notions:

(1) $T_j = \zeta M/2^j$;

(2) $S_j = \{i \in [n] : \|f(A_i)\|_2^2 \in (T_j, 2T_j]\}$ is the j-th level set of A;

(3) a level $j \in [L]$ is *important* if $|S_j| \geq \epsilon 2^j/L$;

(4) $\mathcal{J} \subseteq [L]$ is the set of all important levels.

It follows from the argument in [51], or an argument similar to [3] that the columns we miss contribute to only an $O(\epsilon)$-fraction of the norm, and for each level $j \in \{1, \ldots, j_0\} \cup \mathcal{J}$, each of the recovered columns λ_i ($i \in [s_j]$) corresponds to some $u = u(i) \in S_j$ and satisfies that $(1 - O(\epsilon))\|f(A_{u_i})\|_2^2 \leq \lambda_i \leq (1 + O(\epsilon))\|f(A_{u_i})\|_2^2$.

Next we prove the second part. For a fixed $i \in [n]$, define events

$$\mathcal{E}_i = \{i \text{ falls in a level } j \in [j_0] \cup \mathcal{J}\}$$

and a set of "good" columns

$$I = \{i : \Pr\{\mathcal{E}_i\} \geq \beta\}$$

for some constant $\beta \leq 1/2$. Since all non-important levels always contribute to at most a 2ϵ-fraction of M, it follows that the bad columns contribute to at most a $2\epsilon/(1-\beta)$-fraction of M, that is,

$$\sum_{i \notin I} \|f(A_i)\|_2^2 \leq \frac{2\epsilon}{1-\beta} \cdot M.$$

Next we define the event that

$$\mathcal{F}_i = \{\mathcal{E}_i \text{ and } \|f(A_i)\|_2^2 \in [(1+\epsilon)T_j, 2(1-\epsilon)T_j]\},$$

then it holds for all $i \in I$ that $\Pr\{\mathcal{F}_i\} \geq \Pr\{\mathcal{E}_i\} - O(\epsilon) \geq 0.9\beta$ for ϵ sufficiently small.

Let $\mathcal{G}_j$ denote the event that the magnitude level j is chosen, and $j(i)$ is the index of the magnitude level containing column i. Then for those i's with $j = j(i) \in [j_0] \cup \mathcal{J}$,

$$\Pr\{\mathcal{G}_j | \mathcal{F}_i\} = \frac{(1 \pm O(\epsilon))\widetilde{M}_j}{(1 \pm O(\epsilon))\sum_j \widetilde{M}_j} = \frac{(1 \pm O(\epsilon))M_j}{(1 \pm O(\epsilon))M} = (1 \pm O(\epsilon))\frac{M_j}{M}$$

and

$$\Pr\{u = i \mid \mathcal{G}_j \cap \mathcal{F}_i\} = \frac{\lambda_t^{(j)}}{\widetilde{M}_j} = \frac{1 \pm O(\epsilon)\|f(A_i)\|_2^2}{(1 \pm O(\epsilon))M_j} = (1 \pm O(\epsilon))\frac{\|f(A_i)\|_2^2}{M_j}$$

Hence

$$\begin{aligned}\Pr\{u = i\} &= \Pr\{u = i \mid \mathcal{G}_j \cap \mathcal{F}_i\} \Pr\{\mathcal{G}_j | \mathcal{F}_i\} \Pr\{\mathcal{F}_i\} \\ &\geq 0.9\beta(1 - O(\epsilon))\frac{\|f(A_i)\|_2^2}{M} \geq 0.8\beta\frac{\|f(A_i)\|_2^2}{M},\end{aligned}$$

provided that ϵ is sufficiently small. □

Now we show how to obtain an overestimate $\widehat{M}$ for M. We assume that all entries of A are integer multiples of $\eta = 1/\operatorname{poly}(n)$ and are bounded by $\operatorname{poly}(n)$, which is a common and necessary assumption for streaming algorithms, otherwise storing a single number would take too much space. Let $\tilde{f}(x) = \log^2(1 + |\eta x|)$, then $\|f(A)\|_F^2 = \sum_{i,j} \tilde{f}(\eta^{-1}A)$, where $\eta^{-1}A$ has integer entries. Hence, we can run the algorithm implied by Theorem 2 of [8] on $\eta^{-1}A$ in parallel in order to obtain a constant-factor estimate to $\|f(A)\|_F^2$. To justify this application of the theorem, we verify in Section 2.5 that the function $\tilde{f}(|x|)$ is slow-jumping, slow-dropping and predictable on nonnegative integers as defined by [8].

Finally, we calculate the sketch length. The overall sketch length is dominated by that of Algorithm 2.3. In Algorithm 2.3, there are $\hat{L} = O(\epsilon^{-1}\log n)$ heavy hitter structures $\mathcal{D}_1, \ldots, \mathcal{D}_{\hat{L}}$, each of which has a sketch length of $O(1/(\phi^2\nu^3)\log(nL)) = \operatorname{poly}(L, 1/\epsilon, \log n) = \operatorname{poly}(\log n, 1/\epsilon)$. There is an additional heavy hitter structure $\mathcal{D}_0$ of sketch length $O(\operatorname{poly}(K, L, 1/\epsilon, \log n)) = \operatorname{poly}(\log n, 1/\epsilon)$. Hence the overall sketch length is $\operatorname{poly}(\log n, 1/\epsilon)$. Each cell of the sketch stores an n-dimensional vector. We summarize this in the following theorem.

Theorem 2.3.5. *Suppose that $A \in (\eta\mathbb{Z})^{n\times n}$ with $|A_{ij}| \le \operatorname{poly}(n)$ is given in a turnstile stream, where $\eta = 1/\operatorname{poly}(n)$. There exists a randomized sketching algorithm which maintains a sketch of $n\operatorname{poly}(\epsilon^{-1}\log n)$ space and outputs a vector $v_{j^*,i^*} \in \mathbb{R}^n$ which satisfies the same guarantee as given in Theorem 2.3.4.*

2.3.2 Low-Rank Approximation

Suppose that we have an approximate sampling of the rows of $f(A)$ so that we obtain a sample $f(A_i) + E_i$ with probability p_i satisfying

$$p_i \ge c\frac{\|f(A_i)\|_2^2}{\|f(A)\|_F^2} \tag{2.1}$$

for some absolute constant $c \le 1$. The p_i's are known to us (if $c = 1$, then we do not need to know the p_i).

The following is our main theorem in this section, which is analogous to Theorem 2 of [30].

Theorem 2.3.6. *Let V denote the subspace spanned by s samples drawn independently according to the distribution (2.1), where each sample has the form $f(A_i) + E_i$ for some $i \in [n]$. Suppose that $\|E_i\|_2 \le \gamma\|f(A_i)\|_2$ for some $\gamma > 0$. Then with probability at least $9/10$, there exists an orthonormal set of vectors $y_1, y_2, \ldots, y_k$ in V such that*

$$\left\| f(A) - f(A)\sum_{j=1}^{k} y_j y_j^\top \right\|_F^2 \le \min_{D:\operatorname{rank}(D)\le k} \|f(A) - D\|_F^2 + \frac{10k}{sc}(1+\gamma)^2\|f(A)\|_F^2 .$$

The theorem shows that the subspace spanned by a sample of columns chosen according to (2.1) contains an approximation to $f(A)$ that is nearly the best possible. Note that if the top k right singular vectors of S belong to this subspace, then $f(A)\sum_{t=1}^{k} v_t v_t^\top$ would provide the required approximation to $f(A)$ and we would be done.

Proof. For notational convenience, let $G = f(A)$. Let S be a random sample of s rows chosen from a distribution that satisfies (2.1). We can write the i-th sample as $G_i + E_i$ for some error vector E_i. Consider the singular value decomposition of $G = \sum_t \sigma_t u_t v_t^\top$.

For each t, we define a random vector

$$w_t = \frac{1}{s}\sum_{i\in S}\frac{(u_t)_i}{p_i}(G_i + E_i).$$

Note that S in general consists of sampled columns of $f(A)$ with noise. The vectors w_t are clearly in the subspace generated by S. We first compute $\mathbb{E}\, w_t$. We can view w_t as the average of s i.i.d. random variables $X_1, \ldots, X_s$, where each X_j has the following distribution:

$$X_j = \frac{(u_t)_i}{p_i}(G_i + E_i) \text{ with probability } p_i, \quad i = 1, 2, \ldots n.$$

Taking expectations,

$$\mathbb{E}\, X_j = \sum_{i=1}^{n}\frac{(u_t)_i}{p_i}(G_i + E_i)p_i = u_t^\top(G + E) = \sigma_t v_t^\top + u_t^\top E$$

Hence

$$\mathbb{E}\, w_t = \mathbb{E}\, X_j = \sigma_t v_t^\top + u_t^\top E$$

and

$$\|\mathbb{E}\, X_j\|_2^2 = \sigma_t^2 + 2\langle \sigma_t v_t^\top, u_t^\top E\rangle + \left\|u_t^\top E\right\|_2^2 \le \sigma_t^2 + 2\langle \sigma_t v_t^\top, u_t^\top E\rangle + \|E\|_2^2 .$$

We also calculate that

$$\begin{aligned}
\mathbb{E}\,\|X_j\|_2^2 &= \sum_i \frac{(u_t)_i^2}{p_i^2}\,\|G_i + E_i\|_2^2 \cdot p_i \\
&\le \sum_i \frac{(u_t)_i^2}{p_i}(\|G_i\|_2 + \|E_i\|_2)^2 \\
&\le \sum_i (u_t)_i^2 \frac{\|G\|_F^2}{c\,\|G_i\|_2^2}(1+\gamma)^2\,\|G_i\|_2^2 \\
&= \frac{(1+\gamma)^2}{c}\,\|G\|_F^2 ,
\end{aligned}$$

where we used the assumption (2.1) in the third line and the fact that $\|u_t\|_2 = 1$ in the last line. It follows that

$$\begin{aligned}
\mathbb{E}\,\|w_t\|_2^2 = \mathbb{E}\left\|\frac{1}{s}\sum_j X_j\right\|_2^2 &= \frac{1}{s}\sum_j \mathbb{E}\,\|X_j\|_2^2 + \frac{1}{s^2}\sum_{j\neq\ell}\langle \mathbb{E}\, X_j, \mathbb{E}\, X_\ell\rangle \\
&\le \frac{(1+\gamma)^2}{sc}\,\|G\|_F^2 + \frac{s(s-1)}{s^2}\left(\sigma_t^2 + 2\langle \sigma_t v_t^\top, u_t^\top E\rangle + \|E\|_2^2\right),
\end{aligned}$$

and thus

$$\begin{aligned}
\mathbb{E}\left\|w_t - \sigma_t v_t^\top\right\|_2^2 &= \mathbb{E}\,\|w_t\|_2^2 - 2\langle \mathbb{E}\, w_t, \sigma_t v_t^\top\rangle + \sigma_t^2 \\
&\le \frac{(1+\gamma)^2}{sc}\,\|G\|_F^2 + \sigma_t^2 + 2\langle \sigma_t v_t^\top, u_t^\top E\rangle + \|E\|_2^2 \\
&\quad - 2\sigma_t^2 - 2\langle u_t^\top E, \sigma_t v_t^\top\rangle + \sigma_t^2 \\
&= \frac{(1+\gamma)^2}{sc}\,\|G\|_F^2 .
\end{aligned} \tag{2.2}$$

If w_t were exactly equal to $\sigma_t v_t^\top$ (instead of just in expectation), we would have

$$G\sum_{t=1}^{k} v_t v_t^\top = G\sum_{t=1}^{k} w_t^\top w_t,$$

which would be sufficient to prove the theorem. We wish to carry this out approximately. To this end, define $\hat{y}_t = \frac{1}{\sigma_t} w_t^\top$ for $t = 1, 2, \ldots, s$ and let $V_1 = \text{span}(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_s) \subseteq V$. Let $y_1, y_2, \ldots, y_n$ be an orthonormal basis of $\mathbb{R}^n$ with $V_1 = \text{span}(y_1, y_2, \ldots, y_l)$, where $l = \dim(V_1)$. Let

$$B = \sum_{t=1}^{l} G y_t y_t^\top \quad \text{and} \quad \hat{B} = \sum_{t=1}^{k} G v_t \hat{y}_t^\top.$$

The matrix B will be our candidate approximation to G in the span of S. We shall bound its error using $\hat{B}$. Note that for any $i \leq k$ and $j > l$, we have $(\hat{y}_i)^\top y_j = 0$. Thus,

$$\begin{aligned}\|G - B\|_F^2 = \sum_{i=1}^{n} \left\|(G-B)y^{(i)}\right\|_2^2 = \sum_{i=l+1}^{n} \left\|Gy^{(i)}\right\|_2^2 \\ = \sum_{i=l+1}^{n} \left\|(G-\hat{B})y^{(i)}\right\|_2^2 \leq \left\|G - \hat{B}\right\|_F^2. \quad (2.3)\end{aligned}$$

Also,

$$\|G - \hat{B}\|_F^2 = \sum_{i=1}^{n} \left\|u_i^\top (G - \hat{B})\right\|_2^2 = \sum_{i=1}^{k} \left\|\sigma_i v_i^\top - w_i\right\|_2^2 + \sum_{i=k+1}^{n} \sigma_i^2$$

Taking expectations and using (2.2), we obtain that

$$\mathbb{E}\left\|G - \hat{B}\right\|_F^2 \leq \sum_{i=k+1}^{n} \sigma_i^2 + \frac{k(1+\gamma)^2}{sc} \|G\|_F^2 . \quad (2.4)$$

Note that $\hat{B}$ is of rank at most k and D_k is the best rank-k approximation to G. We have

$$\left\|G - \hat{B}\right\|_F^2 \geq \|G - D_k\|_F^2 = \sum_{i=k+1}^{n} \sigma_i^2$$

Thus $\|G - \hat{B}\|_F^2 - \|G - D_k\|_F^2$ is a non-negative random variable. It follows from (2.4) that

$$\Pr\left\{\left\|G - \hat{B}\right\|_F^2 - \|G - D_k\|_F^2 \geq \frac{10k(1+\gamma)^2}{sc} \|G\|_F^2\right\} \leq \frac{1}{10}.$$

The result follows from (2.3) and the fact that $\|E\|_F^2 \leq \gamma \|G\|_F^2$. □

Now, the difference between Theorem 2.3.4 and the assumption (2.1) is that we do not have control over p_i for an $O(\epsilon)$-fraction of the rows (in squared row norm contribution) in Theorem 2.3.4. Let A' be the submatrix of A after removing those rows, then $\|f(A)\|_F \leq (1 + O(\epsilon))\|f(A')\|_F$. We can apply Theorem 2.3.6 to A' and take more samples such that we obtain s rows from A' (which holds with $1 - \exp(-\Omega(s))$ probability by a Chernoff bound). We therefore have the following corollary.

Corollary 2.3.7. *Let y_i's be as in Algorithm 2.4 and c and ϵ be as in Theorem 2.3.4. It holds with probability at least 0.7 that*

$$\left\| f(A) - f(A)\sum_j y_j y_j^\top \right\|_F^2 \leq \min_{D:\mathrm{rank}(D)\leq k} \|f(A) - D\|_F^2 + \left(\frac{30k}{sc} + \epsilon\right)\|f(A)\|_F^2 .$$

Proof. First, it follows from a Chernoff bound and a union bound that we can guarantee with probability at least 0.9 that all samples have the form $f(A_i) + E_i$ with small $\|E_i\|_2$. Then, it follows from another Chernoff bound that with probability at least 0.9, it holds that there are $s/2$ samples from A'. We apply Theorem 2.3.6 to A' and $s/2$ and obtain that

$$\left\| f(A') - f(A')\sum_j y_j y_j^\top \right\|_F^2 \leq \min_{D:\mathrm{rank}(D)\leq k} \|f(A') - D\|_F^2 + \frac{30k}{sc}\|f(A')\|_F^2 .$$

Suppose that A'' is the submatrix of A which consists of the rows of A that are not in A'. Then $f(A)$ is the (interlacing) concatenation of $f(A')$ and $f(A'')$. Since $\|f(A'')\|_F^2 \leq \epsilon\|f(A)\|_F^2$ and $y_1, \ldots, y_k$ remains valid if we add more samples,

$$\begin{aligned}
&\left\| f(A) - f(A)\sum_j y_j y_j^\top \right\|_F^2 \\
&= \left\| f(A') - f(A')\sum_j y_j y_j^\top \right\|_F^2 + \left\| f(A'') - f(A'')\sum_j y_j y_j^\top \right\|_F^2 \\
&\leq \min_{D:\mathrm{rank}(D)\leq k} \|f(A') - D\|_F^2 + \frac{30k}{sc}\|f(A)\|_F^2 + \|f(A'')\|_F^2 \\
&\leq \min_{D:\mathrm{rank}(D)\leq k} \|f(A) - D\|_F^2 + \left(\frac{30k}{sc} + \epsilon\right)\|f(A)\|_F^2 .
\end{aligned}$$

The overall failure probability combines that of Theorem 2.3.4, Theorem 2.3.6 and the events at the beginning of this proof.

For the second result, take $s = O(k/\epsilon)$ and rescale ϵ. □

Note that Algorithm 2.3 can be easily modified to return the sampling probability of the sampled column, which is just $\lambda_i^{(j)} / \sum_j \widetilde{M}_j$. However, for each sample, we may lose control of it with a small constant probability. To overcome this, inspecting the proof of H-Sketch, we see that for fixed stream downsampling and fixed ζ in Algorithm 2.3, repeating each heavy hitter structure $O(\log(L/\delta))$ times and taking the median of each $\widetilde{M}_j$ will lower the failure probability of estimating the contribution of each important level to $\delta/(L+1)$, allowing for a union bound over all levels. Hence, with probability at least $1-\delta$, we can guarantee that we obtain a $(1 \pm O(\epsilon))$-approximation to $\|(f(A))_u\|_2^2$ and thus a $(1 \pm O(\epsilon))$-approximation to $\|f(A)\|_F^2$. Hence the returned $\hat{p}_u$ is a $(1 \pm O(\epsilon))$-approximation to the true row-sampling probability $p_u = \|(f(A))_u\|_2^2 / \|f(A)\|_F^2$. Different runs of the sampling algorithm may produce different values of $\hat{p}_u$ for the same u but they are all $(1 \pm O(\epsilon))$-approximations to p_u. We can guarantee this for all our s samples by setting $\delta = O(1/s)$, which allows for a union bound over all s samples.

Therefore, at the cost of an extra $O(\log s)$ factor in space, we can assume that $\hat{p}_u = (1 \pm O(\epsilon))p_u$ for all s samples. The overall algorithm is presented in Algorithm 2.4.

The following main theorem follows from Corollary 2.3.7 and the argument in [30].

Algorithm 2.4 Rank-k Approximation using H-Sketch

Input: $A \in \mathbb{R}^{n\times n}$, rank parameter k, number of samples s

1: Initialize s parallel instances of (modified) Algorithm 2.3
2: Let $(h_1, \hat{p}_1), \ldots, (h_q, \hat{p}_q)$ be the returned vectors and the sampling probability from the s instances of (modified) Algorithm 2.3
3: $F \leftarrow$ concatenated matrix $\left(\frac{h_1}{\sqrt{s\hat{p}_1}} \quad \cdots \quad \frac{h_s}{s\hat{p}_s}\right)$
4: Compute the top k left singular vectors of F, forming $L \in \mathbb{R}^{n\times k}$
5: **return** L

Theorem 2.3.8. *Let $s = O(k/\epsilon)$ be the number of samples and $y_1, \ldots, y_k$ be the output of Algorithm 2.4. It holds with probability at least 0.7 that*

$$\left\|f(A) - LL^\top f(A)\right\|_F^2 \le \min_{D:\mathrm{rank}(D)\le k} \|f(A) - D\|_F^2 + \epsilon \|f(A)\|_F^2 .$$

2.4 Application to Linear Regression

In this section, we consider approximately solving the linear regression problem using the H-Sketch from Section 2.3.1.

We shall need to sample rows from the concatenated matrix $Q = \begin{pmatrix} f(A) & b \end{pmatrix}$, where $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$ are given in a turnstile stream, and $f(x) = \ln(1+|x|)$ is the transformation function. This can still be achieved using the same H-Sketch in Section 2.3.1, applied to the concatenated matrix $\begin{pmatrix} A & b \end{pmatrix}$, with the transformation function $f(x)$ ($x \in \mathbb{R}^{d+1}$) replaced with $g(x) = \begin{pmatrix} f(x_{1:d}) & x_{d+1} \end{pmatrix}$, where $x_{1:d}$ denotes the first d coordinate of x and x_{d+1} the last coordinate of x. Then, using identical arguments in Section 2.3.1 for the squared ℓ_2-norm on the first d coordinates and a standard Count-Sketch argument for the last coordinate, it is straightforward to show an analogous version of Theorem 2.3.4 as below. We omit the identical proof.

Theorem 2.4.1. *Let $\epsilon > 0$, and let H_{j_0,i_0} be the vector returned by Algorithm 2.3. With probability at least 0.9, it holds that there exists $u \in [n]$ such that*

$$(1 - O(\epsilon)) \left(\|f(A_u)\|_2^2 + |b_u|^2\right) \le \|g(H_{j_0,i_0})\|_2^2 \le (1 + O(\epsilon)) \left(\|f(A_u)\|_2^2 + |b_u|^2\right).$$

Theorem 2.4.1 states that each sample is a noisy version of the u-th row of Q. Let $p_u = \|Q_u\|_2^2/\|Q\|_F^2$ be the true sampling probability of the u-th row. As argued at the beginning of Section 2.3.2, we may assume, at the cost of an $O(\log s)$ factor in space, that every sample is good, i.e., the returned sampling probability $\hat{p}_u$ satisfies that $\hat{p}_u = (1 \pm O(\epsilon))p_u$ and the noise in each sample is at most an $O(\epsilon)$-fraction in its ℓ_2 norm.

Below we present our algorithm for linear regression in Algorithm 2.5, assuming that every sample is good in the sense that $\hat{p}_u = (1 \pm O(\epsilon))p_u$ and the noise in each sample is at most an $O(\epsilon)$-fraction in ℓ_2-norm. The guarantee is given in Theorem 2.4.2.

Theorem 2.4.2. *Given matrix $A \in \mathbb{R}^{n\times d}$ and vector $b \in \mathbb{R}^n$, let $\kappa = \kappa(f(A))$ be the condition number of the transformed matrix. Let $s = O(\frac{d\kappa^2}{\epsilon^2} \log \frac{d}{\delta})$, then Algorithm 2.5 outputs a vector $\tilde{x} \in \mathbb{R}^d$, which, with probability at least $1 - \delta$, satisfies*

$$\left\|\tilde{M}\tilde{x} - \tilde{b}\right\|_2 \le (1 + \epsilon) \min_{x\in\mathbb{R}^d} \|f(A)x - b\|_2 + \Delta,$$

Algorithm 2.5 Linear Regression using H-Sketch

Require: $A \in \mathbb{R}^{n\times d}$, number of samples s

1: Initialize s parallel instances of Algorithm 2.3
2: Run Algorithm 2.3 on the concatenated matrix $\begin{pmatrix} f(A) & b \end{pmatrix}$ to obtain s row samples $h_1, \ldots, h_s$ and the corresponding sampling probabilities $\hat{p}_1, \ldots, \hat{p}_s$
3: $T \leftarrow$ vertical concatenation of $\frac{h_1}{\sqrt{s\hat{p}_1}}, \ldots, \frac{h_s}{\sqrt{s\hat{p}_s}}$
4: $\tilde{M} \leftarrow$ first d columns of T
5: $\tilde{b} \leftarrow$ last column of T
6: $\tilde{x} \leftarrow \arg\min_{x\in\mathbb{R}^d} \|\tilde{M}x - \tilde{b}\|_2$

where

$$\Delta = \epsilon\left(\sqrt{d + \frac{\|b\|_2^2}{\|f(A)\|_2^2}}\kappa\,\|b\|_2 + \sqrt{\|f(A)\|_F^2 + \|b\|_2^2}\right).$$

The total space used by Algorithm 2.5 is $\tilde{O}(d^2\kappa^2\log\frac{1}{\delta})\cdot\mathrm{poly}(\log n, \frac{1}{\epsilon})$.

By Theorem 2.3.4, for every $i \in [s]$, there exists $j(i)$ such that $h_i = (f(A)_{j(i)}, b_{j(i)}) + F_{j(i)}$, where $F_i = E_i/\sqrt{sp_i}$. We define a new matrix S such that in the i-th row of S, $S_{i,j(i)} = 1/\sqrt{sp_{j(i)}}$ and the other entries are zero. By Theorem 2.4.1, we have that the row-sampling probability we use is a $(1 \pm O(\epsilon))$ approximation to the true sampling probability. Therefore, we define matrix $\hat{S}$ such that in the i-th row of $\hat{S}$, $\hat{S}_{i,j(i)} = 1/\sqrt{s\hat{p}_{j(i)}}$ and the other entries are zero, and matrix $\hat{F}$ is such that $\hat{F}_i = E_i/\sqrt{s\hat{p}_i}$. Then, we find that $\hat{S}\begin{pmatrix} f(A) & b \end{pmatrix} + \hat{F} = T$.

Proof. For notational convenience, we let $G = f(A)$ with singular value decomposition $G = U\Sigma V^\top$. We shall show that $\left\|I_d - (\hat{S}U)^\top(\hat{S}U)\right\|_2$ is small, for which we first show $\left\|I_d - (SU)^\top(SU)\right\|_2$ is small.

Let $X_i = I_d - Y_i^\top Y_i$ and $Y_i = \frac{U_{j(i)}}{\sqrt{p_{j(i)}}}$, where U_t is the t-th row of U, which means that the $j(i)$-th row of M is chosen in the i-th trial. Since

$$\mathbb{E}(X_i) = I_d - \mathbb{E}(Y_i^\top Y_i) = I_d - \sum_{t=1}^n p_t \frac{U_t^\top}{\sqrt{p_t}}\frac{U_t}{\sqrt{p_t}} = I_d - \sum_{t=1}^n U_t^\top U_t = 0,$$

we can apply Lemma 1.2.2 to $X_1, \ldots, X_s$, for which we need to upper bound $\|X_i\|_2$ and $\|\mathbb{E}(X_i^2)\|_2$.

We first bound $\|X_i\|_2$.

$$\begin{aligned}
\|X_i\|_2 = \left\|I_d - Y_i^\top Y_i\right\|_2 &\le 1 + \frac{\left\|U_i^\top U_i\right\|_2}{p_i} \\
&\le 1 + \frac{\|U_i\|_2^2}{c\,\|G_i\|_2^2}\,\|G\|_F^2 \\
&\le 1 + \frac{\sigma_1^2 + \cdots + \sigma_d^2}{c\sigma_d^2} \\
&\le 1 + \frac{d\kappa^2}{c},
\end{aligned}$$

where $\sigma_1 \geq \cdots \geq \sigma_d$ are the singular values of G, and in the penultimate inequality we use the fact that $\|G_i\|_2 = \left\|U_i \Sigma V^\top\right\|_2 = \|U_i \Sigma\|_2 \geq \sigma_d \|U_i\|_2$.

Next, we bound $\|\mathbb{E}(X_i^2)\|_2$. Observe that

$$\begin{aligned}\mathbb{E}(X_i^2 + I_d) &= I_d + \mathbb{E}(I_d - Y_i^\top Y_i)(I_d - Y_i^\top Y_i) = I_d + \mathbb{E}(I_d - 2Y_i^\top Y_i + Y_i^\top Y_i Y_i^\top Y_i) \\ &= 2I_d - \mathbb{E}(Y_i^\top Y_i) + \mathbb{E}(Y_i^\top Y_i \|Y_i\|_2^2) = \mathbb{E}\left(\frac{\left\|U_{j(i)}\right\|_2^2}{p_{j(i)}} Y_i^\top Y_i\right),\end{aligned}$$

and thus

$$\begin{aligned}\left\|\mathbb{E}(X_i^2 + I_d)\right\|_2 &= \left\|\mathbb{E}\left(\frac{\left\|U_{j(i)}\right\|_2^2}{p_{j(i)}} Y_i^\top Y_i\right)\right\|_2 \\ &\leq \left\|\mathbb{E}\left(\frac{\|U_i\|_2^2}{c\|G_i\|_2^2} \|G\|_F^2 Y_i^\top Y_i\right)\right\|_2 \\ &\leq \left\|\mathbb{E}\left(\frac{d\kappa^2}{c} Y_i^\top Y_i\right)\right\|_2 \\ &= \frac{d\kappa^2}{c}.\end{aligned}$$

It follows immediately from the triangle inequality that

$$\left\|\mathbb{E}\, X_i^2\right\|_2 \leq \left\|\mathbb{E}(X_i^2 + I_d)\right\|_2 + \|I_d\|_2 \leq \frac{d\kappa^2}{c} + 1.$$

Invoking Lemma 1.2.2, for

$$W = \frac{1}{s}\sum_{i=1}^s X_i = I_d - \frac{1}{s}\sum_{i=1}^s Y_i^\top Y_i = I_d - (SU)^\top (SU),$$

and $\rho = \sigma^2 = 1 + d\kappa^2/c$, we have that

$$\Pr\left\{\left\|I_d - (SU)^\top (SU)\right\|_2 > \epsilon\right\} \leq 2d \exp\left(-\frac{\epsilon^2 s}{\sigma^2 + \rho\epsilon/3}\right) \leq 2d \exp\left(-\frac{\epsilon^2 s}{2d\kappa^2/c}\right) \leq \delta$$

by our choice of s. Equivalently, it holds that $\Pr\{\left\|I_d - (SU)^\top (SU)\right\|_2 \leq \epsilon\} \geq 1 - \delta$, which implies that $\|SGx\|_2 = (1 \pm \epsilon)\|Gx\|_2$ for all $x \in \mathbb{R}^d$. We condition on this event in the rest of the proof.

Second, we show the error between $\left\|I_d - (SU)^\top (SU)\right\|_2$ and $\left\|I_d - (\hat{S}U)^\top (\hat{S}U)\right\|_2$ is small.

$$\begin{aligned}\left\|I_d - (\hat{S}U)^\top (\hat{S}U)\right\|_2 &\leq \left\|I_d - (SU)^\top (SU)\right\|_2 + \left\|(\hat{S}U)^\top (\hat{S}U) - (SU)^\top (SU)\right\|_2 \\ &\leq \epsilon + \left\|(\hat{S}U)^\top (\hat{S}U) - (SU)^\top (SU)\right\|_2.\end{aligned}$$

Observe that

$$(\hat{S}U)^\top (\hat{S}U) = \sum_{i=1}^s \frac{U_{j(i)}^\top U_{j(i)}}{s\hat{p}_{j(i)}} = \sum_{i=1}^s \frac{U_{j(i)}^\top U_{j(i)}}{(1 \pm O(\epsilon)) s p_{j(i)}} = \frac{(SU)^\top (SU)}{1 \pm O(\epsilon)}$$

and thus

$$\left\|(\hat{S}U)^\top(\hat{S}U) - (SU)^\top(SU)\right\|_2 = O(\epsilon)\left\|(SU)^\top(SU)\right\|_2.$$

We have proved that $\left\|I_d - (SU)^\top(SU)\right\|_2 \le \epsilon$, so we have $\left\|I_d - (\hat{S}U)^\top(\hat{S}U)\right\|_2 \le \epsilon + O(\epsilon)(1+\epsilon) = O(\epsilon)$. By rescaling ϵ', we can assume that $\left\|I_d - (\hat{S}U)^\top(\hat{S}U)\right\|_2 \le \epsilon$.

Now consider the subspace spanned by the columns of M together with b. For any vector $y = Gx - b$, $\left\|\hat{S}y\right\|_2 = (1\pm\epsilon)\left\|y\right\|_2$. Recall that we have defined $\hat{F}_i = E_i/\sqrt{s\hat{p}_i}$, where $\hat{F}_i$ and E_i are the corresponding i-th row of F and E. Let $\hat{F}^{(1)}$ be the first d columns of $\hat{F}$ and $\hat{F}^{(2)}$ be the last column of $\hat{F}$. Hence, the original linear regression problem can be written as $\min\left\|(\hat{S}G + \hat{F}^{(1)})x - (\hat{S}b + \hat{F}^{(2)})\right\|_2$.

Note that $\tilde{x} = \arg\min_x \left\|(\hat{S}G + \hat{F}^{(1)})x - (\hat{S}b + \hat{F}^{(2)})\right\|_2$ satisfies

$$\begin{aligned}
\min_{\tilde{x}}\left\|(\hat{S}G + \hat{F}^{(1)})\tilde{x} - (\hat{S}b + \hat{F}^{(2)})\right\|_2 &\le \left\|(\hat{S}G + \hat{F}^{(1)})x^* - (\hat{S}b + \hat{F}^{(2)})\right\|_2 \\
&\le \left\|\hat{S}(Gx^* - b)\right\|_2 + \left\|\hat{F}^{(1)}x^* - \hat{F}^{(2)}\right\|_2 \\
&\le (1+\epsilon)\left\|Gx^* - b\right\|_2 + \left\|\hat{F}\right\|_2\sqrt{\left\|x^*\right\|_2^2 + 1},
\end{aligned}$$

where the third inequality holds because $\hat{S}$ is a subspace embedding for the column space of G together with b and $x^* = \arg\min_{x\in\mathbb{R}^d}\left\|Gx - b\right\|_2$.

Now, consider the upper bound on $\left\|\hat{F}\right\|_2$. Since

$$\left\|\hat{F}_i\right\|_2^2 = \frac{\left\|E_i\right\|_2^2}{s\hat{p}_i} \le \gamma^2\frac{\left\|G_i\right\|_2^2 + |b_i|^2}{sc(\left\|G_i\right\|_2^2 + |b_i|^2)}(\left\|G\right\|_F^2 + \left\|b\right\|_2^2) \le \frac{\gamma^2}{sc}(\left\|G\right\|_F^2 + \left\|b\right\|_2^2)$$

and

$$\left\|x^*\right\|_2 = \left\|G^\dagger b\right\|_2 \le \frac{\left\|b\right\|_2}{\sigma_{\min}(G)},$$

we have that

$$\begin{aligned}
\min_{\tilde{x}}\left\|(\hat{S}G + \hat{F}^{(1)})\tilde{x} - (\hat{S}b + \hat{F}^{(2)})\right\|_2 &\le (1+\epsilon)\left\|Gx^* - b\right\|_2 + \left\|\hat{F}\right\|_2\sqrt{\left\|x^*\right\|_2^2 + 1} \\
&\le (1+\epsilon)\left\|Gx^* - b\right\|_2 + \frac{\gamma}{\sqrt{c}}\sqrt{\left\|G\right\|_F^2 + \left\|b\right\|_2^2}\cdot\sqrt{\frac{\left\|b\right\|_2^2}{\sigma_{\min}^2(G)} + 1} \\
&\le (1+\epsilon)\left\|Gx^* - b\right\|_2 + \frac{\gamma}{\sqrt{c}}\left(\sqrt{\left\|G\right\|_F^2 + \left\|b\right\|_2^2} + \sqrt{d + \frac{\left\|b\right\|_2^2}{\left\|G\right\|_2^2}}\kappa\left\|b\right\|_2\right).
\end{aligned}$$

By our assumption, $c = 1 - O(\epsilon)$ and $\gamma = O(\epsilon)$. Rescaling ϵ gives the claimed bound, completing the proof of Theorem 2.4.2. □

2.5 Obtaining an Overestimate $\widehat{M}$

In this section we verify that $g(x) = \ln^2(1+\eta x)$ is slow-jumping, slow-dropping, and predictable, where the three properties are defined in [8].

For completeness, we reproduce the definitions of these properties from [8]. We shall only consider functions $f(x)$ that satisfies $f(x) > 0$ for all $x > 0$ and $f(0) = 0$.

- Slow-jumping: for every $\alpha > 0$, there exists Y such that for all $y > Y$, it holds for all $x < y$ that $f(y) \leq \lfloor \frac{y}{x} \rfloor^{2+\alpha} x^\alpha g(x)$.
- Slow-dropping: for every $\alpha > 0$, there exists Y such that for all $y > Y$, it holds for all $x < y$ that $g(y) \geq g(x)/y^\alpha$.
- Predictable: for any $\gamma \in (0, 1)$ and subpolynomial $\epsilon(x)$, there exist X such that for all $x \geq X$ and for all $y \in [1, x^{1-\gamma}]$ such that $|f(x+y) - f(x)| > \epsilon f(x)$, it holds that $f(y) \geq x^{-\gamma} f(x)$.

We first show that our g is slow-jumping. Let $\alpha > 0$. (i) When $x \geq y/2$, it suffices to show that $g(y) \leq x^\alpha g(x)$. Since $g(x)$ is increasing, it reduces to showing $g(y) \leq (y/2)^\alpha g(y/2)$. This clearly holds for all large y because one can easily check that $\ln(1+y) \leq 2\ln(1+\frac{y}{2})$ when $y > 0$. (ii) When $x < y/2$, we shall show that $g(y) \leq (\frac{y}{x} - 1)^{2+\alpha} x^\alpha g(x)$, i.e., $g(y) \leq (\frac{y-x}{x})^2 (y-x)^\alpha g(x)$. Since $x < y/2$, we have $y - x \geq y/2$ and thus it suffices to show that $g(y) \leq \frac{1}{4}(\frac{y}{x})^2 (\frac{y}{2})^\alpha g(x)$, and for large y that $\frac{g(y)}{y^2} \leq \frac{g(x)}{x^2}$, which can be easily verified. This concludes the proof that g is slow-jumping.

It is obvious that g is slow-dropping because it is increasing.

Last, we show that g is predictable. Note that $g(2x)/g(x) \to 1$ as $x \to \infty$. Thus, for any given $\epsilon(x)$, when x is sufficiently large, it would not hold that $g(x+y) > (1+\epsilon(x))g(x)$ for $y \in [1, x]$.

Chapter 3

Online Active Regression

3.1 Introduction

Linear regression is a simple method to model the relationship between the data points in a Euclidean space and their scalar labels. A typical formulation is to solve the minimization problem $\min_x \|Ax-b\|_p$ for $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$, where each row A_i is a data point in $\mathbb{R}^d$ and b_i is its corresponding scalar label. When $p = 2$, the linear regression is precisely the least-squares regression, which admits a closed-form solution and is thus a classical choice due to its computational simplicity. When $p \in [1, 2)$, it is more robust than least-squares as the solution is less sensitive to outliers. A popular choice is $p = 1$ because the regression can be cast as a linear programme, although other values of p are recommended depending on the distribution of noise in the labels. Interested readers may refer to Section 1.3 of [34] for a discussion.

One harder variant of linear regression is *active regression* [66], in which the data points are easy to obtain but the labels are costly. Here one can query the label of any chosen data point and the task is to minimize the number of queries while still being able to solve the linear regression problem approximately. Specifically, one constructs an index set $S \subset [n]$ as small as possible, queries b_S (the restriction of b on S) and computes a solution $\tilde{x}$ based on A, S and b_S such that

$$\|A\tilde{x} - b\|_p^p \leq (1+\epsilon) \min_{x\in\mathbb{R}^d} \|Ax - b\|_p^p. \tag{3.1}$$

For $p = 2$, the classical approach is to sample the rows of A according to the leverage scores. This can achieve (3.1) with large constant probability using $|S| = O(d \log d + d/\epsilon)$ queries. Chen and Price [13] reduced the query complexity to the optimal $O(d/\epsilon)$, based on graph sparsifiers. When $p = 1$, Chen and Derezinski [12] and Parulekar et al. [63] showed that $O((d \log d)/\epsilon^2)$ queries suffice with large constant probability, based on sampling according to Lewis weights. More recently, Musco et al. [59] solved the problem for all values of p with query complexity $\tilde{O}(d/\operatorname{poly}(\epsilon))$ for $p < 2$ and $\tilde{O}(d^{p/2}/\operatorname{poly}(\epsilon))$ for $p > 2$, where the dependence on d is optimal up to logarithmic factors.

Another common setting of linear regression is the *online setting*, which considers memory restrictions that prohibit storing the inputs A and b in their entirety. In such a case, each pair of data points and their labels (i.e. each row of $[A\ b]$) arrives one by one, and the goal is to use as little space as possible to solve the linear regression problem. Again, the case of $p = 2$ has the richest research history, with the state-of-the-art results due to [18] and [40], which retain only $O(\epsilon^{-1} d \log d \log(\epsilon\|A\|_2^2))$ rows of A (where $\|A\|_2$

denotes the operator norm of A). The idea of the algorithms is to sample according to online leverage scores, which was first used by Kapralov et al. [45]. The online leverage score of a row is simply the leverage score of the row in the submatrix of A consisting of all the revealed rows so far. The algorithm of [40] is based on that of [18] with further optimized runtime. The case of $p = 1$ was solved by Braverman et al. [9], who generalized the notion of online leverage scores to online Lewis weights and sampled the rows of A according to the online Lewis weights.

In this chapter, we consider the problem of *online active regression*, a combination of the two variants above. In a similar vein to [18] and [40], the rows of A arrive one by one, and upon receiving a row, one must decide whether it should be kept or discarded and whether to query the corresponding label, without ever retracting these decisions. The problem was considered by [65], who assumed an underlying distribution of the data points together with a noise model of the labels and only considered ℓ_2-regression. Here we do not make such assumptions and need to handle arbitrary input data. To the best of our knowledge, our work is the first to consider the online active regression in the general ℓ_p-norm. Our approach is largely based upon existing techniques for online regression and active regression. A technical contribution of our work is to show that one can compress a fraction of rows in a matrix by sampling these rows according to their Lewis weights while preserving the Lewis weights of the uncompressed rows (see Lemma 3.4.3 for the precise statement), which may be of independent interest.

Our Results. We show that the online active regression problem can be solved, attaining the error guarantee (3.1) using $m = O(\epsilon^{-2} d \log(d/\epsilon) \log n \log \kappa^{\mathrm{OL}}(A))$ queries for $p = 1$, $m = O(\epsilon^{-1} d \operatorname{poly}(\log(d/\epsilon)) \cdot \log(n\kappa^{\mathrm{OL}}))$ queries for $p \in (1, 2)$ (where κ^{OL} is the online condition number of A) and $m = O(\epsilon^{-1} d \operatorname{poly}(\log(d/\epsilon)) \log(n\|A\|_2/\sigma))$ queries for $p = 2$, all with constant probability respectively (where $\|A\|_2$ is the operator norm of A and σ the smallest singular value of the first d rows of A). Our algorithms are sublinear in space complexity, using $O(md)$ words.

The query complexity for $p \in (1, 2)$ is only worse by a factor of $\log n\kappa^{\mathrm{OL}}$ than that of its offline counterpart [59], which is not unexpected, given that the same factor appears in the sketch size for the ℓ_1-subspace embedding under the sliding window model [9].

Follow-up Work. After our work was published in the Proceedings of ICML 2022, Woodruff and Yasuda [82] resolved the active regression problem for $p > 2$ by using one-sided Lewis weights instead of standard Lewis weights and maintaining the quadratic form $A^\top W^{1-2/p} A$ during the reading of matrix A instead of using the compression trick.

3.2 Preliminaries

Notation. For two matrices A and B of the same number of columns, we denote by $A \circ B$ the vertical concatenation of A and B.

Suppose that $A \in \mathbb{R}^{n \times d}$. The *online condition number* of A is defined as $\kappa^{\mathrm{OL}}(A) = \|A\|_2 \max_{S \subseteq [n], S \neq \emptyset} \|A_S^\dagger\|_2$, where A_S is the submatrix of A consisting of the rows with indices in S.

Suppose that $A \in \mathbb{R}^{n \times d}$, $b \in \mathbb{R}^d$ and $p \geq 1$. We define $\textsc{Reg}(A, b, p)$ to be an $x \in \mathbb{R}^d$ that minimizes $\|Ax - b\|_p$. We remark that when $p > 1$, the minimizer is unique.

All sampling matrices in this chapter are of the first kind, unless stated otherwise. Associated with a sampling matrix S are indicator variables $\{(\mathbb{1}_S)_i\}_{i=1,\dots,n}$ (where n is the number of columns of S) defined as follows. For each i, we define $(\mathbb{1}_S)_i = 1$ if the i-th column of S is nonzero, and $(\mathbb{1}_S)_i = 0$ otherwise.

Definition 3.2.1 (Online Lewis Weights)**.** Let $p \in [1, 2)$ and $A \in \mathbb{R}^{n\times d}$. The online ℓ_p Lewis weights, denoted by $w_1^{\mathrm{OL}}(A), \dots, w_n^{\mathrm{OL}}(A)$, are defined as $w_i^{\mathrm{OL}}(A) = w_i(A^{(i)})$, where $A^{(i)}$ is the submatrix consisting of the first i rows of A.

By Lemma 3.3.1, we see that $w_i^{\mathrm{OL}}(A) \geq w_i(A)$. Hence, if we sample the rows of A using online Lewis weights, i.e. replacing $w_i(A)$ with $w_i^{\mathrm{OL}}(A)$ in the construction of S as described in Lemma 1.2.12 (see the remark following the lemma), the resulting S still forms a $(1+\epsilon)$-subspace-embedding for A in the ℓ_p-norm.

Preservation of ℓ_2-Norm. We shall need the Johnson-Lindenstrauss matrix for the online active ℓ_2-regression.

Definition 3.2.2 (Johnson-Lindenstrauss Matrix)**.** Let $X \subseteq \mathbb{R}^d$ be a set of points. A matrix J is said to be a Johnson-Lindenstrauss matrix for X of distortion parameter ϵ (or, an ϵ-JL matrix for X) if $(1-\epsilon)\|x\|_2^2 \leq \|Jx\|_2^2 \leq (1+\epsilon)\|x\|_2^2$ for all $x \in X$.

It is a classical result [44] that when $|X| = T$, there exists a random matrix $J \in \mathbb{R}^{m\times d}$ with $m = O(\epsilon^{-2}\log(T/\delta))$ such that (i) J is an ϵ-JL matrix for X with probability at least $1-\delta$, (ii) each column of J contains $O(\epsilon^{-1}\log(T/\delta))$ nonzero entries and (iii) J can be generated using $O(\log^2(|T|/\delta)\log d)$ bits.

3.3 Additional Properties of Lewis Weights

A useful result of Lewis weights is their monotonicity when $p \leq 2$ [16, Lemma 5.5], which we state formally below.

Lemma 3.3.1 (Monotonicity of Lewis weights)**.** *Suppose that $p \in [1, 2]$, $A \in \mathbb{R}^{n\times d}$ and $r \in \mathbb{R}^d$. It holds for all $i = 1, \dots, n$ that $w_i(A) \geq w_i(A \circ r^\top)$.*

The next two lemmata are helpful in proving approximations to Lewis weights.

Lemma 3.3.2. *Given $A \in \mathbb{R}^{n\times d}$ with ℓ_p-Lewis weights w_i, $i = 1, \dots, n$. Let S be the rescaled sampling matrix of the first kind with respect to $p_1, \dots, p_n$ satisfying that $\min\{\beta w_i, 1\} \leq p_i \leq 1$, where $\beta = \Omega(\epsilon^{-2}\log(d/\delta))$. With probability at least $1-\delta$, it holds that*

$$(1-\epsilon)\sum_{i=1}^n w_i^{1-\frac{2}{p}} a_i a_i^\top \preceq \sum_{i=1}^n \frac{(\mathbb{1}_S)_i}{p_i} w_i^{1-\frac{2}{p}} a_i a_i^\top \preceq (1+\epsilon)\sum_{i=1}^n w_i^{1-\frac{2}{p}} a_i a_i^\top.$$

Proof. We prove the lemma by matrix Bernstein inequality. Without loss of generality, we assume that $p_i \leq 1/\beta$ for all i, otherwise we can restrict the sum to the i's such that $p_i \leq 1/\beta$. We further assume that $A^\top W^{1-\frac{2}{p}} A = I_d$, where $W = \mathrm{diag}\{w_1, \dots, w_n\}$. Let $X_i = \frac{(\mathbb{1}_S)_i}{p_i} \cdot w_i^{1-\frac{2}{p}} a_i a_i^\top - w_i^{1-\frac{2}{p}} a_i a_i^\top$, then $\mathbb{E}\, X_i = 0$. By the definition of Lewis weights, we

have $w_i^{\frac{2}{p}} = a_i^\top (A^\top W^{1-\frac{2}{p}} A)^{-1} a_i$. Hence, we have $\|a_i\|_2^2 = w_i^{\frac{2}{p}}$. Next, $\|X_i\|_2 \le \frac{w_i^{1-\frac{2}{p}}}{p_i}\|a_i\|_2^2 = \frac{w_i}{p_i} \le \frac{1}{\beta}$ and

$$\left\|\mathbb{E}\sum_{i=1}^n X_i X_i^\top\right\|_2 = \left\|\mathbb{E}\sum_{i=1}^n (\frac{1}{p_i}-1) w_i^{2(1-\frac{2}{p})} \|a_i\|_2^2 a_i a_i^\top\right\|_2 \le \left\|\sum_{i=1}^n w_i^{1-\frac{2}{p}} \cdot \frac{a_i a_i^\top}{\beta}\right\|_2 = \frac{1}{\beta}.$$

Applying the matrix Bernstein inequality (Lemma 1.2.2), we have

$$\Pr\left\{\left\|\sum_{i=1}^n X_i X_i^\top\right\|_2 \ge \epsilon\right\} \le 2d\exp\left(\frac{-\epsilon^2}{\frac{1}{\beta}+\frac{\epsilon}{3\beta}}\right) = 2d\exp\left(-\Omega(\beta\epsilon^2)\right) \le \delta. \qquad \square$$

Lemma 3.3.3. *Suppose that $p \in [1,2]$, $A \in \mathbb{R}^{n\times d}$ and $\overline{w_1}, \dots, \overline{w_n}$ are the Lewis weights of A. Let $w_1, \dots, w_n$ be weights such that*

$$\alpha w_i^{2/p} \le a_i^\top \left(\sum_i w_i^{1-2/p} a_i a_i^\top\right)^{-1} a_i \le \beta w_i^{2/p}, \quad \forall i = 1, \dots, n,$$

where $\beta > \alpha > 0$. Then it holds for all i that

$$\alpha w_i \le \overline{w_i} \le \beta w_i.$$

Proof. Let $\gamma = \inf\{c > 0 : w_i \ge c\overline{w_i} \text{ for all } i\}$. It then holds for all i that

$$\begin{aligned} w_i^{2/p} &\ge \frac{1}{\beta} a_i^\top \left(\sum_i w_i^{1-2/p} a_i a_i^\top\right)^{-1} a_i \\ &\ge \frac{1}{\beta} a_i^\top \left(\sum_i (\gamma\overline{w_i})^{1-2/p} a_i a_i^\top\right)^{-1} a_i \\ &= \frac{\gamma^{2/p-1}}{\beta} a_i^\top \left(\sum_i \overline{w_i}^{1-2/p} a_i a_i^\top\right)^{-1} a_i \\ &= \frac{\gamma^{2/p-1}}{\beta} \overline{w_i}^{2/p}. \end{aligned}$$

This implies that

$$\gamma^{2/p} \ge \frac{\gamma^{2/p-1}}{\beta},$$

and thus

$$\gamma \ge \frac{1}{\beta},$$

that is, $w_i \ge \overline{w_i}/\beta$ for all i. Similarly one can show that $w_i \le \overline{w_i}/\alpha$. $\square$

Next we shall prove an important property of Lewis weights of concatenated matrices. In this section, all rescaled sampling matrices are of the first kind.

Lemma 3.3.4. *Let $A_i \in \mathbb{R}^{n_i\times d}$ $(i = 1, \dots, r)$, $B \in \mathbb{R}^{k\times d}$ and $M = A_1 \circ A_2 \circ \cdots \circ A_r \circ B$. For each $i \in [r]$, let $S_i \in \mathbb{R}^{m_i\times n_i}$ be the rescaled sampling matrix of the first kind with respect to $p_{i,1}, \dots, p_{i,n_i}$ with $\min\{\beta w_j(A_i), 1\} \le p_{i,j} \le 1$ for each $j \in [n_i]$, where $\beta = \Omega(\epsilon^{-2}\log(d/\delta))$. Let $M' = S_1 A_1 \circ \cdots \circ S_r A_r \circ B$. The following statements hold with probability at least $1-\delta$.*

1. *For each $i \in [r]$ and each $j \in [m_i]$, it holds that*

$$(1-\epsilon)\frac{w_{n_1+\cdots+n_{i-1}+s_i(j)}(M)}{p_{i,s_i(j)}} \le w_{m_1+\cdots+m_{i-1}+j}(M') \le (1+\epsilon)\frac{w_{n_1+\cdots+n_{i-1}+s_i(j)}(M)}{p_{i,s_i(j)}},$$

where $s_i(j) \in [n_i]$ is the row index such that $(S_i)_{j,s(i,j)} \neq 0$.

2. *For each $j = 1, \ldots, k$, it holds that*

$$(1-\epsilon)w_{n_1+\cdots+n_r+j}(M) \le w_{m_1+\cdots+m_r+j}(M') \le (1+\epsilon)w_{n_1+\cdots+n_r+j}(M).$$

Proof. Define partial sums $\mu_i = m_1 + \cdots + m_i$ with $\mu_0 = 0$ and $\nu_i = n_1 + \cdots + n_i$ with $\nu_0 = 0$. For each $j \in [\mu_r + k]$,

$$w'_j = \begin{cases} w_{\nu_{i-1}+s_i(j)}(M)/p_{i,s_i(j)}, & \mu_{i-1} < j \le \mu_i; \\ w_{j-\mu_r+\nu_r}(M), & j \ge \mu_r. \end{cases}$$

and

$$L = \sum_{i=1}^{r}\sum_{j=1}^{m_i} \frac{(S_iA_i)_j(S_iA_i)_j^\top}{(w'_{\mu_{i-1}+j})^{p/2-1}} + \sum_{j=1}^{k}\frac{b_ib_i^\top}{(w'_{\mu_r+j})^{p/2-1}}.$$

Then we have

$$L = \sum_{i=1}^{r}\sum_{j=1}^{m_i} \frac{(A_i)_{s_i(j)}(A_i)_{s_i(j)}^\top}{p_{i,s_i(j)}(w_{s_i(j)}(A_i))^{p/2-1}} + \sum_{j=1}^{k}\frac{b_ib_i^\top}{(w_{\nu_r+j}(M))^{p/2-1}}.$$

Let $W_M = \mathrm{diag}\{w_1(M), \ldots, w_{\nu_r+k}(M)\}$. Let $p_i = 1$ for $i = \nu_r + 1, \ldots, \nu_r + k$. Also note that $p_{i,j} \ge \min\{\beta w_{\nu_{i-1}+j}(M), 1\}$ since $w_j(A_i) \ge w_{\nu_{i-1}+j}(M)$. It follows from Lemma 3.3.2 that

$$(1-\epsilon)(M^\top W_M^{1-2/p}M) \preceq L \preceq (1+\epsilon)(M^\top W_M^{1-2/p}M),$$

with probability at least $1-\delta$.

Next we verify that $\{w'_j\}_j$ are good weights for M'. When $\mu_{i-1} < j \le \mu_i$,

$$(w'_j)^{2/p} = \frac{(w_{\nu_{i-1}+s_i(j)}(M))^{2/p}}{p_{i,s_i(j)}^{2/p}} = \frac{(A_i)_{s_i(j)}(M^\top W_M^{1-2/p}M)^{-1}(A_i)_{s_i(j)}^\top}{p_{i,s(j)}^{2/p}}$$
$$= \frac{1}{1\pm\epsilon}\cdot\frac{(A_i)_{s_i(j)}L^{-1}(A_i)_{s_i(j)}^\top}{p_{i,s_i(j)}^{2/p}} = \frac{1}{1\pm\epsilon}(S_iA_i)_jL^{-1}(S_iA_i)_j^\top,$$

where $(S_iA_i)_j$ denotes the j-th row of S_iA_i. Similarly, one can show that for $j > \mu_r$,

$$(w'_{\mu_r+j})^{2/p} = \frac{1}{1\pm\epsilon}b_{j-\mu_r}L^{-1}b_{j-\mu_r}^\top.$$

The result follows from Lemma 3.3.3. This completes the proof of 3.3.4. □

3.4 Algorithms and Main Results

In this section, we shall demonstrate how to solve the online active regression with $\tilde{O}(\epsilon^{-2}d\log(n\kappa^{\mathrm{OL}}(A)))$ queries. We shall then improve the number of queries to $\tilde{O}(\epsilon^{-1}d\cdot\log(n\kappa^{\mathrm{OL}}(A)))$ in Section 3.6, attaining the optimal dependence on ϵ.

The high-level approach follows [59] and we give a brief review below. We sample A twice w.r.t. (online) Lewis weights but with different oversampling parameters β, getting SA of $O(d\log d)$ rows and S_1A of $O(d^2\,\mathrm{poly}(\epsilon^{-1}\log d))$ rows, respectively. We use the sketching matrix S to solve $\min_{x\in\mathbb{R}^d}\|Ax-b\|_p$, obtaining a constant-factor approximation solution x_c. The problem is then reduced to solving $\min_{x\in\mathbb{R}^d}\|Ax-z\|_p$ with $z=b-Ax_c$, for which we shall solve $\min_{x\in\mathbb{R}^d}\|S_1Ax-S_1z\|_p$ instead. Since S_1A has $\Omega(d^2)$ rows, we repeat the idea above and further subsample S_1A twice with different sampling parameters, getting S_2S_1A of $O(d\log d)$ rows and S_3S_1A of $O(d\,\mathrm{poly}(\epsilon^{-1}\log d))$ rows. The sampling matrix S_2 is used to obtain a constant-factor approximation solution $\hat{x}_c$ to $\min_{x\in\mathbb{R}^d}\|S_1Ax-S_1z\|_p$ and S_3 is used to solve $\min_{x\in\mathbb{R}^d}\|S_1Ax-(S_1z-S_1Ax'_c)\|_p$ with a near-optimal solution $\bar{x}'$. The near-optimal solution to $\min_{x\in\mathbb{R}^d}\|S_1Ax-S_1z\|_p$ is then $\bar{x}=\hat{x}_c+\bar{x}'$. Finally, the solution to the original problem is $\tilde{x}=x_c+\bar{x}$.

We shall maintain in parallel four independent (rescaled) row-sampled submatrices of the input matrix A, denoted by $\tilde{A}=SA$, $\tilde{A}_1=S_1A$, $\tilde{A}_2=S_2\tilde{A}_1$ and $\tilde{A}_3=S_3\tilde{A}_1$, where S,S_1,S_2,S_3 are rescaled sampling matrices. Recall that $z=b-Ax_c$, where x_c is a constant factor approximation. The corresponding sampled subvectors of b and z are $\tilde{b}=Sb$, $\tilde{b}_2=S_2S_1b$, $\tilde{b}_3=S_3S_1b$, $\tilde{z}_2=\tilde{b}_2-\tilde{A}_2x_c$ and $\tilde{z}_3=\tilde{b}_3-\tilde{A}_3x_c$, respectively. We shall keep updating these sampled submatrices and vectors. We make the rows of A global variables and $a_t\in\mathbb{R}^d$ is the t-th row of A. We denote by $A^{(t)}$ the first t rows of A and $b^{(t)}$ the first t coordinates of b. Furthermore, in our presentation of the algorithm, for a variable X, we denote by $X^{(t)}$ its value at the t-th stage in the online algorithm.

3.4.1 The case $p\in(1,2)$

We present our main algorithm for $p\in(1,2)$ in Algorithm 3.1. The following is the guarantee of the algorithm.

Theorem 3.4.1. *Let $A\in\mathbb{R}^{n\times d}$ and $b\in\mathbb{R}^n$. Algorithm 3.1 outputs a solution $\tilde{x}$ which satisfies that*

$$\|A\tilde{x}-b\|_p\le(1+\epsilon)\min_{x\in\mathbb{R}^d}\|Ax-b\|_p \tag{3.2}$$

with probability at least $0.98-\delta$ and makes

$$O\left(\frac{d}{\epsilon^2}\log^2 d\log^2\frac{d}{\epsilon}\cdot\log\frac{n\kappa^{OL}(A)}{\delta}\log\frac{1}{\delta}\right)$$

queries overall in total.

A major drawback of Algorithm 3.1 is the cost of calculating the online Lewis weights. Recall that the online Lewis weight of a_t is defined with respect to the first t rows of A. A naïve implementation would require storing the entire matrix A, partly defying the purpose of an online algorithm. Furthermore, the iterative procedure described after Definition 1.2.8 takes $O(\log t)$ iterations to reach a constant-factor approximation to the Lewis weights [16], where each iteration takes $O(td^2+d^3)$ time, which would become intolerable as t becomes large. To address this issue, we adopt the compression idea in [9],

Algorithm 3.1 Online Active Regression for $p \in (1, 2)$

Initialize: Let $\tilde{A}^{(d)}, \tilde{A}_1^{(d)}, \tilde{A}_2^{(d)}, \tilde{A}_3^{(d)}$ be the first d rows of A and $\tilde{b}^{(d)}$ be the first d rows of b.

$\beta \leftarrow \Theta(\log d)$
$\beta_1 \leftarrow \Theta(d \log(1/\epsilon\delta)/\epsilon^{2+p})$
$\beta_2 \leftarrow \Theta(\log d)$
$\beta_3 \leftarrow \Theta(\log^2 d \log(d/\epsilon) \log(1/\delta)/\epsilon^2)$
Retain the first d rows of A
while there is an additional row a_t **do**
    $\tilde{w}_t \leftarrow w_t(A^{(t)})$
    $p_t \leftarrow \min\{\beta \tilde{w}^{(t)}, 1\}$
    $(\tilde{A}^{(t)}, \tilde{b}^{(t)}) \leftarrow \textsc{Sample}(a_t, p_t, \tilde{A}^{(t-1)}, \tilde{b}^{(t-1)}, p)$
    $\tilde{w}_{1,t} \leftarrow w_t(A^{(t)})$
    $p_{1,t} \leftarrow \min\{\beta_1 \tilde{w}_{1,t}, 1\}$
    Sample a_t with probability $p_{1,t}$
    if a_t is sampled **then**
        $\tilde{A}_1^{(t)} \leftarrow \tilde{A}_1^{(t-1)} \circ a_t^\top p_{1,t}^{-1/p}$
        $\tilde{w}_{2,t} \leftarrow w_{\text{last}}(\tilde{A}_1^{(t)})$
        $p_{2,t} \leftarrow \min\{\beta_2 \tilde{w}_{2,t}, 1\}$
        $(\tilde{A}_2^{(t)}, \tilde{b}_2^{(t)}) \leftarrow \textsc{Sample}(a_t p_{1,t}^{-1/p}, p_{2,t}, \tilde{A}_2^{(t-1)}, \tilde{b}_2^{(t-1)}, p)$
        $\tilde{w}_{3,t} \leftarrow w_{\text{last}}(\tilde{A}_1^{(t)})$
        $p_{3,t} \leftarrow \min\{\beta_3 \tilde{w}_{3,t}, 1\}$
        $(\tilde{A}_3^{(t)}, \tilde{b}_3^{(t)}) \leftarrow \textsc{Sample}(a_t p_{1,t}^{-1/p}, p_{3,t}, \tilde{A}_3^{(t-1)}, \tilde{b}_3^{(t-1)}, p)$
    end if
end while
$x_c \leftarrow \textsc{Reg}(\tilde{A}, \tilde{b}, p)$
$\tilde{z}_2 \leftarrow \tilde{b}_2 - \tilde{A}_2 x_c$
$\hat{x}_c \leftarrow \textsc{Reg}(\tilde{A}_2, \tilde{z}_2, p)$
$\tilde{z}_3 \leftarrow \tilde{b}_3 - \tilde{A}_3 x_c$
$\bar{x}' \leftarrow \textsc{Reg}(\tilde{A}_3, \tilde{z}_3 - \tilde{A}_3 \hat{x}_c, p)$
$\bar{x} \leftarrow \hat{x}_c + \bar{x}'$
$\tilde{x} \leftarrow x_c + \bar{x}$
return $\tilde{x}$

Algorithm 3.2 $\textsc{Sample}(a_t, p_t, \tilde{A}^{(t-1)}, \tilde{b}^{(t-1)}, p)$

Sample a_t with probability p_t
if a_t is sampled **then**
    Query b_t
    $(\tilde{A}^{(t)}, \tilde{b}^{(t)}) \leftarrow (\tilde{A}^{(t-1)} \circ a_t^\top p_t^{-1/p}, \tilde{b}^{(t-1)} \circ b_t p_t^{-1/p})$
else
    $(\tilde{A}^{(t)}, \tilde{b}^{(t)}) \leftarrow (\tilde{A}^{(t-1)}, \tilde{b}^{(t-1)})$
end if

Algorithm 3.3 Compression algorithm for computing online Lewis weights

Initialize: B_0 contains the first d rows of A; $B_1, \dots, B_{\log n}$ are empty matrices; $Q = \Theta(\epsilon^{-2} d \log^3 n)$.

1: $\beta \leftarrow \Theta(\epsilon^{-2} \log(n/\delta) \log^2 n)$
2: **while** there is an additional row a_t **do**
3: $\quad B_0 \leftarrow B_0 \circ a_t$
4: $\quad$ **if** the size of B_0 exceeds Q **then**
5: $\quad\quad j \leftarrow$ the smallest index i such that B_i is empty
6: $\quad\quad M \leftarrow B_{i-1} \circ B_{i-2} \circ \cdots \circ B_0$
7: $\quad\quad p_i \leftarrow \min\{\beta w_i(M), 1\}$ for all i
8: $\quad\quad S \leftarrow$ rescaled sampling matrix with respect to probabilities $\{p_i\}_i$
9: $\quad\quad B_i \leftarrow SM$
10: $\quad\quad B_0, B_1, \dots, B_{i-1} \leftarrow$ empty matrix
11: $\quad$ **end if**
12: **end while**

which maintains $O(\log n)$ rescaled row-sampled submatrices of A, each having a small number of rows. The 'compression' algorithm is presented in Algorithm 3.3.

With the compression algorithm for A which maintains $B_0, \dots, B_{\log n}$, we can replace Line 7 of Algorithm 3.1 with

$$\tilde{w}_t \leftarrow w_{\text{last}}(B_{\log n} \circ B_{\log n - 1} \circ \cdots \circ B_0). \tag{3.3}$$

Similarly, we run an additional compression algorithm for each of $\tilde{A}_1$ and replace Lines 10, 15 and 18 with updates analogous to (3.3).

By the construction of the blocks B_i's, each B_i contains at most $R = O(\eta^{-2} d \log(n/\delta) \log^2 n)$ rows with probability at least $1 - \delta/\operatorname{poly}(n)$, sufficient for taking a union bound over all the blocks throughout the process of reading all n rows of A. Hence we may assume that each block B_i always contains at most R rows. Now, $\tilde{w}_t$ is calculated to be the Lewis weight of a matrix of $R' = O(Q + R \log n) = O(R \log n)$ rows, which can be done in $O((R'd^2 + d^3) \log R') = O(\eta^{-2} d^3 \operatorname{poly}(\log(n/\delta\eta)))$ time for a $(1 \pm \eta)$-factor approximation of Lewis weights, where the dependence on n is only polylogarithmic. The remaining question is correctness and the following theorem is the key to proving the correctness.

Theorem 3.4.2. *Let $A \in \mathbb{R}^{n \times d}$. With Algorithm 3.3 maintaining $B_0, \dots, B_{\log n}$, let $\tilde{w}_t$ be as in* (3.3) *for each $t \le n$. Then it holds with probability at least $1 - \delta/\operatorname{poly}(n)$ that*

$$(1 - \eta) w_t(A^{(t)}) \le \tilde{w}_t \le (1 + \eta) w_t(A^{(t)}), \quad \forall t \le n,$$

where $A^{(t)}$ is the submatrix consisting of the first t rows of A. The weights $\tilde{w}_t$ can be calculated in $O(\eta^{-2} d^3 \operatorname{poly}(\log(n/\delta\eta)))$ time and Algorithm 3.3 needs $O(\eta^{-2} d^2 \operatorname{poly}(\log(n/\delta)))$ words of space overall in total.

The following lemma is the key to the proof.

Lemma 3.4.3. *Let $A_i \in \mathbb{R}^{n_i \times d}$ $(i = 1, \dots, r)$, $B \in \mathbb{R}^{k \times d}$ and $M = A_1 \circ A_2 \circ \cdots \circ A_r \circ B$. For each $i \in [r]$, let $S_i \in \mathbb{R}^{m_i \times n_i}$ be the rescaled sampling matrix with respect to $p_{i,1}, \dots, p_{i,n_i}$ with $\min\{\beta w_j(A_i), 1\} \le p_{i,j} \le 1$ for each $j \in [n_i]$, where $\beta = \Theta(\eta^{-2} \log(d/\delta))$. Let $M' = S_1 A_1 \circ \cdots \circ S_r A_r \circ B$. Then, with probability at least $1 - \delta$, it holds for all $j = 1, \dots, k$ that*

$$(1 - \eta) w_{n_1 + \cdots + n_r + j}(M) \le w_{m_1 + \cdots + m_r + j}(M') \le (1 + \eta) w_{n_1 + \cdots + n_r + j}(M).$$

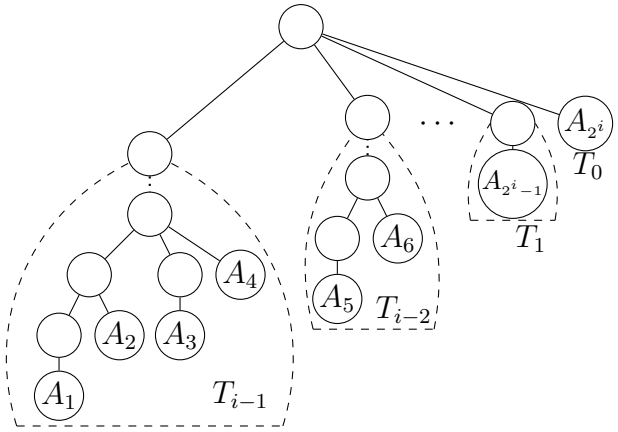


Figure 3.1: Tree structure of T_i for block B_i

A full version of the preceding lemma and its proof are postponed to Lemma 3.3.4. Now we turn to prove Theorem 3.4.2.

Proof. Observe that each block B_i is the compressed version of 2^i smaller matrices, say, $A_1, \ldots, A_{2^i}$, and each smaller matrix is compressed at most i times. The compression scheme inside B_i can be represented by a tree T_i, which satisfies that the root of T_i has i children $T_{i-1}, T_{i-2}, \ldots, T_0$. Every internal node of the tree represents a compression operation, which subsamples (with rescaling) the vertical concatenation of its children. An illustration of T_i is shown in Figure 3.1.

Consider a decompression process which begins at the root and goes down the tree level by level. When going down a level, we decompress each internal node on that level into the vertical concatenation of its children. When the decompression process is completed, we will have a vertical concatenation of the leaves, namely, $A_1 \circ A_2 \circ \cdots \circ A_{2^i}$, which is a submatrix of $A^{(t)}$.

Let i^* be the largest i such that B_i is nonempty. Consider the decompression process of all blocks $B_{\log n} \circ \cdots \circ B_0$. This process will terminate in i^* steps,

$$A^{(t,i^*)} \to A^{(t,i^*-1)} \to \cdots \to A^{(t,0)},$$

where $A^{(t,i^*)} = B_{\log n} \circ \cdots \circ B_0$ and $A^{(t,0)} = A^{(t)}$. Let $\tilde{w}_{t,j} = w_{\text{last}}(A^{(t,j)})$. Note that $\tilde{w}_{t,0} = w_t(A^{(t)})$. By Lemma 3.4.3 and our choices of parameters, we have

$$\left(1 - \frac{\eta}{2\log n}\right) \tilde{w}_{t,j} \le \tilde{w}_{t,j+1} \le \left(1 + \frac{\eta}{2\log n}\right) \tilde{w}_{t,j}$$

with probability at least $1 - \delta/\operatorname{poly}(n)$. Iterating yields that

$$\left(1 - \frac{\eta}{2\log n}\right)^{i*} w_t(A^{(t)}) \le \tilde{w}_{t,i^*} \le \left(1 + \frac{\eta}{2\log n}\right)^{i^*} w_t(A^{(t)}).$$

Note that $\tilde{w}_{t,i^*} = \tilde{w}_t$ per (3.3). Since $i^* \le \log n$, we have

$$(1-\eta) w_t(A^{(t)}) \le \tilde{w}_{t,i^*} \le (1+\eta) w_t(A^{(t)}).$$

Taking a union bound over all t gives the claimed result. □

Taking η to be a constant in Theorem 3.4.2 for constant-factor approximations to Lewis weights, we can now strengthen Theorem 3.4.1 as follows.

Theorem 3.4.4 (Strengthening Theorem 3.4.1)**.** *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$. Algorithm 3.1 outputs a solution $\tilde{x}$ which satisfies* (3.2) *with probability at least $0.98 - \delta$, making*

$$m = O\left(\frac{d}{\epsilon^2}\log^2 d \log^2 \frac{d}{\epsilon} \cdot \log \frac{n\kappa^{OL}(A)}{\delta} \log \frac{1}{\delta}\right)$$

queries. Furthermore, when implemented using the compression technique as explained above, with probability at least $0.98 - \delta$, Algorithm 3.1 uses $O(md)$ words of space in total and uses $O(nd^3 \operatorname{poly}(\log(n/\delta)))$ time to process the data stream (Lines 5–22).

The failure probability in Theorem 3.4.4 is $0.02 + \delta$. One 0.01 comes from obtaining constant-factor approximations x_c and $\hat{x}_c$. This 0.01 failure probability can be reduced to δ by employing the same boosting procedure in [59] which computes $\log(1/\delta)$ solutions, each being a constant-factor approximation with a constant probability, and then finds a good solution among them. The other 0.01 comes from bounding $\|S_1 z\|_p^p = O(\|z\|_p^p)$ and $\|S_3 S_1 z\|_p^p = O(\|S_1 z\|_p^p)$. The failure probability of bounding $\|S_3 S_1 z\|_p^p$ can be reduced to δ by employing the same boosting procedure in [60, Section 4.2.2], which uses $\log \frac{1}{\delta}$ independent copies of S_3, removes the largest 10% of $\|S_3 S_1 z\|_p^p$ and chooses an arbitrary remaining $\|S_3 S_1 z\|_p^p$. For $\|S_1 z\|_p^p$, we use Markov's inequality, obtaining that $\|S_1 z\|_p^p \le \|z\|_p^p/\delta$ with probability at least $1 - \delta$. Rescaling $\epsilon = \epsilon\delta$ yields

$$\beta_1 = \frac{d}{\delta^{2+p}\epsilon^{2+p}} \log \frac{1}{\epsilon\delta}.$$

Hence, the algorithm's overall failure probability can be reduced to δ (after rescaling) while maintaining asymptotically the same query and space complexity.

To conclude, the guarantee of Algorithm 3.1, with the aforementioned modification to boost the success probability, is as follows.

Theorem 3.4.5 ($1 - \delta$ success probability)**.** *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$. There exists an algorithm outputting a solution $\tilde{x}$ which satisfies* (3.2) *with probability at least $1 - \delta$, making*

$$m = O\left(\frac{d}{\epsilon^2}\log^2 d \log^2 \frac{d}{\epsilon} \cdot \log \frac{n\kappa^{OL}(A)}{\delta} \log^2 \frac{1}{\delta}\right)$$

queries. Furthermore, with probability at least $1 - \delta$, the algorithm uses $O(md)$ words of space in total and uses $O(nd^3 \operatorname{poly}(\log(n/\delta)))$ time to process the data stream.

3.4.2 The case $p = 2$

As mentioned in the preceding subsection, it is computationally expensive to compute Lewis weights in general. A special case is $p = 2$, where the Lewis weights are leverage scores and are much easier to compute. In this case, $w_i(A) = a_i^\top (A^\top A)^{-1} a_i$, and correspondingly, the online Lewis weights become online leverage scores, which are $w_i^{\mathrm{OL}}(A) = a_i^\top ((A^{(i)})^\top A^{(i)})^{-1} a_i$. It is much easier to compute $w_i^{\mathrm{OL}}(A)$ in the online setting because one can simply maintain $(A^{(i)})^\top A^{(i)}$ by adding $a_i a_i^\top$ when reading a new row a_i (viewed as a column vector). A naïve implementation of this algorithm would require inverting a $d \times d$ matrix at each step and we can further optimize the running time by noticing that $((A^{(i)})^\top A^{(i)})^{-1}$ receives a rank-one update at each step. This is the approach taken by [17] and [40] to calculate the online leverage scores in the online setting. Adopting this approach, we present our fast algorithm for $p = 2$ in Algorithm 3.4 and its guarantee below.

Theorem 3.4.6. *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$. Assume that the minimum singular value of the first d rows of A is $\sigma > 0$. With probability at least $0.98 - \delta$, Algorithm 3.4 makes*

$$m = O\left(\frac{d}{\epsilon^2}\log^2 d \log^2 \frac{d}{\epsilon} \cdot \log\left(n\frac{\|A\|_2}{\sigma}\right)\log\frac{1}{\delta}\right)$$

queries in total and maintains for each $T = d+1, \ldots, n$ a solution $\tilde{x}^{(T)}$ which satisfies that

$$\left\|A^{(T)}\tilde{x}^{(T)} - b^{(T)}\right\|_2 \leq (1+\epsilon)\min_{x\in\mathbb{R}^d}\left\|A^{(T)}x - b^{(T)}\right\|_2.$$

With probability at least $1 - \delta$, Algorithm 3.4 runs in a total of

$$O\left(\mathrm{nnz}(A)\log\frac{n}{\delta} + \frac{d^3}{\epsilon^4}\log n \log\frac{\|A\|_2}{\sigma}\log\frac{1}{\epsilon\delta}\left(\log\frac{n}{\delta} + d\right)\right)$$

time for processing the entire matrix A. Furthermore, with probability at least $1 - \delta$, it uses $O(md)$ words of space in total.

Remark. The failure probability of Theorem 3.4.6 can be reduced to δ by following the same approach as in Theorem 3.4.5. The query and space complexity remain asymptotically the same, but the runtime is increased to

$$O\left(\mathrm{nnz}(A)\log\frac{n}{\delta} + \frac{d^3}{\epsilon^4\delta^4}\log n \log\frac{\|A\|_2}{\sigma}\log\frac{1}{\epsilon\delta}(\log\frac{n}{\delta} + d)\right)$$

because of independent copies of S_1.

Remark. Compared to [40], our algorithm only requires a constant-factor approximation from the Johnson-Lindenstrauss matrix, saving a $1/\epsilon^2$ factor for the $\mathrm{nnz}(A)$ term in the runtime. In contrast, [40] generate a new Johnson-Lindenstrauss matrix every time for robustness against adversarial attacks.

In addition to the fast runtime, a major benefit of Algorithm 3.4 over the previous Algorithm 3.2 is that we can now output a guaranteed $(1+\epsilon)$-approximation solution $\hat{x}^{(t)}$ efficiently in all intermediate steps. instead of the outputting $\hat{x}^{(t)}$ only at the end. It is possible to do the same in Algorithm 3.1 for the general p, however, solving a general ℓ_p regression is computationally expensive and so we do not pursue maintaining the solution throughout the process. We also remark that the dependence on the online condition number of A in Theorems 3.4.1 and 3.4.4 is improved to $\log(\|A\|_2/\sigma) \leq \log\kappa^{\mathrm{OL}}(A)$.

3.4.3 The case $p = 1$

The case of $p = 1$ admits a simple sampling algorithm, based on [13] and [63]. We can simply sample the rows of $(A\ b)$ according to the online Lewis weights of A without the multiple sampling schemes described at the beginning of the section. The algorithm is presented in Algorithm 3.7, which can be implemented with the compression technique as described in Section 3.4.1 for the approximations of online Lewis weights. The guarantee of the algorithm is then as follows.

Theorem 3.4.7. *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$. Algorithm 3.7 outputs a solution $\tilde{x}$ which satisfies that*

$$\|A\tilde{x} - b\|_1 \leq \min_{x\in\mathbb{R}^d}\|Ax - b\|_1$$

with probability at least $1-\delta$ *and makes*

$$m = O\left(\frac{d}{\epsilon^2}\log\frac{d}{\epsilon\delta}\log n\log\kappa^{OL}(A)\right)$$

queries in total. When implemented with the compression technique, Algorithm 3.7 uses $O(md)$ *words of space in total with probability at least* $1-\delta$.

3.5 Proofs of Main Results

The framework of our Algorithms 3.1 and 3.4 follows from the algorithm of [59]. Below we shall refer to the full version [61] of the conference version [59] in our analysis. We first give a high-level idea of the proof in the offline case. Suppose that $z = b - Ax_c$ is the residual of a constant-factor approximation solution x_c and $R = \min_x \|Ax-b\|_p$ is the optimal error. Let $\mathcal{B}$ be an index set such that $\mathcal{B} = \{i \in [n] : \frac{|z_i|^p}{R^p} \geq \frac{w_i(A)}{\epsilon^p}\}$. Let $\bar{z}$ be equal to z but with all entries in $\mathcal{B}$ set to 0. It can be shown that $|\|S(z-\bar{z})\|_p^p - \|z-\bar{z}\|_p^p| = O(\epsilon R^p)$ and the most arduous and difficult argument is to establish that $|\|SAx - S\bar{z}\|_p^p - \|Ax-\bar{z}\|_p^p| = O(\epsilon R^p)$.

In the online case, we can analogously define $\mathcal{B} = \{i \in [n] : \frac{|z_i|^p}{R^p} \geq \frac{w_i^{\mathrm{OL}}(A)}{\epsilon^p}\}$ using the online Lewis weights and we would still have $|\|S(z-\bar{z})\|_p^p - \|z-\bar{z}\|_p^p| = O(\epsilon R^p)$ in this case as the proof in [61] still goes through. However, the key step, i.e. upper bounding $|\|SAx - S\bar{z}\|_p^p - \|Ax-\bar{z}\|_p^p|$, requires considerable change, since our algorithm is a sampling algorithm in order to accommodate the online setting while Musco et al.'s algorithm is an iterative algorithm which reduces the dimension by a constant factor in each iteration and the proof of the error guarantee in each iteration cannot be easily "flattened" to fit a one-shot sampling algorithm. Therefore, we adopt the framework in [16] with intermediate technical results in [59] and prove the following error guarantee.

Lemma 3.5.1 (Main lemma)**.** *Let* $S \in \mathbb{R}^{r\times n}$ *be the rescaled sampling matrix with respect to* $\{p_i\}_{(i)}$ *such that* $p_i = \min\{\beta\tilde{w}_i^{OL}(A), 1\}$ *for* $\beta = \Omega(\frac{1}{\epsilon^2}\log^2 d\log n\log\frac{1}{\delta})$ *and* $0 < \gamma \leq 1$*, it holds that*

$$\Pr\left\{\max_{\|Ax\|_p \leq \sqrt{\gamma}R}\left|\|SAx - S\bar{z}\|_p^p - \|Ax-\bar{z}\|_p^p - \left(\|S\bar{z}\|_p^p - \|\bar{z}\|_p^p\right)\right| \geq \epsilon R^p\right\} \leq \delta.$$

The detailed statement and proof of Lemma 3.5.1 will be given in Section 3.5.3. We note that for the case $p = 2$, Theorem 3.4.6 requires a different version of Lemma 3.5.1 because the sampling matrices in Algorithm 3.4 do not have independent rows. The details are further postponed in Appendix 3.5.4.

In the next two subsections, we shall establish two basic results regarding online Lewis weights and their approximations, namely,

(i) the online ℓ_p Lewis weights calculated in Algorithms 3.1 and 3.4 are within an absolute constant factor of the corresponding true ℓ_p online Lewis weights, and

(ii) the sum of approximate ℓ_p online Lewis weights are bounded.

When $p = 1$, the framework of our Algorithm 3.7 simply adapts the ℓ_1 Lewis weight sampling scheme in [12] and [63] to online Lewis weight sampling. In order to prove Theorem 3.4.7, it suffices to show conditions (i) and (ii) hold for $p = 1$.

We remark that our proof relies on the fact that $w_i^{OL}(A) \geq w_i(A)$, which is guaranteed by the monotonicity of Lewis weights (Lemma 3.3.1) when $p \leq 2$. The monotonicity property does not always hold when $p > 2$; thus, we only consider the case of $p \leq 2$ in this chapter and leave the case of $p > 2$ to future work.

3.5.1 Approximating Online Lewis Weights

We first deal with condition (i), namely, Algorithms 3.1 and 3.4 calculate good approximations to Lewis weights.

The guarantee of approximate ℓ_2 online Lewis weights follows from the works of [18] and [40], which we cite below.

Lemma 3.5.2 ([18, Theorem 2.3], [40, Lemma 3.4])**.** *Let $\{\tilde{w}_i\}_i$ be the approximate Lewis weights in Algorithm 3.4 and $\beta = \Theta(\epsilon^{-2}\log(d/\delta))$. Let S be the rescaled sampling matrix with respect to $\{\tilde{w}_i\}_i$. It holds with probability at least $1-\delta$ that*

$$(1-\epsilon)(A^{(t)})^\top A^{(t)} \preceq (SA^{(t)})^\top (SA^{(t)}) \preceq (1+\epsilon)(A^{(t)})^\top A^{(t)}$$

for all $t \in \{d+1,\ldots,n\}$ and S has $O(\beta\sum_{i=1}^n \tilde{w}_i)$ rows.

As a consequence,

$$\tilde{w}_t \geq \frac{1}{1+\epsilon}\cdot a_i^\top((A^{(t)})^\top A^{(t)})^{-1}a_i \geq (1-\epsilon)w_t^{\mathrm{OL}}(A), \quad t = d+1,\ldots,n.$$

This establishes (i) when $p=2$. The case of general p follows from Theorem 3.4.2.

3.5.2 Sum of Online Lewis Weights

Suppose that condition (i) holds, (ii) would follow from that the sum of true ℓ_p online Lewis weights are bounded, which are exactly the following two lemmas, for $p=2$ and $p\in[1,2)$, respectively.

Lemma 3.5.3 ([18, Lemma 2.2])**.** *Let $p=2$. Suppose that the first d rows of A has the smallest singular value $\sigma>0$. It holds that $\sum_{i=1}^n w_i^{OL}(A) = O(d\log(\|A\|_2/\sigma))$.*

Lemma 3.5.4. *Let $p\in[1,2)$. It holds that $\sum_{i=1}^n w_i^{OL}(A) = \mathcal{O}(d\log n\cdot\log\kappa^{OL}(A))$.*

The case $p=1$ of Lemma 3.5.4 appeared in [9]. The remainder of this section is dedicated to proving Lemma 3.5.4 for $p\in(1,2)$. We shall follow the same approach as in the proof of [9, Lemma 5.15].

We start with the following lemma.

Lemma 3.5.5. *If the leverage scores of A are at most $C>0$, then the ℓ_p Lewis weights of A are at most C for $p\in[1,2]$.*

Proof. This is the generalization of [9, Lemma 5.12] and we follow the same argument.

By the assumption, we have $a_i^\top(A^\top A)^{-1}a_i \leq C$ for $i\in[n]$. We prove by induction that for iteration j in the Lewis weight iteration, we have $W^{(j)} \preceq C^{1-(1-p/2)^j}I_n$.

For the base case $j=1$, we have $W_{i,i}^{(j)} = (a_i^\top(A^\top A)^{-1}a_i)^{p/2} \leq C^{p/2}$. Thus $W^{(1)} \preceq C^{p/2}I_n$ as desired.

For iteration j, by the induction hypothesis, we have $W^{(j-1)} \preceq C^{1-(1-p/2)^{j-1}}I_n$, which implies that $(W^{(j-1)})^{1-2/p} \succeq C^{(1-(1-p/2)^{j-1})(1-2/p)}I_n$ since $1-2/p\leq 0$. Thus,

$$A^\top(W^{(j-1)})^{1-2/p}A \succeq C^{(1-(1-p/2)^{j-1})(1-2/p)}A^\top A,$$

and

$$(A^\top(W^{(j-1)})^{1-2/p}A)^{-1} \preceq C^{(1-(1-p/2)^{j-1})(2/p-1)}(A^\top A)^{-1}.$$

It then follows from the definition of Lewis weights (Definition 1.2.8) that

$$(W^{(j)}_{i,i})^{2/p} = a_i^\top (A^\top (W^{(j-1)})^{1-2/p} A)^{-1} a_i \le C^{(1-(1-p/2)^{j-1})(2/p-1)} a_i^\top (A^\top A)^{-1} a_i \le C^{(1-(1-p/2)^{j-1})(2/p-1)+1}.$$

Notice that $((1-(1-p/2)^{j-1})(2/p-1)+1)p/2 = 1-(1-p/2)^j$, we have obtained that $W^{(j)}_{i,i} \le C^{1-(1-p/2)^j}$ for all i, i.e., $W^{(j)} \preceq C^{1-(1-p/2)^j} I_n$. The induction step is established.

The claim follows the convergence of Lewis weight iteration [16]. □

Lemma 3.5.6. *Given $A = [a_1, \ldots, a_n]^\top \in \mathbb{R}^{n\times d}$, let $B \in \mathbb{R}^{(n+1)\times d}$ be defined as*

$$B = [a_1, \ldots, a_{j-1}, b_j, a_{j+1}, \ldots, a_n, b_{n+1}]^\top,$$

where $b_j = (1-\gamma)^{1/p} a_j$ and $b_{n+1} = \gamma^{1/p} a_j$ for some $\gamma \in [0,1]$ and $j \in [n]$. Then we have $w_i(A) = w_i(B)$ for $i \ne j, n+1$, $w_j(B) = (1-\gamma) w_j(A)$ and $w_{n+1}(B) = \gamma w_j(A)$.

Proof. Without loss of generality, we suppose $j = n$. Let $W \in \mathbb{R}^{n\times n}$ be the diagonal Lewis weight matrix of A, i.e., $W_{i,i} = w_i(A)$. Let $\overline{W}^{(n+1)\times(n+1)}$ be a diagonal matrix where $\overline{W}_{i,i} = w_i(A)$ for $i = 1, \ldots, n-1$, $\overline{W}_{n,n} = (1-\gamma) w_n(A)$ and $\overline{W}_{n+1,n+1} = \gamma w_n(A)$. According to the uniqueness of Lewis weights, it suffices to show that $\tau_i(\overline{W}^{1/2-1/p} B) = \overline{W}_{i,i}$ for $i \in [n+1]$.

Notice that the first $n-1$ rows of $\overline{W}^{1/2-1/p} B$ are the same as those of $\overline{W}^{1/2-1/p} A$. The last two rows of $\overline{W}^{1/2-1/p} B$ are $w_n(A)^{1/2-1/p}(1-\gamma)^{1/2-1/p}(1-\gamma)^{1/p} a_n = w_n(A)^{1/2-1/p}(1-\gamma)^{1/2} a_n$ and $w_n(A)^{1/2-1/p}\gamma^{1/2} a_n$, respectively. Thus it holds that for any vector y, $\|W^{1/2-1/p} A y\|_2^2 = \|\overline{W}^{1/2-1/p} B y\|_2^2$, which indicates that the leverage scores of the first $n-1$ rows of $W^{1/2-1/p} A$ are the same as those of $\overline{W}^{1/2-1/p} B$, i.e., $\tau_i(\overline{W}^{1/2-1/p} B) = W_{i,i} = \overline{W}_{i,i}$ for $1 \le i \le n-1$.

For the last two rows, we have $\tau_n(\overline{W}^{1/2-1/p} B) = (1-\gamma)\tau_n(W^{1/2-1/p} A) = \overline{W}_{n,n}$ and $\tau_{n+1}(\overline{W}^{1/2-1/p} B) = \gamma \cdot \tau_n(W^{1/2-1/p} A) = \overline{W}_{n+1,n+1}$. Thus we have $\tau_i(\overline{W}^{1/2-1/p} B) = \overline{W}_{i,i}$ for all $i \in [n+1]$. □

Corollary 3.5.7. *For any matrix $A \in \mathbb{R}^{n\times d}$. Let $B \in \mathbb{R}^{n\times d}$ have the same rows but with the j-th row reweighted by a factor $\alpha \in [0,1]$. Then for all $i \ne j$, $w_i(B) \ge w_i(A)$.*

Proof. Let $\gamma = 1-\alpha^p$ and $\bar{B} \in \mathbb{R}^{(n+1)\times d} = [a_1, \ldots, a_{j-1}, (1-\gamma)^{1/p} a_j, a_{j+1}, \ldots, a_n, \gamma^{1/p} a_j]^\top$. By Lemma 3.5.6, we have $w_i(\bar{B}) = w_i(A)$ for $i \ne j$. Then by Lemma 3.3.1 we have $w_i(B) \ge w_i(\bar{B}) = w_i(A)$. □

As mentioned before, we follow the approach in [9] to upper bound the sum of the online Lewis weights. It makes a critical use of an upper bound on the sum of online λ-leverage scores of a matrix but does not provide a proof for the case of small λ. We reproduce the upper bound and provide a proof for completeness below.

Definition 3.5.8 (Online Ridge Leverage Scores)**.** Let $A \in \mathbb{R}^{n\times d}$ and $\lambda \ge 0$ be a regularization parameter. The online ridge leverage score of row $a_i \in \mathbb{R}^d$ is defined to be $\tau_i^{\mathrm{OL}}(A;\lambda) = \min\{a_i^\top ((A^{(i-1)})^\top A^{(i-1)} + \lambda I_d)^{-1} a_i, 1\}$.

The next lemma upper bounds the sum of online ridge leverage scores, which was stated as [9, Lemma 2.2].

Lemma 3.5.9 (Bound on Sum of Online Ridge Leverage Scores)**.** *Suppose that $A \in \mathbb{R}^{n\times d}$ and $\sigma^* = \min_{i\in[n]} \sigma_{\min}(A^{(i)})$, where $\sigma_{\min}$ denotes the smallest singular value of a matrix. Let $\lambda > 0$. It holds that*

$$\sum_{i=1}^{n} \tau_i^{OL}(A;\lambda) = \begin{cases} O(d\log(\|A\|_2^2/\lambda)), & \lambda \geq (\sigma^*)^2; \\ O(d\log \kappa^{OL}(A)), & \lambda < (\sigma^*)^2. \end{cases}$$

The exact case of $\lambda \geq (\sigma^*)^2$ was proved in [18]. The claim of the case $\lambda < (\sigma^*)^2$ appeared in [9] without an explicit proof. We shall supply a proof below for completeness. First we need the following observation of the leverage scores.

Proposition 3.5.10. *For $A \in \mathbb{R}^{n\times d}$, let τ_i be the leverage score of row a_i. Then $\tau_i = \min_{x\in\mathbb{R}^d} \|x\|_2^2$ subject to $x^\top A = a_i^\top$. Suppose $\mu \in \mathbb{R}^d$ is the minimizer, then $\mu_i \in [0,1]$.*

Proof. Suppose that $A = U\Sigma V^\top$ is the singular value decomposition of A, where $U \in \mathbb{R}^{n\times d}$ has orthonormal columns and $\Sigma, V \in \mathbb{R}^{d\times d}$ are invertible. Then $\tau_i = a_i^\top (A^\top A)^{-1} a_i = \|e_i^\top U\|_2^2$, where e_i is the i-th canonical basis vector. Suppose that $x^\top A = a_i^\top = e_i^\top A$. Multiplying both sides by $V\Sigma^{-1}$ leads to $x^\top U = e_i^\top U$ and thus $\|x\|_2^2 = \|x\|_2^2 \|U\|_2^2 \geq \|x^\top U\|_2^2 = \tau_i$. Next, we show there exists a vector $\mu \in \mathbb{R}^n$ satisfying $\mu^\top A = a_i^\top$ and $\|\mu\|_2^2 = \tau_i$.

The matrix $U \in \mathbb{R}^{n\times d}$ can be transformed to $U' = \begin{pmatrix} I_d \\ 0 \end{pmatrix}$ through an orthogonal transformation E, that is, $U' = EU$. Hence, we can find a vector $y \in \mathbb{R}^n$ such that $e_i^\top U = y^\top U' = y^\top EU$ and y only has nonzero entries in the first d coordinates. Hence, let $\mu = y^\top E$ and we get $\mu^\top A = (\mu^\top U)\Sigma V^\top = (e_i^\top U)\Sigma V^\top = a_i^\top$ and $\|\mu\|_2^2 = \|y\|_2^2 = \|e_i^\top U\|_2^2 = \tau_i$.

Without loss of generality, we assume $i = n$. We decompose μ as $\mu^\top = (\nu^\top \ \mu_n)$, where $\nu \in \mathbb{R}^{n-1}$ is the first $n-1$ coordinates of μ. Since $\tau_i \in [0,1]$, we know that $|\mu_n| \leq 1$. Next we show that $\mu_n \geq 0$ by contradiction. Suppose that $\mu_n < 0$. Observe that $\mu^\top A = (\nu^\top \ \mu_n) \begin{pmatrix} A^{(n-1)} \\ a_n^\top \end{pmatrix} = \nu^\top A^{(n-1)} + \mu_n a_n^\top = a_n^\top$, whence it follows that $a_n^\top = \frac{\nu^\top A^{(n-1)}}{1-\mu_n}$. Let $x = (\frac{\nu^\top}{1-\mu_n} \ 0)$, then $x^\top A = a_n^\top$ while $\|x\|_2 = \left\|\frac{\nu^\top}{1-\mu_n}\right\|_2 < \|\nu^\top\|_2 \leq \|\mu\|_2$, contradicting the minimality of $\|\mu\|_2$. Therefore, we conclude that $\mu_n \geq 0$. □

Now we are ready to prove Lemma 3.5.9.

Proof of Lemma 3.5.9. The case $\lambda \geq (\sigma^*)^2$ is exactly [16, Theorem 2.2], which actually holds for all $\lambda > 0$. In the remainder of the proof, we assume that $\lambda < (\sigma^*)^2$.

We claim that

$$\frac{1}{8}\tau_{i+1}^{\mathrm{OL}}(A,\lambda) \leq a_{i+1}^\top (A^{(i+1)\top} A^{(i+1)} + \sigma^{*2} I_d)^{-1} a_{i+1}. \tag{3.4}$$

Assuming that the claim is true for now, we will have

$$\begin{aligned} \sum_{i=1}^{n} \tau_i^{\mathrm{OL}}(A,\lambda) &\leq 8 \sum_i a_i^\top (A^{(i)\top} A^{(i)} + \sigma^{*2} I_d)^{-1} a_i \\ &\leq 8 \sum_i a_i^\top (A^{(i-1)\top} A^{(i-1)} + \sigma^{*2} I_d)^{-1} a_i \\ &= 8 \sum_i \tau_i^{\mathrm{OL}}(A;\sigma^{*2}) \end{aligned}$$

$$= O\left(d \log \frac{\|A\|_2}{\sigma^{*2}}\right)$$
$$= O\left(d \log \kappa^{\mathrm{OL}}(A)\right)$$

as desired. For the inequalities above, the first one follows from the claim (3.4), the second one the fact that $A^{(i)\top}A^{(i)} \preceq A^{(i)\top}A^{(i)} + a_{i+1}a_{i+1}^\top = A^{(i+1)\top}A^{(i+1)}$. In the rest of the proof, we prove the claim (3.4).

Suppose that the singular value decomposition of $A^{(i)}$ and $A^{(i+1)}$ are $U^{(i)}\Sigma^{(i)}V^{(i)\top}$ and $U^{(i+1)}\Sigma^{(i+1)}V^{(i+1)\top}$ respectively. Let the singular values of $A^{(i)}$ and $A^{(i+1)}$ be $\sigma_1^{(i)}, \ldots, \sigma_d^{(i)}$ and $\sigma_1^{(i+1)}, \ldots, \sigma_d^{(i+1)}$ both with descending order. Let $e_{i+1}^\top$ be the $(i+1)$-st canonical basis vector and τ be the leverage score of row a_{i+1} in $A^{(i+1)}$. By the definition of leverage scores, we have $\tau = \|e_{i+1}U^{(i+1)}\|_2$. According to Proposition 3.5.10, there exists $(\mu, w) \in \mathbb{R}^i \times [0,1]$ such that $(\mu^\top \ w)\begin{pmatrix} A^{(i)} \\ a_{i+1}^\top \end{pmatrix} = a_{i+1}^\top$ and $\|\mu\|_2^2 + w^2 = \tau$. Hence,

$$\begin{aligned}
\tau_{i+1}^{\mathrm{OL}}(A, \lambda) &= a_{i+1}^\top (A^{(i)\top}A^{(i)} + \lambda I_d)^{-1} a_{i+1} \\
&= \frac{1}{(1-w)^2}\mu^\top A^{(i)}(A^{(i)\top}A^{(i)} + \lambda I_d)^{-1}A^{(i)\top}\mu \\
&= \frac{1}{(1-w)^2}\mu^\top U^{(i)} \begin{pmatrix} \frac{\sigma_1^{(i)2}}{\sigma_1^{(i)2}+\lambda} & & \\ & \ddots & \\ & & \frac{\sigma_d^{(i)2}}{\sigma_d^{(i)2}+\lambda} \end{pmatrix} U^{(i)\top}\mu \\
&\le \frac{1}{(1-\sqrt{\tau})^2}\mu^\top U^{(i)}U^{(i)\top}\mu \\
&= \frac{\tau}{(1-\sqrt{\tau})^2},
\end{aligned}$$

and similarly,

$$\begin{aligned}
& a_{i+1}^\top (A^{(i+1)\top}A^{(i+1)} + (\sigma^*)^2 I_d)^{-1} a_{i+1} \\
&= e_{i+1}^\top U^{(i+1)} \begin{pmatrix} \frac{\sigma_1^{(i+1)2}}{\sigma_1^{(i+1)2}+\sigma^{*2}} & & \\ & \ddots & \\ & & \frac{\sigma_d^{(i+1)2}}{\sigma_d^{(i+1)2}+\sigma^{*2}} \end{pmatrix} U^{(i+1)\top} e_{i+1} \\
&\ge \frac{1}{2}\tau.
\end{aligned}$$

For notation convenience, let R denote the right-hand side of (3.4). When $\tau \ge \frac{1}{4}$, we have $\tau_{i+1}^{\mathrm{OL}}(A;\lambda) \le 1 \le 8R$, where $\tau_{i+1}^{\mathrm{OL}}(A;\lambda) \le 1$ follows from the Definition 3.5.8. When $0 \le \tau \le \frac{1}{4}$, it holds that $\tau_{i+1}^{\mathrm{OL}}(A;\lambda) \le 4\tau \le 8R$. Hence, it always holds that $\tau_{i+1}^{\mathrm{OL}}(A,\lambda) \le 8R$ when $0 \le \lambda \le (\sigma^*)^2$, establishing the claim. □

Finally, we present our proof of Lemma 3.5.4, following the first part in the proof of [9, Lemma 5.15] but with a different argument in the second part.

Proof of Lemma 3.5.4. We follow the proof idea of Lemma 5.15 in [9]. Suppose that $\lambda > 0$. Let $B_0 = \lambda I_d$, $B = \underbrace{B_0 \circ \cdots \circ B_0}_{n \text{ times}}$ and $X \triangleq B \circ A$. Let T be the upper bound of the sum of

online leverage scores of A with regularization parameter λ. Following the proof of Lemma 5.15 of [9], we have $\sum_{i=1}^{n} w_i^{\mathrm{OL}}(X) = O(T \log n) = O(d \log n \log \kappa^{\mathrm{OL}}(A))$ by Lemma 3.5.9.

Now, let W_A be the Lewis weight matrix of A and $L = A^\top W_A^{1-2/p} A$. Let $\sigma = \lambda_{\min}(L)$, the smallest eigenvalue of L, and $\rho = \min_i (L^{-1})_{ii}$, the smallest diagonal element of L_i^{-1}. Choose $\lambda \leq \left(\frac{\sigma}{n}\right)^{1/p} \rho^{(2-p)/(2p)}$, $\mu = (\frac{n\lambda^2}{\sigma})^{p/(2-p)}$, $U_X = \mu I_{nd}$ and $W_X = \begin{bmatrix} U_X & \\ & W_A \end{bmatrix}$. We claim that

$$\frac{1}{2}\mu^{2/p} \leq B_j^\top \left(A^\top W_A^{1-2/p} A + B^\top U_X^{1-2/p} B\right)^{-1} B_j, \tag{3.5}$$

$$\frac{1}{2}(w_i(A))^{2/p} \leq a_i^\top \left(A^\top W_A^{1-2/p} A + B^\top U_X^{1-2/p} B\right)^{-1} a_i \tag{3.6}$$

for all $j \in [nd]$ and all $i \in [n]$. Observe that $B^\top U_X^{1-2/p} B = n\lambda^2 \mu^{1-2/p} I_d \preceq \sigma I_d \preceq L$. Thus,

$$a_i^\top (L + n\lambda^2 \mu^{1-2/p} I_d)^{-1} a_i \geq \frac{1}{2} a_i^\top L^{-1} a_i = \frac{1}{2}(w_i(A))^{2/p},$$

establishing (3.6). Similarly, since $B_j = \lambda e_i$ for some i,

$$B_j^\top (L + n\lambda^2 \mu^{1-2/p} I_d)^{-1} B_j \geq \frac{1}{2}\lambda^2 (L^{-1})_{i,i} \geq \frac{1}{2}\lambda^2 \rho \geq \frac{1}{2}\mu^{2/p},$$

establishing (3.5). It then follows from Lemma 3.3.3 that $w_i(A) \leq 2 w_{nd+i}(X)$. Applying the argument above to the n submatrices which consist of the first i rows of A for each $i = 1, \ldots, n$, we see that we can choose λ to be sufficiently small such that $w_i^{\mathrm{OL}}(A) \leq 2 w_{nd+i}^{\mathrm{OL}}(X)$ for all i. Therefore, $\sum_i w_i^{\mathrm{OL}}(A) = O(d \log n \log \kappa^{\mathrm{OL}}(A))$. This completes the proof of Lemma 3.5.4. □

In the analysis of Algorithm 3.1, we shall apply Lemma 3.5.4 to $\tilde{A}_1 = S_1 A$, where S_1 is a rescaled sampling matrix w.r.t. the online Lewis weights of A. To upper bound $\kappa^{\mathrm{OL}}(S_1 A)$, we shall need the following auxiliary lemma.

Lemma 3.5.11. *Let $p \in [1, 2)$ and S is a rescaled sampling matrix w.r.t. the online Lewis weights of A and the oversampling parameter β. With probability at least $1 - \delta$, it holds that $\log \kappa^{OL}(SA) = O(\log(n\kappa^{OL}(A)/(\beta\delta)))$.*

First, we note the following facts. For any two matrices A and B, $\|AB\|_2 \leq \|A\|_2 \|B\|_2$, and when A has full row rank and $B \neq 0$, $\sigma_{\min}(AB) \geq \sigma_{\min}(A)\sigma_{\min}(B)$, where $\sigma_{\min}(\cdot)$ denotes the smallest nonzero singular value of a matrix.

It is clear that S, which is a rescaled sampling matrix, has full row rank. By the definition of the online condition number,

$$\begin{aligned}
\kappa^{\mathrm{OL}}(SA) = \|SA\|_2 \max_i \frac{1}{\sigma_{\min}(SA^{(i)})} &\leq \|S\|_2 \|A\|_2 \max_i \frac{1}{\sigma_{\min}(SA^{(i)})_i} \\
&\leq \|S\|_2 \|A\|_2 \max_i \frac{1}{\sigma_{\min}(S)\sigma_{\min}(A)_i} \\
&= \frac{\sigma_{\max}(S)}{\sigma_{\min}(S)} \kappa^{\mathrm{OL}}(A).
\end{aligned}$$

Now, observe that $\sigma_{\max}(S) = \max_i p_i^{-1/p} = (\min_i p_i)^{-1/p}$ and $\sigma_{\min}(S) = \min_i p_i^{-1/p} = (\max_i p_i)^{-1/p}$, where $\min\{\beta w_i^{\mathrm{OL}}(A), 1\} \leq p_i \leq 1$. It is clear that $\sigma_{\min}(S) \geq 1$. For the

upper bound of $\sigma_{\max}(S)$, note that a row i with $w_i^{\mathrm{OL}}(A) \leq \delta/n$ will be sampled with probability

$$1 - \left(1 - \frac{\delta}{n}\right)^n \leq \delta.$$

Hence, with probability at least $1-\delta$, none of the rows i with $w_i^{\mathrm{OL}}(A) \leq \delta/n$ is sampled and so $\min_i p_i \geq \beta\delta/n$ and $\sigma_{\max}(S) \leq (n/(\delta\beta))^{1/p}$. Therefore, we conclude that with probability at least $1-\delta$,

$$\kappa^{\mathrm{OL}}(SA) \leq \left(\frac{n}{\beta\delta}\right)^{1/p} \kappa^{\mathrm{OL}}(A).$$

3.5.3 Proof of Theorem 3.4.4

We may assume that $n > (d/\epsilon^2)\,\mathrm{poly}(\log(d \log n \log \kappa^{\mathrm{OL}}))\log(1/\delta)$, otherwise it will not be necessary to sample for solving the regression problem.

The main lemma we shall prove is Lemma 3.5.15. Before proving it, we state a series of lemmas, namely Lemmas 3.5.12, 3.5.13 and 3.5.14, which, together with Lemma 3.5.15, will prove Theorem 3.4.4. We do not repeat the proof of Lemmas 3.5.12, 3.5.13 and 3.5.14 because they are almost identical to those in [61], except that Lewis weights are replaced with online Lewis weights, which does not affect all these proofs since it is always true that $w_i^{\mathrm{OL}}(A) \geq w_i(A)$.

Lemma 3.5.12 (Theorem 3.2 in [61]). *Let $A \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$, $p \in [1,2]$. If we sample A and obtain x_c by Algorithm 3.1 or Algorithm 3.4 with $\beta = \Theta(\log(d/\delta))$ then with probability at least $1-\delta$,*

$$\|Ax_c - b\|_p \leq \frac{2^{1+\frac{1}{p}}3}{\delta^{\frac{1}{p}}} R.$$

In the remainder of this subsection, we define $z = b - Ax_c$ with a constant δ in Lemma 3.5.12 and $R = \|z\|_p$, then $R \leq C \min_{x\in\mathbb{R}^d} \|Ax - b\|_p$ for some constant $C > 0$.

Lemma 3.5.13 (Lemma 3.6 in [61]). *Let $\mathcal{B}$ be an index set defined as*

$$\mathcal{B} = \left\{ i \in [n] : \frac{|z_i|^p}{R^p} \geq \frac{w_i^{OL}(A)}{\epsilon^p} \right\}.$$

Let $\bar{z}$ be equal to z but with all entries in $\mathcal{B}$ set to 0. Then for all $x \in \mathbb{R}^d$ with $\|Ax\|_p \leq R$,

$$\left| \|Ax - z\|_p^p - \|Ax - \bar{z}\|_p^p - \|z - \bar{z}\|_p^p \right| = O(\epsilon)R^p.$$

Lemma 3.5.14 (Lemma 3.7 in [61]). *Consider the same setting as in Lemma 3.5.13. Let S be a sampling matrix according to the online Lewis weights with oversampling parameter $\beta = \Omega(\log d)$. With probability at least 0.995, $\|Sz\|_p^p = O(R^p)$ and for all $x \in \mathbb{R}^d$ with $\|Ax\|_p \leq R$, it holds that*

$$\left| \|SAx - Sz\|_p^p - \|SAx - S\bar{z}\|_p^p - \|Sz - S\bar{z}\|_p^p \right| = O(\epsilon)R^p.$$

Next, We follow the idea in [59], narrowing the region of x from $\{x : \|Ax\|_p \leq R\}$ to $\{x : \|Ax\|_p \leq \sqrt{\gamma}R\}$, where $0 < \gamma \leq 1$. This will guarantee that $\|SAx - S\bar{z}\|_p$ is close to $\|Ax - \bar{z}\|_p$ for all x near $x^* = \mathrm{argmin}_x \|Ax - \bar{z}\|_p$, which is enough for approximately solving the original regression problem $\min_x \|Ax - b\|_p$. Now, we restate our main lemma below.

Lemma 3.5.15 (Main Lemma)**.** *Let $A \in \mathbb{R}^{n\times d}$ and $\bar{z}$ be as defined in Lemma 3.5.13. Let S be the rescaled sampling matrix of the first kind with sampling probabilities $p_1, \ldots, p_n$, where $p_i = \min\{\beta\tilde{w}_i^{OL}(A), 1\}$ and $\beta = \Theta(\frac{\gamma}{\epsilon^2}\log^2 d \log n \log\frac{1}{\delta})$. It holds that*

$$\Pr\left\{\max_{\|Ax\|_p\le\sqrt{\gamma}R}\left|\|SAx - S\bar{z}\|_p^p - \|Ax-\bar{z}\|_p^p - \left(\|S\bar{z}\|_p^p - \|\bar{z}\|_p^p\right)\right| \ge \epsilon R^p\right\} \le \delta.$$

For notational simplicity, given $A \in \mathbb{R}^{n\times d}$ and $\epsilon, R > 0$, we say $z \in \mathbb{R}^n$ conforms to (A, ϵ, R) if $|z_i|^p \le (R/\epsilon)^p w_i^{\mathrm{OL}}(A)$ for all i.

Below we first prove Theorem 3.4.4 using Lemmas 3.5.12 to 3.5.15. Then we prove Lemma 3.5.15 in Section 3.5.3, taking $\gamma = 1$. In Section 3.6, we shall explain the reason for introducing the parameter γ.

Lemma 3.5.16. *Let $A \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$ and $0 < \gamma \le 1$. Let S be as defined in Lemma 3.5.15. Let x_c be the constant factor approximation obtained by Algorithm 3.1 or 3.4 and $z = b - Ax_c$. Suppose that $\tilde{x}$ satisfies $\|SA\tilde{x} - Sz\|_p \le (1+\epsilon)\min_{x\in\mathbb{R}^d}\|SAx - Sz\|_p$, $x^* = \arg\min_{x\in\mathbb{R}^d}\|Ax - z\|_p$ and $\|A\tilde{x} - Ax^*\|_p \le \sqrt{\gamma}R$. It holds that*

$$\|A\tilde{x} - z\|_p \le (1 + O(\epsilon))\min_{x\in\mathbb{R}^d}\|Ax - z\|_p$$

with probability at least $0.99 - O(\delta)$.

Proof. Let $x' = \tilde{x} - x^*$ and $z' = z - Ax^*$. We have that

$$\begin{aligned}
&\|A\tilde{x} - z\|_p^p - \|Ax^* - z\|_p^p \\
&= \|A\tilde{x} - z\|_p^p - \|SA\tilde{x} - Sz\|_p^p + \|SA\tilde{x} - Sz\|_p^p - \|SAx^* - Sz\|_p^p \\
&\qquad + \|SAx^* - Sz\|_p^p - \|Ax^* - z\|_p^p \\
&\le \|A\tilde{x} - z\|_p^p - \|SA\tilde{x} - Sz\|_p^p + \|SAx^* - Sz\|_p^p - \|Ax^* - z\|_p^p \\
&= \|Ax' - z'\|_p^p - \|SAx' - Sz'\|_p^p + \|Sz'\|_p^p - \|z'\|_p^p.
\end{aligned}$$

Here, it holds that $\|Ax'\|_p \le \sqrt{\gamma}R$ and $\|z'\|_p = R$. Then,

$$\begin{aligned}
&\|Ax' - z'\|_p^p - \|SAx' - Sz'\|_p^p + \|Sz'\|_p^p - \|z'\|_p^p \\
&= \|Ax' - z'\|_p^p - \|SAx' - Sz'\|_p^p + \|S\bar{z}'\|_p^p + \|Sz' - S\bar{z}'\|_p^p - \|\bar{z}'\|_p^p - \|z' - \bar{z}'\|_p^p \\
&= \|Ax' - z'\|_p^p - \|Ax' - \bar{z}'\|_p^p - \|z' - \bar{z}'\|_p^p \\
&\quad - \left(\|SAx' - Sz'\|_p^p - \|SAx' - S\bar{z}'\|_p^p - \|Sz' - S\bar{z}'\|_p^p\right) \\
&\quad - \left(\|SAx' - S\bar{z}'\|_p^p - \|Ax' - \bar{z}'\|_p^p + \|\bar{z}'\|_p^p - \|S\bar{z}'\|_p^p\right) \\
&= O(\epsilon)R^p
\end{aligned}$$

with probability at least $0.99 - O(\delta)$, where the last line follows from Lemmas 3.5.13, 3.5.14 and 3.5.16 (with $x = x'$ and $z = z'$). It then follows that

$$\|A\tilde{x} - z\|_p \le (1 + O(\epsilon))\min_{x\in\mathbb{R}^d}\|Ax - z\|_p$$

with probability at least $0.99 - O(\delta)$. □

We are now ready to prove Theorem 3.4.4.

Proof of Theorem 3.4.4. Recall that we can write $\tilde{A}_1 = S_1 A$ for a sampling matrix S_1 with respect to the online Lewis weights of an oversampling parameter $\beta_1 = \Theta(\epsilon^{-(2+p)} d \log \frac{1}{\epsilon\delta})$. Following the notation of Algorithm 3.1, $\bar{x} = \hat{x}_c + \bar{x}'$ where $\hat{x}_c$ is the constant-factor-solution to $\mathrm{REG}(S_1 A, S_1 z)$ and $\bar{x}'$ is the exact solution to $\mathrm{REG}(S_3 S_1 A, S_3 S_1 z')$, where $z' = z - A\hat{x}_c$. Suppose that $x^*_S = \arg\min_{x\in\mathbb{R}^d} \|S_1 A x - S_1 z'\|_p$. We can verify that with probability at least 0.99,

$$\begin{aligned}
\|S_1 A\bar{x}' - S_1 A x^*_S\|_p &\le 2\|S_3 S_1 A\bar{x}' - S_3 S_1 A x^*_S\|_p \\
&\le 2\|S_3 S_1 A\bar{x}' - S_3 S_1 z'\|_p + 2\|S_3 S_1 A x^*_S - S_3 S_1 z'\|_p \\
&\le 2\|S_3 S_1 A x^*_S - S_3 S_1 z'\|_p + 2\|S_3 S_1 A x^*_S - S_3 S_1 z'\|_p \\
&\le 4C\|S_1 A x^*_S - S_1 z'\|_p,
\end{aligned}$$

where the last line follows from Markov's inequality and that $\mathbb{E}_{S_3} \|S_3 S_1 A x^*_S - S_3 S_1 z'\|_p^p = \|S_1 A x^*_S - S_3 S_1 z'\|_p^p$. Therefore, we can apply Lemma 3.5.16 with $\gamma = 1$, $A = S_1 A$, $b = S_1 z$, $S = S_3$ and $R = 4C\|S_1 A x^*_S - S_1 z\|_p$, which gives us that with probability at least $0.99 - O(\delta)$,

$$\|S_1 A\bar{x} - S_1 z\|_p^p \le (1 + O(\epsilon)) \min_{x\in\mathbb{R}^d} \|S_1 A x - S_1 z\|_p^p. \tag{3.7}$$

Note that the two constant failure probabilities above come from the same event, which is $\|S_3 S_1 A x^*_S - S_3 S_1 z'\|_p^p \lesssim \|S_1 A x^*_S - S_1 z'\|_p^p$.

Using a similar reasoning of bounding $\|S_1 A\bar{x}' - S_1 A x^*_S\|_p^p$, it holds that with probability at least 0.99,

$$\begin{aligned}
\|A\bar{x} - A x^*\|_p &\le 2\|S_1 A\bar{x} - S_1 A x^*\|_p \\
&\le 2\|S_1 A\bar{x} - S_1 z\|_p + 2\|S_1 A x^* - S_1 z\|_p \\
&\le 2(1 + O(\epsilon)) \min_{x\in\mathbb{R}^d} \|S_1 A x - S_1 z\|_p + 2\|S_1 A x^* - S_1 z\|_p \\
&\le 6\|S_1 A x^* - S_1 z\|_p \\
&\le 6C\|A x^* - z\|_p.
\end{aligned}$$

Applying Lemma 3.5.16 with $\gamma = 1$, $A = A$, $z = b - Ax_c$, $S = S_1$ and $R = 6C\|Ax^* - z\|_p$, we can obtain that, with probability at least $0.99 - O(\delta)$,

$$\|A\bar{x} - z\|_p^p \le (1 + O(\epsilon)) \min_{x\in\mathbb{R}^d} \|Ax - z\|_p^p.$$

Still the above two constant failure probability 0.01 come from the same event which is upper bounding $\|S_1 A x^* - S_1 z\|_p$. It then follows immediately that with probability at least $0.98 - O(\delta)$,

$$\|A\tilde{x} - b\|_p^p \le (1 + O(\epsilon)) \min_{x\in\mathbb{R}^d} \|Ax - b\|_p^p,$$

where $\tilde{x} = x_c + \bar{x}$.

For the results above to go through, S_1 should have the oversampling parameter $\beta_1 = \Theta(\frac{d}{\epsilon^{2+p}} \log \frac{1}{\epsilon\delta})$, resulting in

$$N = O\left(\beta_1 \sum_{i=1}^n w_i^{\mathrm{OL}}(A)\right) = O\left(\frac{d^2}{\epsilon^{2+p}} \log \frac{1}{\epsilon\delta} \log n \log \kappa^{\mathrm{OL}}(A)\right)$$

rows of S_1A with probability at least $1-\delta$. Also, S_3 should have an oversampling parameter

$$\beta_3 = \Theta\left(\frac{\log^2 d}{\epsilon^2}\log\frac{1}{\delta}\log N\right) = O\left(\frac{\log^2 d}{\epsilon^2}\log\frac{1}{\delta}\left(\log\frac{d}{\epsilon}+\log\log\frac{1}{\delta}\right)\right),$$

resulting in

$$\begin{aligned} m &= O\left(\beta_3\sum_{i=1}^{n} w_i^{\mathrm{OL}}(SA)\right) \\ &= O\left(\frac{d\log^2 d}{\epsilon^2}\log^2 N\log\frac{n\kappa^{\mathrm{OL}}(A)}{\beta_3\delta}\log\frac{1}{\delta}\right) \\ &= O\left(\frac{d\log^2 d}{\epsilon^2}\left(\log\frac{d}{\epsilon}+\log\log\frac{1}{\delta}\right)^2\log\frac{n\kappa^{\mathrm{OL}}(A)}{\delta}\log\frac{1}{\delta}\right) \end{aligned}$$

rows of S_3S_1A with probability at least $1-\delta$. Here we upper bound $\kappa^{\mathrm{OL}}(SA)$ by Lemma 3.5.11.

The total number of queried labels is dominated by m. Rescaling ϵ and δ gives the claimed result. □

Proof of Lemma 3.5.15

The next lemma is an analogous version to Lemma 7.4 in [16], adapted to the online active regression setting. It reduces the problem from a general matrix A to a matrix A with Lewis weights uniformly bounded by $O(dW/n)$.

Lemma 3.5.17. *Suppose there exists $\ell \geq 1$ such that whenever a matrix $A \in \mathbb{R}^{n\times d}$ has Lewis weights uniformly bounded by $O(dW/n)$, where $W \in \mathbb{R}$ satisfies that $n = \Omega(\frac{d}{\epsilon^2}(W\log^2 d\log n\log\frac{1}{\delta}))$ and $0<\gamma\leq 1$, it holds for all $R>0$ and $\bar{z}\in\mathbb{R}^n$ conforming to (A,ϵ,R) that*

$$\mathop{\mathbb{E}}_{\sigma}\left[\left(\max_{\|Ax\|_p\leq\sqrt{\gamma}R}\left|\sum_{k=1}^{n}\sigma_k\left(\left|a_k^\top x-\bar{z}_k\right|^p-|\bar{z}_k|^p\right)\right|\right)^\ell\right] \leq (\epsilon R^p)^\ell\delta.$$

Then, let $A\in\mathbb{R}^{n\times d}$, $\bar{z}\in\mathbb{R}^n$ be as defined in Lemma 3.5.13 and S be the rescaled sampling matrix with respect to the online Lewis weights of A with oversampling parameter $\beta=\Theta(\frac{1}{\epsilon^2}\log^2 d\log n\log\frac{1}{\delta})$. With probability at least $1-\delta$, it holds that

(i) the number of rows in S is

$$O\left(\frac{d}{\epsilon^2}\log^2 d\log^2 n\log\kappa^{OL}(A)\log\frac{1}{\delta}\right)$$

and

(ii)

$$\max_{\|Ax\|_p\leq\sqrt{\gamma}R}\left|\|SAx-S\bar{z}\|_p^p-\|Ax-\bar{z}\|_p^p-\left(\|S\bar{z}\|_p^p-\|\bar{z}\|_p^p\right)\right|\leq\epsilon R^p.$$

Proof. Note that the sampling probability $p_i = \min\{\beta w_i, 1\}$. If $\beta w_i > 1$, we have $p_i = 1$ and hence $S(Ax - \bar{z})_i = (Ax - \bar{z})_i$. Therefore, we only consider the case $\beta w_i \le 1$. Let

$$M = \left(\max_{\|Ax\|_p \le R} \left| \|SAx - S\bar{z}\|_p^p - \|Ax - \bar{z}\|_p^p - \left(\|S\bar{z}\|_p^p - \|z\|_p^p \right) \right| \right)^\ell.$$

By the standard symmetrization trick, we have

$$\mathop{\mathbb{E}}_{S} M \le 2^\ell \mathop{\mathbb{E}}_{S,\sigma} \left[\left(\max_{\|Ax\|_p \le R} \left| \sum_{k=1}^n \frac{(\mathbb{1}_S)_k}{p_k} \sigma_k \left(\left| a_k^\top x - \bar{z}_k \right|^p - |\bar{z}_k|^p \right) \right| \right)^\ell \right] =: 2^\ell \mathop{\mathbb{E}}_{S,\sigma} M'.$$

It follows from Lemma 1.2.7 that S is a $1/2$-subspace-embedding matrix for A with probability at least $1 - \delta/3$. Furthermore, by Theorem 3.4.2, with probability at least $1 - \delta/3$, the Lewis weights of the rows from SA are within $[\frac{1}{2\beta}, \frac{3}{2\beta}]$. Also, by the Chernoff bounds, SA has $N = \Theta(\beta \sum_i w_i^{\mathrm{OL}}(A))$ rows with probability at least $1 - \delta/3$. Let $\mathcal{E}$ denote the event on S that these three conditions above hold. Then $\Pr(\mathcal{E}) \ge 1 - \delta$.

Next, fix $S \in \mathcal{E}$. Consider the conditional expectation

$$\mathop{\mathbb{E}}_{\sigma} [\, M' |\, S] = \mathop{\mathbb{E}}_{\sigma} \left[\max_{\|Ax\|_p \le R} \left| \sum_{k=1}^N \sigma_k \left(\left| a_k'^\top x - \bar{z}_k' \right|^p - \left| \bar{z}_k' \right|^p \right) \right|^\ell \middle| S \right],$$

where $a_k'^\top x$ and $\bar{z}_k'$ are the k-th coordinates of SAx and $S\bar{z}$.

Recall that S is a $1/2$-subspace-embedding matrix for A when conditioned on $\mathcal{E}$, we have for all $x \in \mathbb{R}^d$ that $\|SAx\|_p \le \frac{3}{2}\|Ax\|_p \le 2R$, which implies that

$$\mathop{\mathbb{E}}_{\sigma} \max_{\|Ax\|_p \le R} \left| \sum_{k=1}^N \sigma_k \left(\left| a_k'^\top x - \bar{z}_k' \right|^p - |\bar{z}_k'|^p \right) \right|^\ell \le \mathop{\mathbb{E}}_{\sigma} \max_{\|SAx\|_p \le 2R} \left| \sum_{k=1}^N \sigma_k \left(\left| a_k'^\top x - \bar{z}_k' \right|^p - |\bar{z}_k'|^p \right) \right|^\ell$$

Now, we verify that SA has small Lewis weights and $S\bar{z}$ conforms to $(SA, \epsilon, 2R)$. First, recall that the Lewis weights of SA are within $[\frac{1}{2\beta}, \frac{3}{2\beta}]$, where $\frac{1}{\beta} = \Theta(\frac{d \log n \log \kappa^{\mathrm{OL}}(A)}{N})$. Hence, $W = \log n \log \kappa^{\mathrm{OL}}(A)$ and $N = \Omega(d\beta \log n \log \kappa^{\mathrm{OL}}(A)) = \Omega(\frac{d}{\epsilon^2} W \log^2 d \log n \log \frac{1}{\delta})$ as desired. Second, the coordinates $|S\bar{z}_k|^p = \frac{|\bar{z}_k|^p}{\beta w_k(A)} \le \frac{R^p}{\epsilon^p \beta} \le 2(\frac{R}{\epsilon})^p w_k(SA) \le (\frac{2R}{\epsilon})^p w_k(SA)$. It then follows from the assumption of the lemma that

$$\mathop{\mathbb{E}}_{\sigma} [\, M' |\, S] \le (\epsilon 2^p R^p)^\ell \, \delta.$$

Rescaling $\epsilon = \epsilon / 2^{p+1}$ and taking expectation over S while conditioned on $\mathcal{E}$, we have that

$$\mathop{\mathbb{E}}_{S,\sigma} [\, M' |\, \mathcal{E}] \le \left(\frac{\epsilon R^p}{2} \right)^\ell \delta.$$

It then follows that

$$\mathop{\mathbb{E}}_{S} [\, M |\, \mathcal{E}] \le (\epsilon R^p)^\ell \, \delta$$

and, by Markov's inequality,

$$\begin{aligned} &\Pr \left\{ \max_{\|Ax\|_p \le R} \left| \|SAx - S\bar{z}\|_p^p - \|Ax - \bar{z}\|_p^p - \left(\|S\bar{z}\|_p^p - \|\bar{z}\|_p^p \right) \right| \ge \epsilon R^p \middle| \mathcal{E} \right\} \\ &\le \Pr \left\{ M \ge (\epsilon R^p)^\ell \middle| \mathcal{E} \right\} \\ &\le \delta. \end{aligned}$$

Rescaling $\delta = \delta/2$ for a union bound completes the proof. □

The next lemma is the generalization of Lemma 3.8 in [59]; here we give the ℓ-th moment bound. It proves that the assumption in Lemma 3.5.17 holds.

Lemma 3.5.18 (Online version of [59, Lemma 3.8])**.** *Let $p \in [1,2], 0 < \gamma \le 1$ and $\epsilon, R > 0$. Suppose that $A \in \mathbb{R}^{n\times d}$ with Lewis weights bounded by $O(dW/n)$, where W satisfies that $n = \Omega(\epsilon^{-2} dW \log^2 d \log n \log(1/\delta))$, and $\bar{z} \in \mathbb{R}^n$ conforms to (A, ϵ, R). Let*

$$\Lambda = \max_{\|Ax\|_p \le \sqrt{\gamma} R} \left| \sum_{k=1}^{n} \sigma_k \left(\left| a_k^\top x - \bar{z}_k \right|^p - |\bar{z}_k|^p \right) \right|,$$

where σ_k's are independent Rademacher variables, then it holds for $\ell = \log(1/\delta)$ that

$$\mathop{\mathbb{E}}_{\sigma} \left[\Lambda^\ell \right] \le (\epsilon R^p)^\ell \delta.$$

Lemma 3.8 in [61] proves that Λ has a subgaussian tail. The ℓ-th moment follows from the property of subgaussian variables.

Proof of Lemma 3.5.18. We will follow the approach in the proof of [61, Lemma 3.8] and only highlight the changes. Their lemma assumes that the Lewis weights are uniformly $O(d/n)$ and we shall modify it to $O(dW/n)$. This upper bound on Lewis weights was used in [59, Equations (4) and (5)]. Define

$$J = \{k : w_k > \epsilon^p d n^{-2}\},$$
$$T = \{y = Ax - \bar{z} : \|Ax\|_p \le R\},$$

where $w_k = w_k^{\mathrm{OL}}(A)$.

In order to bound $\mathbb{E}_\sigma \Lambda^\ell$, the summation over k is split into two parts: indices $k \in J$, associated with large Lewis weights, and indices $k \notin J$, associated with small Lewis weights. The second part (small Lewis weights) is handled as in [59, Lemma 3.11] since our definition of J is identical to theirs. For the first part (large Lewis weights), [59, Equation (4)] becomes, under the change of the upper bound of Lewis weights mentioned above,

$$\mathbb{E} \left(\sup_{y\in T} \left| \sum_{k\in J} \sigma_k w_k \left(|w_k^{-1/p} y_k|^p - |w_k^{-1/p} \bar{z}_k|^p \right) \right| \right)^\ell$$
$$\le \mathbb{E} \left(C_1 \sqrt{\frac{W}{n}} \sup_{y\in T} \left| \sum_{k\in J} \sigma_k (dw_k)^{1/2} \left(|w_k^{-1/p} y_k|^p - |w_k^{-1/p} \bar{z}_k|^p \right) \right| \right)^\ell,$$

where $y = (y_1, y_2, \ldots, y_n)^\top$ and $C_1 > 0$ is an absolute constant. Thus, abusing the notation, we now redefine Λ to be

$$\Lambda = C_1 \sqrt{\frac{W}{n}} \cdot \sup_{y\in T} \left| \sum_{k\in J} \sigma_k (dw_k)^{1/2} \left(|w_k^{-1/p} y_k|^p - |w_k^{-1/p} \bar{z}_k|^p \right) \right|$$

and shall upper bound $\mathbb{E}\,\Lambda^\ell$.

The task is now to study the Rademacher process

$$Z(y) = \sum_{k\in J} \sigma_k (dw_k)^{\frac{1}{2}} |w_k^{-\frac{1}{p}} y_k|^p,$$

where $y = (y_1, y_2, \ldots, y_n)^\top \in T$. Then

$$\Lambda = \sqrt{\frac{W}{n}} \sup_{y \in T} |Z(y) - Z(-\bar{z})|.$$

Define the pseudo-metric $d(x, y)$ for $x, y \in T$ as

$$d(x, y) = \left(\sum_{k \in J} dw_k (|w_k^{-\frac{1}{p}} x_k|^p - |w_k^{-\frac{1}{p}} y_k|^p)^2 \right)^{\frac{1}{2}}.$$

It follows from [59, Lemma 3.17] that $\mathrm{diam}(T) \lesssim \sqrt{d} R^p$ and from the calculation in the proof of [59, Theorem 3.8] that

$$\int_0^\infty \sqrt{\log N(T, d, \epsilon)} d\epsilon \lesssim \sqrt{d} \log d \sqrt{\log n} R^p.$$

The tail-bound version of Dudley's integral then implies that

$$\Pr\left\{ \Lambda \geq C' \sqrt{\frac{d}{n}} \left(\log d \sqrt{W \log n} + z \right) R^p \right\} \leq 2 \exp(-z^2),$$

where $C' > 0$ is an absolute constant. Hence, let $L = \log d \sqrt{W \log n}$ and we have

$$\Pr\left\{ \frac{\Lambda}{CR^p \sqrt{d/n}} - L > z \right\} \leq 2 \exp(-z^2).$$

Applying Proposition 1.2.5 to $\frac{\Lambda}{CR^p \sqrt{d/n}} - L$ yields that

$$\mathop{\mathbb{E}}_{\sigma} \left(\frac{\Lambda}{CR^p \sqrt{d/n}} \right)^\ell \leq 2^\ell (L + K\sqrt{\ell})^\ell.$$

Therefore, we have

$$\mathop{\mathbb{E}}_{\sigma} \Lambda^\ell \leq \left(2C \cdot \frac{L + K\sqrt{\ell}}{\sqrt{n/d}} \right)^\ell (R^p)^\ell \leq \left(\frac{1}{e} \cdot \epsilon \right)^\ell (R^p)^\ell \leq (\epsilon R^p)^\ell \delta,$$

where the second inequality follows from our assumptions that $n \gtrsim (d/\epsilon^2)(L + K\sqrt{\ell})^2$ and that $\ell = \log \frac{1}{\delta}$. □

Finally, Lemma 3.5.15 is now immediate by combining Lemmas 3.5.17 and 3.5.18.

3.5.4 Proof of Theorem 3.4.6

When $p = 2$, the sampling matrices in Algorithm 3.4 do not have independent rows since the online leverage scores are calculated with respect to sampled rows instead of all the rows that have been revealed. As a result, we cannot use a Bernstein bound, which is exactly where the proof of [59, Theorem 4.1] needs to be modified. Comparing with [59, Lemma 3.27], we analyze the sampling process via a martingale because the rows sampled are not independent. Therefore, we shall use Freedman's inequality (Lemma 1.2.3) instead of Bernstein's inequality. This approach was used by [18] for online ℓ_2-regression.

Lemma 3.5.19. *Let $\tilde{w}_i$ be the approximate online Lewis weights of A obtained by Algorithm 3.4. Consider the same setting of A, b and $\bar{z}$ as Lemma 3.5.13. Let S be the rescaled sampling matrix with respect to $p_i = \min\{\beta\tilde{w}_i, 1\}$ and $\beta = \Omega(\frac{1}{\epsilon^4}(d\log\frac{1}{\epsilon} + \log\frac{1}{\delta}))$. With probability at least $1-\delta$, it holds that $|\|SAx - S\bar{z}\|_2^2 - \|Ax - \bar{z}\|_2^2| = O(\epsilon)R^2$ for all $x \in \mathbb{R}^d$ with $\|Ax\|_2 \le R$.*

Proof. We prove the lemma by the Freedman inequality. Let $Y_i = \|(SA)_{[i]}x - (S\bar{z})_{[i]}\|_2^2 - \|A_{[i]}x - \bar{z}_{[i]}\|_2^2$, where $M_{[i]}$ denotes the first i rows of M. Also let $Y_0 = 0$ and $X_i = Y_i - Y_{i-1}$. We claim that $|X_i|$ is uniformly bounded. First, observe that

$$|X_i| = \left| \left\| \frac{(\mathbb{1}_S)_i}{\sqrt{p_i}}(a_i x - \bar{z}_i) \right\|_2^2 - \|a_i x - \bar{z}_i\|_2^2 \right| \le \frac{1}{p_i} \cdot \|a_i x - \bar{z}_i\|_2^2.$$

If $i \in \mathcal{B}$, we have that $\bar{z}_i = 0$ and it follows from Cauchy-Schwarz inequality that $\|a_i x\|_2^2 \le w_i^{\mathrm{OL}}(A)\|Ax\|_2^2 \le w_i^{\mathrm{OL}}(A)R^2$. Otherwise, $\|a_i x - \bar{z}_i\|_2^2 \le (\frac{1}{\epsilon}+1)^2 w_i^{\mathrm{OL}}(A)R^2$. Since $p_i = \min(\beta w_i^{\mathrm{OL}}(A), 1)$, we have $|X_i| \le \frac{4}{\beta\epsilon^2}R^2$, proving the claim.

For brevity of notation, we denote $\mathbb{E}(\cdot|Y_i, \ldots, Y_1)$ by $\mathbb{E}_{i-1}(\cdot)$. Then

$$\begin{aligned}
\mathop{\mathbb{E}}_{i-1} X_i^2 &= \mathbb{E}\left(\left\| \frac{(\mathbb{1}_S)_i}{\sqrt{p_i}}(a_i x - \bar{z}_i) \right\|_2^2 - \|a_i x - \bar{z}_i\|_2^2 \right)^2 \\
&= \mathbb{E}\left(\frac{(\mathbb{1}_S)_i}{p_i} - 1 \right)^2 \|a_i x - \bar{z}_i\|_2^4 \\
&= \left(\frac{1}{p_i} - 1 \right) \|a_i x - \bar{z}_i\|_2^4 \\
&\le \frac{w_i^{\mathrm{OL}}(A)}{p_i} \left(\frac{1}{\epsilon} + 1 \right)^2 R^2 \|a_i x - \bar{z}_i\|_2^2 \\
&\le \frac{4}{\beta\epsilon^2} R^2 \|a_i x - \bar{z}_i\|_2^2.
\end{aligned}$$

Therefore, $\sum_{i=1}^n \mathbb{E}_{i-1} X_i^2 \le \frac{4}{\beta\epsilon^2}R^2 \sum_{i=1}^n \|a_i x - \bar{z}_i\|_2^2$. Since $\|Ax\|_2^2 \le R^2$ and $\|\bar{z}\|_2^2 = O(R^2)$, we obtain that $\sum_{i=1}^n \mathbb{E}_{i-1} X_i^2 \le \frac{1}{\beta\epsilon^2}O(R^4)$.

It follows from the Freedman inequality and $\beta \ge \frac{2}{\epsilon^4}(d\log\frac{3}{\epsilon} + \log\frac{1}{\delta})$ that

$$\Pr(|Y_n| \ge C\epsilon R^2) \le \exp\left(\frac{-C^2\epsilon^2 R^4}{\frac{1}{\beta\epsilon^2}O(R^4) + \frac{O(R^4)}{3\beta\epsilon}} \right) \le \exp\left(\frac{-\beta\epsilon^4}{2} \right) \le \left(\frac{\epsilon}{3}\right)^d \delta$$

for a constant C large enough. This implies that for a fixed $x \in \mathbb{R}^d$ such that $\|Ax\|_2 \le R$,

$$\left| \|SAx - S\bar{z}\|_2^2 - \|Ax - \bar{z}\|_2^2 \right| \le O(\epsilon)R^2$$

with probability at least $1 - (\frac{\epsilon}{3})^d\delta$.

Next, we make a net argument. Assume, without loss of generality, that $R = 1$. Consider the ϵ-net $\mathcal{N}$ of the ellipsoid $\mathbf{B} = \{x \in \mathbb{R}^d : \|Ax\|_2 \le 1\}$ endowed with distance $d(x,y) = \|A(x-y)\|_2$. We can choose $\mathcal{N}$ with at most $(3/\epsilon)^d$ points. After applying a union bound over the net, we have that $|\|SAx - S\bar{z}\|_2^2 - \|Ax - \bar{z}\|_2^2| = O(\epsilon)$ holds for all $x \in \mathcal{N}$ simultaneously with probability at least $1-\delta$.

For any $x \in \mathbf{B}$, there exists $y \in \mathcal{N}$ such that $\|Ax - Ay\|_2 \le \epsilon$. According to Lemma 3.5.2, when $\beta = \Omega(\log\frac{d}{\delta})$, we have that $\frac{1}{2}\|Ax\|_2^2 \le \|SAx\|_2^2 \le \frac{3}{2}\|Ax\|_2^2$ for all x

simultaneously with probability at least $1-\delta$, so it holds that $\|SAx - SAy\|_2 \le \sqrt{\frac{3}{2}}\epsilon$. Hence, by the triangle inequality we have

$$\begin{aligned}
&\left|\|SAx - S\bar{z}\|_2^2 - \|Ax - \bar{z}\|_2^2\right| \\
&= \left|\|SAx - SAy + SAy - S\bar{z}\|_2^2 + \|Ax - \bar{z}\|_2^2\right| \\
&\le \left|\|SAx - SAy\|_2^2 + \|SAy - S\bar{z}\|_2^2 + 2\langle SAx - SAy, SAy - S\bar{z}\rangle - \|Ax - \bar{z}\|_2^2\right| \\
&\le \left|\|SAy - S\bar{z}\|_2^2 - \|Ax - Ay + Ay - \bar{z}\|_2^2\right| + \|SAx - SAy\|_2^2 \\
&\qquad + 2\|SAx - SAy\|_2 \|SAy - S\bar{z}\|_2 \\
&\le \left|\|SAy - S\bar{z}\|_2^2 - \|Ay - \bar{z}\|_2^2\right| + \|SAx - SAy\|_2^2 + \|Ax - Ay\|_2^2 \\
&\qquad + 2\|SAx - SAy\|_2 \|SAy - S\bar{z}\|_2 + 2\|Ax - Ay\|_2 \|Ay - \bar{z}\|_2 \\
&\le O(\epsilon) + \frac{3}{2}\epsilon^2 + \epsilon^2 + 2\sqrt{\frac{3}{2}}\epsilon(\|Ay - \bar{z}\|_2 + \sqrt{\epsilon}) + 2\epsilon \cdot \|Ay - \bar{z}\|_2 \\
&\le O(\epsilon) + 2\epsilon \cdot (2\sqrt{6} + \sqrt{\epsilon} + 2) \\
&= O(\epsilon)
\end{aligned}$$

for all x such that $\|Ax\|_2 \le 1$, where we have used that $\|SAy - S\bar{z}\|_2^2 \le \|Ay - \bar{z}\|_2^2 + \epsilon \le (\|Ay - \bar{z}\|_2 + \sqrt{\epsilon})^2$ and $\|Ay - \bar{z}\|_2 \le \|Ay\|_2 + \|\bar{z}\|_2 \le 2$. □

3.5.5 Time Complexity for $p = 2$

Lemma 3.5.20. *With probability at least $1-\delta$, the running time of Algorithm 3.4 over n iterations is $O(\mathrm{nnz}(A)\log\frac{n}{\delta} + \frac{d^3}{\epsilon^4}\log\frac{\|A\|_2}{\sigma}\log\frac{1}{\epsilon\delta}(\log\frac{n}{\delta} + d))$.*

Proof. We analyze the time complexity following Lemma 3.8 in [40]. Note that total runtime is dominated by calls to UPDATE. The approximate Lewis weights are calculated by $\tilde{w}_t = \|H^{(t-1)}a_t\|_2^2$, which takes $O(\mathrm{nnz}(A)\log\frac{n}{\delta})$ time over n iterations. Observe that the runtime of each call to UPDATE is dominated by the time to compute $F^{(t)}$ and $H^{(t)}$, which takes $O(d\log\frac{n}{\delta} + d^2)$ time. Calls to UPDATE only happen when there is a new row a_t is sampled and the number of samples is dominated by the maximum of the number of rows of S and that of S_1, which with probability at least $1-\delta$ are $O(d\log d)$ and $O(\frac{d^2}{\epsilon^4}\log\frac{1}{\epsilon\delta}\log n\log\frac{\|A\|_2}{\sigma})$, respectively. Hence, the total running time is $O(\mathrm{nnz}(A)\log\frac{n}{\delta} + \frac{d^4}{\epsilon^4}\log\frac{1}{\epsilon\delta}\log n\log\frac{\|A\|_2}{\sigma} + \frac{d^3}{\epsilon^4}\log\frac{n}{\delta}\log\frac{1}{\epsilon\delta}\log n\log\frac{\|A\|_2}{\sigma})$. □

3.6 Optimal dependence on ϵ

The query complexity in Theorem 3.4.4 has a quadratic dependence on $1/\epsilon$. In this section, we shall improve it to $1/\epsilon$, which is the optimal for active regression [59] and is thus optimal for online active regression. Again, we follow the idea in [59], narrowing the region of x from $\{x : \|Ax\|_p \le R\}$ to $\{x : \|Ax\|_p \le \sqrt{\gamma}R\}$, where $0 < \gamma < 1$. This will guarantee that $\|SAx - S\bar{z}\|_p$ is close to $\|Ax - \bar{z}\|_p$ for all x near $x^* = \mathrm{argmin}_x \|Ax - \bar{z}\|_p$, which is enough for approximately solving the original regression problem $\min_x \|Ax - b\|_p$.

Proof of Sufficiency for Considering a Narrowed Region

Now, we explain why it suffices to consider only x satisfying that $\|Ax\|_p \le \sqrt{\gamma}R$. First, we prove all good approximate solutions are near to the optimal solution.

Lemma 3.6.1. *Let $A \in \mathbb{R}^{n\times d}$, $z \in \mathbb{R}^n$ and $0 < \gamma < 1$. We assume that $\|z\|_p \le R$ where $R = \min_{x\in\mathbb{R}^d} \|Ax - z\|_p$. Let $x^* = \arg\min_{x\in\mathbb{R}^d} \|Ax - z\|_p$. If $x \in \mathbb{R}^d$ satisfies $\|Ax - z\|_p \le (1 + c\gamma)R$, we have that $\|Ax^* - Ax\|_p \le \sqrt{\gamma}R$, where $c \in (0, 1]$ is an absolute constant.*

Proof. The same statement is proved for matrices with bounded Lewis weights in [61, Theorem 3.19]. Now we prove for a general matrix $A \in \mathbb{R}^{n\times d}$. Let $w_i(A)$ be the Lewis weight of row a_i and $k_i = \lceil \frac{w_i(A)}{d/n} \rceil$. We replace each row a_i by k_i copies of $\frac{a_i}{k_i^{1/p}}$, obtaining a new matrix A'. Then we have

$$\frac{a_i^\top}{k_i^{1/p}} \left(\sum_{i=1}^n k_i \cdot (\frac{w_i}{k_i})^{1-\frac{2}{p}} \frac{a_i}{k_i^{1/p}} \frac{a_i^\top}{k_i^{1/p}} \right)^{\frac{p}{2}} \frac{a_i}{k_i^{1/p}} = \frac{a_i^\top}{k_i^{1/p}} \left(\sum_{i=1}^n w_i^{1-\frac{2}{p}} a_i a_i^\top \right)^{\frac{p}{2}} \frac{a_i}{k_i^{1/p}} = \frac{w_i}{k_i}.$$

Therefore, the Lewis weight $w_i(A')$ is bounded by $\frac{d}{n}$. It is also clear that $\|Ax\|_p = \|A'x\|_p$. We also split every entry of z into k_i copies, obtaining z'. Thus we have $\|Ax - z\|_p = \|A'x - z'\|_p$. Note that $\arg\min_{x\in\mathbb{R}^d} \|Ax - z\|_p = \arg\min_{x\in\mathbb{R}^d} \|A'x - z'\|_p$. Hence we can use [61, Theorem 3.19] to get $\|Ax^* - Ax\|_p = \|A'x^* - A'x\|_p \le O(\sqrt{\gamma}R)$. This completes the proof. □

Theorem 3.6.2. *Let $A \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$. Suppose that S is a rescaled sampling matrix according to $w_i^{OL}(A)$ with oversampling factor $\beta = \Theta(\frac{\log^2 d}{\epsilon} \log n \log \frac{1}{\delta})$ and S' is a rescale dsampling matrix according to $w_i^{OL}(A)$ with oversampling factor $\beta' = \Theta(\log d)$. Let $x_c = \arg\min_{x\in\mathbb{R}^d} \|S'Ax - S'b\|_p$, $z = b - Ax_c$, $\hat{x} = \arg\min_{x\in\mathbb{R}^d} \|SAx - Sz\|_p$ and $\tilde{x} = x_c + \hat{x}$. It holds that*

$$\|A\tilde{x} - z\|_p \le (1 + \epsilon) \min_{x\in\mathbb{R}^d} \|Ax - z\|_p$$

with probability at least $0.99 - \delta$ and the query complexity is

$$O\left(\frac{d}{\epsilon} \log^2 d \log^2 n \log \kappa^{OL}(A) \log \frac{1}{\delta} \right)$$

for $1 < p < 2$ and

$$O\left(\frac{d}{\epsilon} \log^2 d \log^2 n \log \frac{\|A\|_2}{\sigma} \log \frac{1}{\delta} \right)$$

for $p = 2$.

Proof. Let $c \in (0, 1]$ be the same absolute constant in Lemma 3.6.1 and

$$K = \begin{cases} \Theta(\frac{d}{c^2} \log^2 d \log^2 n \log \kappa^{\mathrm{OL}}(A) \log \frac{1}{\delta}), & 1 < p < 2 \\ \Theta(\frac{d}{c^2} \log^2 d \log^2 n \log \frac{\|A\|_2}{\sigma} \log \frac{1}{\delta}), & p = 2. \end{cases}$$

When the query complexity is $K\epsilon$, it follows from Lemma 3.5.15 that $\|A\tilde{x} - b\|_p \le (1 + c\sqrt{\epsilon})R$ with probability at least $0.99 - \delta$. Let $x^* = \arg\min_{x\in\mathbb{R}^d} \|Ax - z\|_p$ and $\gamma = c\sqrt{\epsilon}$, then $\|A\bar{x} - Ax^*\|_p \le \sqrt{\gamma}R$ by Lemma 3.6.1.

Let ϵ_1 be such that $K/\epsilon = \gamma K/\epsilon_1^2$, thus $\epsilon_1 = c^{1/2}\epsilon^{3/4}$. By Lemma 3.5.16, we have $\|A\tilde{x} - b\|_p \le (1 + c\epsilon_1)R$ with probability at least $1 - O(\delta)$. Note that here the failure probability is not $0.01 + O(\delta)$ because we have already assumed that $\|Sz\|_p^p = O(\|z\|_p^p)$ with probability at least 0.995 and $\|z\|_p^p \le R^p$ with probability at least 0.995.

Using Lemma 3.6.1, we can get that $\|A\tilde{x} - Ax^*\|_p \le \sqrt{\epsilon_1}R$. Now, taking $\gamma = \epsilon_1$ in Lemma 3.5.15, we see that, with probability at least $1 - O(\delta)$, $\|A\tilde{x} - b\|_p \le (1 + \epsilon_2)R$ for

ϵ_2 such that $\epsilon_1 K/\epsilon_2^2 = K/\epsilon$, i.e, $\epsilon_2 = c^{1/4}\epsilon^{7/8}$. Repeating this process, we can obtain that $\|A\tilde{x} - b\|_p \le (1+\epsilon_i)R$ with probability at least $1 - O(i \cdot \delta)$, where $\epsilon_i^2 = \epsilon_{i-1}\epsilon$. We can solve that $\epsilon_i = c^{\frac{1}{2^i}}\epsilon^{1-\frac{1}{2^{i+1}}}$. Letting $i = \log\log(1/\epsilon)$ yields $\epsilon_i \le 2\epsilon$, that is, $\|A\tilde{x} - b\|_p \le (1+2\epsilon)R$ with probability at least $1 - O(\delta \log\log(1/\epsilon))$. Rescaling $\epsilon = \epsilon/2$ and $\delta = \Theta(\delta/\log\log(1/\epsilon))$ completes the proof. □

Theorem 3.6.3 (Main results)**.** *Let $A \in \mathbb{R}^{n\times d}$ and $b \in \mathbb{R}^n$.*

Algorithm 3.1 modified as in Theorem 3.4.5 outputs a solution $\tilde{x}$ which satisfies that

$$\|A\tilde{x} - b\|_p \le (1+\epsilon)\min_{x\in\mathbb{R}^d}\|Ax - b\|_p \tag{3.8}$$

with probability at least $1-\delta$ and makes

$$O\left(\frac{d}{\epsilon}\log^2 d \log^2\frac{d}{\delta\epsilon}\cdot\log\frac{n\kappa^{OL}(A)}{\delta}\log^2\frac{1}{\delta}\right)$$

queries overall in total.

Furthermore, with probability at least $1-\delta$, it uses $O(md)$ words of space in total.

Algorithm 3.4 modified by the sampling scheme in Theorem 3.4.5 makes

$$m = O\left(\frac{d}{\epsilon}\log^2 d\log^2\frac{d}{\delta\epsilon}\cdot\log\left(n\frac{\|A\|_2}{\sigma\delta}\right)\log^2\frac{1}{\delta}\right)$$

queries in total and maintains for each $T = d+1,\dots,n$ a solution $\tilde{x}^{(T)}$ which satisfies that

$$\left\|A^{(T)}\tilde{x}^{(T)} - b^{(T)}\right\|_2 \le (1+\epsilon)\min_{x\in\mathbb{R}^d}\left\|A^{(T)}x - b^{(T)}\right\|_2.$$

Furthermore, with probability at least $1-\delta$, it uses $O(md)$ words of space in total.

Proof. We shall only prove for the case $1 < p < 2$ below, as the case $p = 2$ follows the same approach with a different sum of online Lewis weights. Using the boosting procedure explained prior to Theorem 3.4.5, we can obtain a constant factor approximation with probability at least $1-\delta$.

The proof is similar to that of Theorem 3.4.4. We can write $\tilde{A}_1 = S_1 A$ for a sampling matrix S_1 with respect to the online Lewis weights. It follows from Theorem 3.6.2 that with probability at least $0.995 - \delta$,

$$\|S_1 A\hat{x} - S_1 z\|_p^p \le (1+\epsilon)\min_{x\in\mathbb{R}^d}\|S_1 Ax - S_1 z\|_p^p,$$

where $z = b - Ax_c$. Following the same approach in the proof of Theorem 3.4.4, we can obtain that, with probability at least $0.995 - O(\delta)$,

$$\|A\tilde{x} - b\|_p^p \le (1+O(\epsilon))\min_{x\in\mathbb{R}^d}\|Ax - b\|_p^p.$$

For the results above to go through, S_1 should have oversampling parameter $\beta_1 = \Theta(\frac{d}{\delta^{2+p}\epsilon^{2+p}}\log\frac{1}{\epsilon\delta})$, resulting in

$$N = O\left(\beta_1\sum_{i=1}^n w_i^{\mathrm{OL}}(A)\right) = O\left(\frac{d^2}{\delta^{2+p}\epsilon^{2+p}}\log\frac{1}{\epsilon\delta}\log n\log\kappa^{\mathrm{OL}}(A)\right)$$

rows of S_1A with probability at least $1-\delta$. The $\frac{1}{\delta^{2+p}}$ term in β_1 is for bounding $\|S_1z\|_p^p = O(\|z\|_p^p)$ with probability at least $1-\delta$. Also, S_3 should have an oversampling parameter

$$\beta_3 = \Theta\left(\frac{\log^2 d}{\epsilon}\log\frac{1}{\delta}\log N\right) = O\left(\frac{\log^2 d}{\epsilon}\log\frac{d}{\epsilon\delta}\log\frac{1}{\delta}\right),$$

resulting in

$$m = O\left(\beta_3\sum_{i=1}^{n} w_i^{\mathrm{OL}}(SA)\right) = O\left(\frac{d\log^2 d}{\epsilon}\log^2\frac{d}{\epsilon\delta}\log\frac{n\kappa^{\mathrm{OL}}(A)}{\delta}\log\frac{1}{\delta}\right)$$

rows of S_3S_1A with probability at least $1-\delta$. Here we upper bound $\kappa^{\mathrm{OL}}(SA)$ by Lemma 3.5.11. Now the constant failure probability 0.005 is from bounding $\|S_3(S_1z - S_1A\hat{x}_c)\|_p^p = O(\|z - S_1A\hat{x}_c\|_p^p)$. In order to obtain $1-\delta$ success probability, we use the same boosting method for Theorem 3.4.5 and sample $\log\frac{1}{\delta}$ independent copies of S_3, which results in

$$m = O\left(\frac{d\log^2 d}{\epsilon}\log^2\frac{d}{\epsilon\delta}\log\frac{n\kappa^{\mathrm{OL}}(A)}{\delta}\log^2\frac{1}{\delta}\right).$$

The total number of queried labels is dominated by m. Rescaling ϵ and δ gives the claimed result. □

Algorithm 3.4 Online Active Regression for $p = 2$

Initialize: Let $\tilde{A}^{(d)}, \tilde{A}_1^{(d)}, \tilde{A}_2^{(d)}, \tilde{A}_3^{(d)}$ be the first d rows of A and $\tilde{b}^{(d)}, \tilde{b}_2^{(d)}, \tilde{b}_3^{(d)}$ be the first d rows of b. Let $x_c^{(d)} = \textsc{Reg}(\tilde{A}^{(d)}, \tilde{b}^{(d)}, 2)$, $\tilde{z}_2^{(d)} = \tilde{z}_3^{(d)} = \tilde{b}^{(d)} - \tilde{A}^{(d)} x_c^{(d)}$, $\hat{x}_c^{(d)} = \textsc{Reg}(\tilde{A}_2^{(d)}, \tilde{z}_2^{(d)}, 2)$ and $\bar{x}'_d = \textsc{Reg}(\tilde{A}_3^{(d)}, \tilde{z}_3^{(d)} - \tilde{A}_3^{(d)} \hat{x}_c^{(d)}, 2)$. Let $\mathring{G}^{(d)} = ((\tilde{A}^{(d)})^\top \tilde{A}^{(d)})^{-1}$ and $H^{(d)} = \tilde{A}^{(d)} \mathring{G}^{(d)}$. Also let $\mathring{G}_i^{(d)} = ((\tilde{A}_i^{(d)})^\top \tilde{A}_i^{(d)})^{-1}$ and $H_i^{(d)} = \tilde{A}_i^{(d)} \mathring{G}_i^{(d)}$ for $i = 1, 2, 3$. Let $J^{(d)} \in \mathbb{R}^{O(\log \frac{n}{\delta}) \times d}$ be a constant-factor approximation JL matrix.

$\beta \leftarrow \Theta(\log d)$
$\beta_1 \leftarrow \Theta((d \log(1/\epsilon) + \log(1/\delta))/\epsilon^4)$
$\beta_2 \leftarrow \Theta(\log d)$
$\beta_3 \leftarrow \Theta((\log^2 d) \log(d/\epsilon) \log(1/\delta)/\epsilon^2)$
retain the first d rows of A
while there is an additional row a_t **do**
    $\tilde{w}_t \leftarrow \|H^{(t-1)} a_t\|_2^2$
    $(x_c^{(t)}, \tilde{A}^{(t)}, \tilde{b}^{(t)}, \mathring{G}^{(t)}, H^{(t)})$
        $\leftarrow \textsc{SampleQuery}(a_t, \tilde{b}^{(t-1)}, \perp, \perp, \tilde{A}^{(t-1)}, \beta, \tilde{w}_t, \mathring{G}^{(t-1)}, 1)$
    $\tilde{w}_{1,t} \leftarrow \|H_1^{(t)} a_t\|_2^2$
    $p_{1,t} \leftarrow \min\{\beta_1 \tilde{w}_{1,t}, 1\}$
    Sample a_t with probability $p_{1,t}$
    if a_t is sampled **then**
        $\tilde{A}_1^{(t)} \leftarrow \tilde{A}_1^{(t-1)} \circ \frac{a_t^\top}{\sqrt{p_{1,t}}}$
        $(\mathring{G}_1^{(t)}, H_1^{(t)}) \leftarrow \textsc{Update}(\frac{a_t}{\sqrt{p_{1,t}}}, \perp, \perp, \tilde{A}_1^{(t-1)}, \mathring{G}_1^{(t-1)})$
        $\tilde{w}_{2,t} \leftarrow \|H_2^{(t)} \frac{a_t}{\sqrt{p_{1,t}}}\|_2^2$
        $(\hat{x}_c^{(t)}, \tilde{A}_2^{(t)}, \tilde{b}_2^{(t)}, \mathring{G}_2^{(t)}, H_2^{(t)})$
            $\leftarrow \textsc{SampleQuery}(\frac{a_t}{\sqrt{p_{1,t}}}, \tilde{b}_2^{(t-1)}, x_c^{(t)}, \perp, \tilde{A}_2^{(t-1)}, \beta_2, \tilde{w}_{2,t}, \mathring{G}_2^{(t-1)}, 2)$
        $\tilde{w}_{3,t} = \|H_3^{(t)} \frac{a_t}{\sqrt{p_{1,t}}}\|_2^2$
        $(\bar{x}'^{(t)}, \tilde{A}_3^{(t)}, \tilde{b}_3^{(t)}, \mathring{G}_3^{(t)}, H_3^{(t)})$
            $\leftarrow \textsc{SampleQuery}(\frac{a_t}{\sqrt{p_{1,t}}}, \tilde{b}_3^{(t-1)}, x_c^{(t)}, \hat{x}_c^{(t)}, \tilde{A}_3^{(t-1)}, \beta_3, \tilde{w}_{3,t}, \mathring{G}_3^{(t-1)}, 3)$
    end if
    $\bar{x}^{(t)} \leftarrow \hat{x}_c^{(t)} + \bar{x}'^{(t)}$
    $\tilde{x}^{(t)} \leftarrow \bar{x}^{(t)} + x_c^{(t)}$
end while
return $\tilde{x}^{(t)}$

Algorithm 3.5 Subroutine SAMPLEQUERY in Algorithm 3.4

procedure SAMPLEQUERY($a_t, \tilde{b}^{(t-1)}, x_c^{(t)}, \hat{x}_c^{(t)}, \tilde{A}^{(t-1)}, \beta, \tilde{w}_t, \mathring{G}^{(t-1)}, \chi$)
  $p_t \leftarrow \min\{\beta \tilde{w}_t, 1\}$
  Sample a_t with probability p_t
  if a_t is sampled **then**
    $\tilde{A}^{(t)} \leftarrow \tilde{A}^{(t-1)} \circ \frac{a_t^\top}{\sqrt{p_t}}$
    Query b_t
    if $\chi = 1$ **then**
      $\tilde{b}^{(t)} \leftarrow \tilde{b}^{(t-1)} \circ \frac{b_t}{\sqrt{p_t}}$
    else
      $b^{(t)} \leftarrow b^{(t-1)} \circ \frac{b_t}{\sqrt{p_{1,t} p_t}}$
      $z^{(t)} \leftarrow b^{(t)} - \tilde{A}^{(t)} x_c^{(t)}$
    end if
    $(x^{(t)}, \mathring{G}^{(t)}, H^{(t)}) \leftarrow$ UPDATE$(a_t, \tilde{b}^{(t)}, \hat{x}_c^{(t)}, \tilde{A}^{(t)}, \mathring{G}^{(t-1)})$
  else
    $(\tilde{A}^{(t)}, \tilde{b}^{(t)}) \leftarrow (\tilde{A}^{(t-1)}, \tilde{b}^{(t-1)})$
    $(x^{(t)}, \mathring{G}^{(t)}, H^{(t)}) \leftarrow (x^{(t-1)}, \mathring{G}^{(t-1)}, H^{(t-1)})$
  end if
  return $(x^{(t)}, \tilde{A}^{(t)}, \tilde{b}^{(t)}, \mathring{G}^{(t)}, H^{(t)})$
end procedure

Algorithm 3.6 Subroutine UPDATE in Algorithm 3.5

procedure UPDATE($a_t, \tilde{b}^{(t)}, \hat{x}_c^{(t)}, \tilde{A}^{(t)}, \mathring{G}^{(t-1)}$)
  $g \leftarrow a_t^\top \mathring{G}^{(t-1)} a_t / p_t$
  $\mathring{G}^{(t)} \leftarrow \mathring{G}^{(t-1)} - \frac{1}{1+g} \mathring{G}^{(t-1)} \frac{a_t a_t^\top}{p_t} \mathring{G}^{(t-1)}$
  $J^{(t)} \leftarrow$ updated JL matrix after adding a new independent column
  $F^{(t)} \leftarrow J^{(t)} \tilde{A}^{(t)}$
  $H^{(t)} \leftarrow F^{(t)} \mathring{G}^{(t)}$
  if $\tilde{b}^{(t)} = \perp$ **then**
    return $(\mathring{G}^{(t)}, H^{(t)})$
  else if $\hat{x}_c^{(t)} = \perp$ **then**
    $x^{(t)} \leftarrow \mathring{G}^{(t)} \tilde{A}^{(t)\top} \tilde{b}^{(t)}$
  else
    $x^{(t)} \leftarrow \mathring{G}^{(t)} \tilde{A}^{(t)\top} (\tilde{b}^{(t)} - \tilde{A}^{(t)} \hat{x}_c^{(t)})$
  end if
  return $(x^{(t)}, \mathring{G}^{(t)}, H^{(t)})$
end procedure

Algorithm 3.7 Online Active Regression for $p = 1$

Initialize: Let $\tilde{A}^{(d)}$ be the first d rows of A and $\tilde{b}^{(d)}$ be the first d rows of b.

$\beta \leftarrow \Theta(\log d)$
retain the first d rows of A
while there is an additional row a_t **do**
    $\tilde{w}_t \leftarrow w_{\text{last}}(\tilde{A}^{(t)})$
    $p_t \leftarrow \min(\beta \tilde{w}_t, 1)$
    $(\tilde{A}^{(t)}, \tilde{b}^{(t)}) \leftarrow \textsc{Sample}(a_t, p_t, \tilde{A}^{(t-1)}, \tilde{b}^{(t-1)}, 1)$
    $\tilde{x^t} \leftarrow \textsc{Reg}(\tilde{A}^{(t)}, \tilde{b}^{(t)}, 1)$
end while
return $\tilde{x}$

Chapter 4

Active Regression with Shared Labels

4.1 Introduction

The rapid advancements in deep learning have led to a substantial increase in demand for extensive labelled data points to train models. However, data labelling remains costly due to its reliance on human labour. To address this challenge, active learning (AL) [69, 64] has emerged as an effective strategy for mitigating the expenses. The specific setting of AL for linear regression is introduced in Section 3.1.

Recently, there has been a significant surge in the demand for machine learning on diverse resource-constrained devices such as in speech recognition and face recognition systems [21, 35, 57]. These systems usually need to support a variety of machines with different computing and memory resources. As a result, the task of training multiple models with shared labels has arisen [10], leading to a new setting of AL where there are multiple target models to be learned simultaneously [73]. The traditional approach involves iteratively querying labels and updating models, which can be particularly expensive when dealing with multiple target models, as each query iteration requires training multiple deep models. A more cost-effective strategy would be one-shot querying, where all label queries are made in a single iteration without retraining the models. Most existing one-shot AL methods query a representative set of instances using the distance between feature vectors [84, 78, 41, 70]. However, this approach faces challenges when handling multiple models, as the same instance can exhibit different feature representations in different models.

In this chapter, we propose a one-shot AL method for multiple deep models, accompanied by a rigorous analysis. Our method is based on the fact that a deep model can be viewed as a linear prediction layer (i.e., multiple neuron models) and a nonlinear feature extractor (i.e., the network backbone). Therefore, training multiple deep models can be described as learning linear prediction layers from the outputs of distinct network backbones. In this way, active learning from diverse data representations can be formulated as optimizing a shared sampling matrix to minimize the error of each linear predictor. To facilitate computing and analysis, we consider the learning of the prediction layer as an ℓ_p-regression problem with $p \geq 1$. Specifically, suppose that there are k models and $A^j \in \mathbb{R}^{n\times d}$ $(j = 1, \ldots, k)$ is the feature matrix obtained by feeding the dataset into the j-th network backbone. Let $f : \mathbb{R} \to \mathbb{R}$ be an L-Lipschitz function with $f(0) = 0$. Typical choices of f are activation functions such as ReLU, Sigmoid, and so on. We abuse the notation and apply f to a vector $v \in \mathbb{R}^n$ coordinatewise, i.e. $f(v) = (f(v_1), \ldots, f(v_n))^\top$. Suppose that $b^1, \ldots, b^c \in \mathbb{R}^n$ are c label vectors and the task is to minimize the loss $\sum_{i=1}^{c} \|f(A^j x^{ij}) - b^i\|_p^p$ over $x^{1j}, \ldots, x^{cj} \in \mathbb{R}^d$ for all models j simultaneously. Since the

construction of S is independent of $b^1, \ldots, b^c$, we henceforth assume that $c = 1$, with a single label vector $b \in \mathbb{R}^n$. Therefore, we seek a shared reweighted sampling matrix S such that we can, from the labels of the sampled instances Sb, approximately solve the regression problem $\min_x \|f(A^j x) - b\|_p^p$ for all models j simultaneously.

The simplest case is when there is a single model, i.e., $k = 1$. In this case, Gajjar et al. [31] are the first to study the problem of actively learning a single neuron model. They cast the problem as a least-squares regression problem (i.e. $p = 2$) $\min_x \|f(Ax) - b\|_2^2$ and find an $\tilde{x}$ such that

$$\|f(A\tilde{x}^j) - b\|_2^2 \leq C \cdot \big(\|f(Ax^*) - b\|_2^2 + \epsilon L^2 \|Ax^*\|_2^2\big),$$

where $x^* = \arg\min_x \|f(Ax) - b\|_2^2$ is the minimizer, C is an absolute constant and ϵ is an accuracy parameter. Recall that L is the Lipschitz constant of f. Gajjar et al. [31] also show that an additive error term to $\|f(Ax^*) - b\|_2^2$ is necessary for poly(d) queries. For $k > 1$ and general p, we seek approximate solutions $\tilde{x}_1, \ldots, \tilde{x}_k$ with the following error guarantee of a similar form on each individual model:

$$\|f(A^j\tilde{x}^j) - y\|_p^p \leq C \cdot \big(\|f(A^j x^j) - y\|_p^p + \epsilon L^p \|A^j x^j\|_p^p\big), \tag{4.1}$$

where $x^j = \arg\min_x \|f(A^j x) - y\|_p^p$ is the minimizer for model j and $C = C(p) > 0$ is a constant depending only on p.

Gajjar et al. [31] construct S to be a leverage score sampling matrix and solve $\tilde{x} = \arg\min_{x \in E} \|f(SAx) - Sy\|_2^2$ with $E = \{x : \|SAx\|_2^2 \leq \|Sy\|_2^2/(\epsilon L^2)\}$. At the core of their argument lies the classical fact that such an S gives an ℓ_2 subspace embedding for A, i.e., $\|SAx\|_2 \approx \|Ax\|_2$ for all x simultaneously. In fact, it is not necessary to sample the rows of A according to the exact leverage scores $\tau_1(A), \ldots, \tau_n(A)$; any sampling probability proportional to $t_i \gtrsim \tau_i(A)$ for i-th row will suffice, with the number of samples being proportional to $\sum_i t_i$. This very fact motivates us to tackle the task of data selection from diverse representations by sampling the rows according to the maximum of leverage scores across A^j's, i.e., letting $t_i \sim \max_j \tau_i(A^j)$. Solving for each model j by $\tilde{x}^j = \arg\min_{x \in E^j} \|f(SA^j x) - Sy\|_2^2$ with $E^j = \{x : \|SA^j x\|_2^2 \leq \|Sy\|_2^2/(\epsilon L^2)\}$ will then achieve (4.1) for $p = 2$. This indicates that the queried instances are effective in learning each of the linear predictors, which fits our problem well. A potential caveat is that the number of samples needed will be proportional to $\sum_i t_i \sim \sum_i \max_j \tau_i(A^j)$, which could be as large as kd. However, empirical studies show that this is not the case for real-world datasets and our approach will thus be efficient. For details, see [37].

For general p, instead of leverage scores, it is natural to consider Lewis weights, which can be seen as generalizations of leverage scores for general p. It is known that an ℓ_p Lewis weight sampling matrix S gives an ℓ_p subspace embedding, i.e., $\|SAx\|_p \approx \|Ax\|_p$ for all x simultaneously [16]. The approach mentioned above extends to general p naturally, attaining (4.1) for general p, by sampling according to the maximum Lewis weights and solving an ℓ_p-regression problem for $\tilde{x}^j$ with an ℓ_p-version of E^j.

Acknowledgment of Experimental Work. It is important to note that the experimental work, which includes the detailed procedures, data collection and initial analysis, was all conducted by my co-author Ying-Peng Tang. As such, I shall not delve into the specifics of these experiments within this project. Interested readers can refer to Huang et al. [37] for experiment details.

Algorithm 4.1 Algorithm for multiple regression problems with shared labels

Require: Matrices $A^1, \ldots, A^k \in \mathbb{R}^{n\times d}$, a query access to the entries of b, number of samples m, accuracy parameter ϵ

Compute Lewis weights $w_i(A^j)$ for each $j = 1, \ldots, k$ and $i = 1, \ldots, n$
$p_i \leftarrow \max_{1\le j\le k} w_i(A^j)$ for $i = 1, \ldots, n$
$p_i \leftarrow p_i/\|p\|_1$ for $i = 1, \ldots, n$
$S \leftarrow$ zero matrix of dimensions $m \times n$
for $j = 1, \ldots, m$ **do**
    Choose an $i_j \in [n]$ according to the probability distribution $(p_1, \ldots, p_n)$
    $S_{j,i_j} \leftarrow (m \cdot p_{i_j})^{-1/p}$
    Query b_{i_j}
end for
for $j = 1, \ldots, k$ **do**
    $\tilde{x}^j \leftarrow \arg\min_{x\in E} \|f(SA^jx) - Sb\|_p^p$, where $E = \{x : \|SA^jx\|_p^p \le \frac{1}{\epsilon L^p}\|Sb\|_p^p\}$
end for
return $\tilde{x}^1, \ldots, \tilde{x}^k$

4.2 Our Results

For $k = 1$, the latest result is to use $\tilde{O}(d/\epsilon^4)$ queries [32], with an analysis specific to $p = 2$. We generalize the approach to ℓ_p-Lewis weight sampling for $p \ge 1$ and extends it to $k \ge 1$. The following is our main theorem, corresponding to $k = 1$.

Theorem 4.2.1. *Let $p \ge 1$, $f(x)$ be an L-Lipschitz function with $f(0) = 0$ and $A \in \mathbb{R}^{n\times d}$. Suppose that $b \in \mathbb{R}^n$ is accessible via coordinate queries. Suppose that $t_1, \ldots, t_n \in \mathbb{R}$ satisfy that $t_i \ge \beta w_i(A)$, where*

$$\beta \gtrsim_p \begin{cases} \epsilon^{-4}\log(\sum_{i=1}^n t_i), & p = 1 \\ \epsilon^{-4} d^{\max\{\frac{p}{2}-1,0\}} \log^2 d \log(\sum_{i=1}^n t_i), & p > 1. \end{cases} \tag{4.2}$$

Let $m = \sum_i t_i$, $p_i = t_i/m$, $S \in \mathbb{R}^{m\times n}$ be a rescaled sampling matrix of the second kind with sampling probabilities $p_1, \ldots, p_n$ and $\tilde{x} = \arg\min_{x\in E} \|Sf(Ax) - Sb\|_p$, where $E = \{x : \|SAx\|_p^p \le \|Sb\|_p^p/(\epsilon L^p)\}$. It holds with probability at least 0.9 that

$$\|f(A\tilde{x}) - b\|_p^p \le C\left(\|f(Ax^*) - b\|_p^p + \epsilon L^p \|Ax^*\|_p^p\right),$$

where $x^* = \arg\min_x \|f(Ax) - b\|_p$ and $C > 0$ is a constant depending only on p.

Note that for a single matrix $A \in \mathbb{R}^{n\times d}$, we have $\sum_i w_i(A) = d$ so Theorem 4.2.1 implies a sample complexity of $\tilde{O}(d^{\max\{p/2,1\}}/\epsilon^4)$, recovering the result in [32] for $p = 2$.

The following result for $k > 1$ is an immediate corollary of Theorem 4.2.1 with our algorithm given in Algorithm 4.1.

Corollary 4.2.2. *Let $A_1, \ldots, A_k \in \mathbb{R}^{n\times d}$ and $T = \sum_{i=1}^n \max_{j\in[k]} w_i(A^j)$. Let $f(x)$ be an L-Lipschitz function with $f(0) = 0$ and $y \in \mathbb{R}^n$ be the target vector. Algorithm 4.1, when called with*

$$m \sim_p \begin{cases} \epsilon^{-4} T \log(T/\epsilon), & p = 1 \\ \epsilon^{-4} T d^{\max\{\frac{p}{2}-1,0\}} \log^2 d \log(dT/\epsilon), & p > 1, \end{cases} \tag{4.3}$$

outputs solutions $\tilde{x}^1, \ldots, \tilde{x}^k \in \mathbb{R}^d$ such that (4.1) holds for all $j \in [k]$ with probability at least 0.9.

Proof. Let $t_i = \beta \cdot \max_j w_i(A^j)$, then for any fixed j, it holds that $t_i \geq \beta w_i(A^j)$. Also, $m = \sum_i t_i = \beta T$. The sampling probability $p_i = t_i/m = \max_j w_i(A^j)/T$, which is exactly our sampling scheme in Algorithm 4.1. Take

$$\beta \sim \begin{cases} \epsilon^{-4} \log d, & p = 1 \\ \epsilon^{-4} d^{\max\{\frac{p}{2}-1,0\}} \log^2 d \log(dT/\epsilon), & p > 0 \text{ and } p \neq 1 \end{cases},$$

then β satisfies the condition (4.2) in Theorem 4.2.1, whence the conclusion follows. □

Follow-up Work. After our work was accepted to ICLR 2024, Gajjar et al. [33] improved the query complexity for a single matrix A (i.e. $k = 1$) to $\tilde{O}(d/\epsilon^2)$ for $p = 2$. They also studied the agnostic case where the Lipschitz function f is also unknown; that is, the queries to b are for solving the minimization problem $\min_{f,x} \|f(Ax) - b\|_2$. They used a different, although more complicated, argument to achieve the linear dependence on d in query complexity.

4.3 Proof of Theorem 4.2.1

We first need a simple inequality.

Fact 4.3.1. *Suppose that $a, b > 0$ and $p > 0$. It holds that $(a + b)^p \leq 2^{|p-1|}(a^p + b^p)$.*

Let $\mathrm{OPT} = \min_x \|A\theta - y\|_p$. Theorem 4.2.1 is proved by the following chain of inequalities.

$$\begin{aligned} \|f(A\tilde{x}) - y\|_p^p &\overset{(A)}{\leq} 2^{|p-1|}(\|f(A\tilde{x}) - f(Ax^*)\|_p^p + \mathrm{OPT}^p) \\ &\overset{(B)}{\leq} 2^{|p-1|}(\|Sf(A\tilde{x}) - Sf(Ax^*)\|_p^p + \epsilon^2 L^p R^p + \mathrm{OPT}^p) \\ &\overset{(C)}{\leq} 2^{|p-1|}(2^{|p-1|} \|Sf(A\tilde{x}) - Sy\|_p^p + C_1 \mathrm{OPT}^p + \epsilon^2 L^p R^p) \\ &\overset{(D)}{\leq} 2^{|p-1|} \left[C_2 \left(\mathrm{OPT}^p + \epsilon L^p \|Ax^*\|_p^p \right) + C_1 \mathrm{OPT}^p + \epsilon^2 L^p R^p \right] \\ &\overset{(E)}{\leq} C(\mathrm{OPT}^p + \epsilon L^p \|Ax^*\|_p^p) \end{aligned}$$

where inequalities (A) and (C) use Fact 4.3.1, inequality (D) uses [31, Claim 1]. Inequality (E) follows from that

$$\begin{aligned} R^p := \max(\|A\tilde{x}^p\|, \|Ax^*\|^p) &\leq \|A\tilde{x}\|^p + \|Ax^*\|^p \\ &\overset{(EA)}{\leq} 2\|SA\tilde{x}\|_p^p + \|Ax^*\|_p^p \\ &\overset{(EB)}{\leq} 2\frac{\|Sy\|_p^p}{\epsilon L^p} + \|Ax^*\|_p^p \\ &\overset{(EC)}{\leq} 100\frac{\|y\|_p^p}{\epsilon L^p} + \|Ax^*\|_p^p \\ &\overset{(ED)}{\leq} 100 \cdot 2^{|p-1|} \frac{\|f(Ax^*) - y\|_p^p + L^p \|Ax^*\|_p^p}{\epsilon L^p} + \|Ax^*\|_p^p \\ &= 100 \cdot 2^{|p-1|} \frac{\|f(Ax^*) - y\|_p^p}{\epsilon L^p} + \left(\frac{100 \cdot 2^{|p-1|}}{\epsilon} + 1 \right) \|Ax^*\|_p^p, \end{aligned}$$

where inequality (EA) holds because S is a subspace embedding matrix for A, inequality (EB) is from the constraint of our approximate solution in Line 13, inequality (EC) holds with probability at least 49/50 by Markov's inequality and inequality (ED) follows from Fact 4.3.1.

We shall prove inequality (B) in the following lemma. We note that the following lemma is proved in [31, Lemmata 2 and 3], but their sampling complexity is $\tilde{O}(d^2/\epsilon^4)$ with an additional d factor compared with ours. We improve their result by using the reduction technique and removing the ϵ-net argument.

Lemma 4.3.2. *Suppose that $A \in \mathbb{R}^{n\times d}$ and $t_1, \ldots, t_n \in \mathbb{R}$ such that $t_i \geq \beta w_i(A)$ for all i and $p \geq 1$. Let $m = \sum_i t_i$ and $S \in \mathbb{R}^{m\times n}$ be a rescaled sampling matrix of with row sampling probabilities $p_1, \ldots, p_n$, where $p_i = \frac{t_i}{m}$. If*

$$\beta \gtrsim \frac{d^{\max\{\frac{p}{2}-1,0\}}}{\epsilon^2}\left(\log^2 d \log m + \log\frac{1}{\delta}\right)$$

then with probability at least $1-\delta$ and fixed constant $R > 0$, it holds for all pairs of vectors $x_1, x_2 \in \mathbb{R}^d$ with $\|Ax_1\|_p \leq R$ and $\|Ax_2\|_p \leq R$ that

$$\|Sf(Ax_1) - Sf(Ax_2)\|_p^p = \|f(Ax_1) - f(Ax_2)\|_p^p \pm \epsilon L^p R^p.$$

Proof. Let $\alpha = f(Ax_1) - f(Ax_2)$ and $\gamma = Ax_1 - Ax_2$. Denote T to be the set $\mathcal{B}(R)\times\mathcal{B}(R) = \{(x_1, x_2) : \|SAx_1\|_p \leq R, \|SAx_2\|_p \leq R\}$. We shall try to upper bound

$$\mathop{\mathbb{E}}_{S}\left(\max_{(x_1,x_2)\in T}\left|\|S\alpha\|_p^p - \|\alpha\|_p^p\right|\right)^\ell$$

for $\ell = \log\frac{1}{\delta}$.

Since taking the ℓ-th moment of the maximum is a convex function and $\mathbb{E}\,\|S\alpha\|_p^p = \|\alpha\|_p^p$, the symmetrization trick yields that

$$\mathop{\mathbb{E}}_{S}\left(\max_{(x_1,x_2)\in T}\left|\|S\alpha\|_p^p - \|\alpha\|_p^p\right|\right)^\ell \leq 2^\ell \mathop{\mathbb{E}}_{S,\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k=1}^m \sigma_k \frac{|\alpha_{i_k}|^p}{mp_{i_k}}\right|\right)^\ell,$$

where σ_k's are Rademacher variables.

It follows from Theorem 1.2.12 that S is a $\frac{1}{2}$-subspace embedding matrix of A with probability at least $1-\delta/2$. Furthermore, by Lemma 4.3.5, with probability at least $1-\delta/2$, the Lewis weights of SA is upper bounded by $\frac{1}{\beta}$. Let $\mathcal{E}$ denote the event on S that the above two conditions hold. Then $\Pr(\mathcal{E}) \geq 1-\delta$. We assume the following proof is conditioned on $\mathcal{E}$.

Next, we prove the conditional expectation over S and σ when conditioned on $\mathcal{E}$ satisfies that

$$\mathop{\mathbb{E}}_{S,\sigma}\left[\left(\max_{(x_1,x_2)\in T}\left|\sum_{k=1}^m \sigma_k \frac{|\alpha_{i_k}|^p}{mp_{i_k}}\right|\right)^\ell \middle| \mathcal{E}\right] \leq \left(\frac{\epsilon}{2}L^p R^p\right)^\ell \delta. \tag{4.4}$$

Once (4.4) is established, it would follow Markov's inequality that

$$\begin{aligned}
&\Pr\left\{\max_{(x_1,x_2)\in T}\left|\|S\alpha\|_p^p-\|\alpha\|_p^p\right|\geq\epsilon L^pR^p\,\middle|\,\mathcal{E}\right\}\\
&\leq\frac{\mathbb{E}_{S,\sigma}[\left(\max_{(x_1,x_2)\in T}\left|\|S\alpha\|_p^p-\|\alpha\|_p^p\right|\right)^\ell|\mathcal{E}]}{(\epsilon L^pR^p)^\ell}\\
&\leq 2^\ell\frac{\mathbb{E}_{S,\sigma}\left[\left(\max_{(x_1,x_2)\in T}\left|\sum_{k=1}^m\sigma_k\frac{|x_{i_k}|^p}{mp_{i_k}}\right|\right)^\ell\middle|\,\mathcal{E}\right]}{(\epsilon L^pR^p)^\ell}\\
&\leq 2^\ell\frac{(\frac{\epsilon}{2}L^pR^p)^\ell\delta}{(\epsilon L^pR^p)^\ell}\qquad\text{(by (4.4))}\\
&=\delta.
\end{aligned}$$

and then a union bound that

$$\Pr\left\{\max_{(x_1,x_2)\in T}\left|\|S\alpha\|_p^p-\|\alpha\|_p^p\right|\geq\epsilon L^pR^p\,\middle|\,\mathcal{E}\right\}<2\delta,$$

which would complete the proof after rescaling δ to $\delta/2$.

Now we focus on the proof of (4.4), which mostly follows the same approach of Theorem 15.13 in [47].

Let

$$u_k=\frac{f(a_{i_k}^\top x_1)-f(a_{i_k}^\top x_2)}{(mp_{i_k})^{1/p}},\quad v_k=\frac{a_{i_k}^\top x_1-a_{i_k}^\top x_2}{(mp_{i_k})^{1/p}},\quad k\in[m].$$

Then $u=Sx$ and $v=Sy$. We also denote

$$\Lambda=\max_{(x_1,x_2)\in T}\left|\sum_{k=1}^m\sigma_k|u_k|^p\right|,$$

so (4.4) can be rewritten as

$$\mathop{\mathbb{E}}_{S,\sigma}\left[\Lambda^\ell\middle|\,\mathcal{E}\right]\leq\left(\frac{\epsilon}{2}L^pR^p\right)^\ell\delta.$$

We shall split the sum in Λ into two parts: large Lewis weights and small Lewis weights. Specifically, we define $\lambda_k=w_k(SA)/d$ to be the rescaled Lewis weight of SA and $J=\{k\in[m]:\lambda_k\geq\frac{1}{m^2}\}$.

First consider those coordinates not in J (small Lewis weights).

$$\max_{(x_1,x_2)\in T}\left|\sum_{\substack{1\leq k\leq m\\k\notin J}}\sigma_k|u_k|^p\right|\leq\sum_{k\notin J}|u_k|^p\leq L^p\sum_{k\notin J}\lambda_k|\lambda_k^{-\frac{1}{p}}v_k|^p\leq\frac{2^p}{m}d^{\max(1,\frac{p}{2})}L^pR^p,$$

where the last inequality follows from the fact (see [47, Lemma 15.17]) that

$$\max_{k\in[m]}|\lambda_k^{-\frac{1}{p}}v_k|\leq d^{\max(\frac{1}{p},\frac{1}{2})}\|v\|_p\tag{4.5}$$

and (by the definition of $\mathcal{B}(R)$) that $\|v\|_p\leq 2R$.

Next we consider the coordinates in J (large Lewis weights). By the contraction principle, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\sigma_k|u_k|^p\right|\middle|\mathcal{E}\right)^{\ell} &= \mathop{\mathbb{E}}_{\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\lambda_k\sigma_k|\lambda_k^{-\frac{1}{p}}u_k|^p\right|\middle|\mathcal{E}\right)^{\ell}\\ &\le \max_k(\sqrt{\lambda_k})^{\ell}\mathop{\mathbb{E}}_{\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\sqrt{\lambda_k}\sigma_k|\lambda_k^{-\frac{1}{p}}u_k|^p\right|\middle|\mathcal{E}\right)^{\ell}\\ &= (\frac{1}{d\beta})^{\frac{\ell}{2}}\mathop{\mathbb{E}}_{\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\sqrt{\lambda_k}\sigma_k|\lambda_k^{-\frac{1}{p}}u_k|^p\right|\middle|\mathcal{E}\right)^{\ell},\end{aligned}$$

By the triangle inequality, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{\sigma}[\Lambda^{\ell}|\mathcal{E}] &\le \left(\frac{2^p}{m}d^{\max(1,\frac{p}{2})}L^pR^p\right)^{\ell} + (\frac{1}{d\beta})^{\frac{\ell}{2}}\mathop{\mathbb{E}}_{\sigma}\left(\max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\sqrt{\lambda_k}\sigma_k|\lambda_k^{-\frac{1}{p}}u_k|^p\right|\middle|\mathcal{E}\right)^{\ell}\\ &=: \left(\frac{2^p}{m}d^{\max(1,\frac{p}{2})}L^pR^p\right)^{\ell} + (\frac{1}{d\beta})^{\frac{\ell}{2}}\mathop{\mathbb{E}}_{\sigma}[\Xi^{\ell}|\mathcal{E}],\end{aligned}$$

where

$$\Xi = \max_{(x_1,x_2)\in T}\left|\sum_{k\in J}\sqrt{\lambda_k}\sigma_k|\lambda_k^{-\frac{1}{p}}u_k|^p\right|.$$

To bound $\mathbb{E}_{\sigma}[\Xi^{\ell}|\mathcal{E}]$, we introduce the associated distance $\delta((x_1,x_2),(x_1',x_2'))$ so that it is enough to bound it by the estimated entropy of $\mathcal{B}(R)$. We define the distance to be

$$\begin{aligned}&\delta^2((x_1,x_2),(x_1',x_2'))\\ &= \sum_{k\in J}\lambda_k\left(\frac{\left|\lambda_k^{-\frac{1}{p}}[f(a_{i_k}^{\top}x_1)-f(a_{i_k}^{\top}x_2)]\right|^p}{mp_{i_k}} - \frac{\left|\lambda_k^{-\frac{1}{p}}[f(a_{i_k}^{\top}x_1')-f(a_{i_k}^{\top}x_2')]\right|^p}{mp_{i_k}}\right)^2 \qquad (4.6)\\ &:= \sum_{k\in J}\lambda_k(|\lambda_k^{-\frac{1}{p}}u_k|^p - |\lambda_k^{-\frac{1}{p}}u_k'|^p)^2\end{aligned}$$

and the norm

$$\|\theta\|_J := \max_{k\in J}\frac{|\lambda_k^{-\frac{1}{p}}a_{i_k}^{\top}\theta|}{(mp_{i_k})^{\frac{1}{p}}}. \qquad (4.7)$$

By the tail bound of Dudley's integral (see e.g. [77, Theorem 8.1.6]), it holds that

$$\Pr\left\{\Xi \gtrsim \int_0^{\infty}(\log N(T,\delta,\epsilon))^{\frac{1}{2}}d\epsilon + z\cdot\mathrm{diam}(T)\middle|\mathcal{E}\right\} \le \exp(-z^2).$$

According to Lemma 4.3.3, it holds that

$$\int_0^{\infty}(\log N(T,\delta,\epsilon))^{\frac{1}{2}}d\epsilon \lesssim d^{\max(\frac{p-2}{4},0)}L^pR^{p-1}\int_0^{\infty}(\log N(\mathcal{B}(R),B_J,\epsilon))^{\frac{1}{2}}d\epsilon.$$

For $p \geq 2$, the entropy estimate in [47, Proposition 15.18] gives that

$$\begin{aligned}
&d^{\frac{p-2}{4}} L^p R^{p-1} \int_0^\infty (\log N(\mathcal{B}(R), B_J, \epsilon))^{\frac{1}{2}} d\epsilon \\
&= d^{\frac{p-2}{4}} L^p R^{p-1} \int_0^\infty (\log N(\mathcal{B}(1), B_J, \frac{\epsilon}{R}))^{\frac{1}{2}} d\epsilon \\
&\lesssim d^{\frac{p-2}{4}} L^p R^{p-1} \left(\int_0^1 \left(d \log \left(1 + \frac{R\sqrt{d}}{\epsilon} \right) \right)^{\frac{1}{2}} d\epsilon + \int_1^{2\sqrt{d}} \left(\frac{R^2}{\epsilon^2} d \log m \right)^{\frac{1}{2}} d\epsilon \right) \\
&\lesssim d^{\frac{p}{4}} L^p R^p \log d\sqrt{\log m}.
\end{aligned}$$

For $1 < p \leq 2$, it follows from the entropy estimate in [47, Proposition 15.19] and a similar argument to that for $p \geq 2$ that

$$\int_0^\infty (\log N(T, \delta, \epsilon))^{\frac{1}{2}} d\epsilon \lesssim d^{\frac{1}{2}} L^p R^p \log d\sqrt{\log m}.$$

By the property of subgaussian variables (see e.g. Proposition 1.2.5), we have

$$\mathop{\mathbb{E}}_{\sigma}[\Xi^\ell | \mathcal{E}] \leq K^\ell (\sqrt{\ell} d^{\max\{\frac{p}{4}, \frac{1}{2}\}} L^p R^p + d^{\max(\frac{p}{4}, \frac{1}{2})} L^p R^p \log d\sqrt{\log m})^\ell.$$

Hence, given $\ell = \log(1/\delta)$, as long as $\beta \geq 2^{p+1} e \cdot \epsilon^{-2} K^2 d^{\max(\frac{p}{2}-1, 0)} (\log(1/\delta) + \log^2 d \log m)$, it follows that

$$\begin{aligned}
\mathop{\mathbb{E}}_{\sigma}[\Lambda^\ell | \mathcal{E}] &\leq \left(\frac{2^p}{m} d^{\max(1, \frac{p}{2})} L^p R^p \right)^\ell + (\frac{1}{d\beta})^{\frac{\ell}{2}} \mathop{\mathbb{E}}_{\sigma}[\Xi^\ell | \mathcal{E}] \\
&\leq \left(\frac{2^p}{d\beta} d^{\max(1, \frac{p}{2})} L^p R^p \right)^\ell + \left(\frac{K d^{\max(\frac{p}{4}, \frac{1}{2})} L^p R^p (\sqrt{\ell} + \log d\sqrt{\log m})}{\sqrt{d\beta}} \right)^\ell \\
&\leq \left(\frac{\epsilon^2 L^p R^p}{\log(1/\delta) + \log^2 d \log m} \right)^\ell + (\epsilon L^p R^p)^\ell \delta \\
&\leq (\epsilon L^p R^p)^\ell \delta.
\end{aligned}$$

Therefore, taking expectation over S while conditioned on $\mathcal{E}$, we have that $\mathbb{E}_{S,\sigma}[\Lambda^\ell | \mathcal{E}] \leq (\epsilon L^p R^p)^\ell \delta$. Rescaling $\epsilon = \epsilon/2$ completes the proof of (4.4), as desired. □

Lemma 4.3.3. *Let $\delta((x_1, x_2), (x_1', x_2'))$ and $\|x\|_J$ be as defined in (4.6) and (4.7), respectively. It holds that*

$$\delta((x_1, x_2), (x_1', x_2')) \lesssim \begin{cases} d^{\frac{p-2}{4}} L^p R^{p-1} (\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J) & p \geq 2, \\ L^p R^{\frac{p}{2}} (\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)^{\frac{p}{2}} & 1 \leq p \leq 2. \end{cases}$$

Hence $\operatorname{diam}(T)$*, the diameter of the subspace T, is at most $O(d^{\max(\frac{p}{4}, \frac{1}{2})} L^p R^p)$.*

Proof. For $p \geq 2$, we have

$$\begin{aligned}
&\delta^2((x_1, x_2), (x_1', x_2')) \\
&\overset{(A)}{\leq} \sum_{k \in J} \lambda_k |\lambda_k^{-\frac{1}{p}} u_k - \lambda_k^{-\frac{1}{p}} u_k'|^2 (|\lambda_k^{-\frac{1}{p}} u_k|^{p-1} + |\lambda_k^{-\frac{1}{p}} u_k'|^{p-1})^2
\end{aligned}$$

$$
\begin{aligned}
&\overset{\text{(B)}}{\leq} 2L^{2p}p\sum_{k\in J}\lambda_k\left(\frac{\left|\lambda_k^{-\frac{1}{p}}(a_{i_k}^\top x_1 - a_{i_k}^\top x_1')\right| + \left|\lambda_k^{-\frac{1}{p}}(a_{i_k}^\top x_2 - a_{i_k}^\top x_2')\right|}{(mp_{i_k})^{\frac{1}{p}}}\right)^2 \\
&\qquad\qquad \cdot\left(|\lambda_k^{-\frac{1}{p}}v_k|^{2p-2} + |\lambda_k^{-\frac{1}{p}}v_k'|^{2p-2}\right) \\
&\overset{\text{(C)}}{\leq} 2^{p-1}pd^{\frac{p-2}{2}}L^{2p}R^{p-2}\sum_{k\in J}\lambda_k\left(\frac{\left|\lambda_k^{-\frac{1}{p}}(a_{i_k}^\top x_1 - a_{i_k}^\top x_1')\right| + \left|\lambda_k^{-\frac{1}{p}}(a_{i_k}^\top x_2 - a_{i_k}^\top x_2')\right|}{(mp_{i_k})^{\frac{1}{p}}}\right)^2 \\
&\qquad\qquad \cdot\left(|\lambda_k^{-\frac{1}{p}}v_k|^{p} + |\lambda_k^{-\frac{1}{p}}v_k'|^{p}\right) \\
&\overset{\text{(D)}}{\leq} 2^{p-1}pd^{\frac{p-2}{2}}L^{2p}R^{p-2}\left(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J\right)^2\sum_{k\in J}\lambda_k(|\lambda_k^{-\frac{1}{p}}v_k|^p + |\lambda_k^{-\frac{1}{p}}v_k'|^p) \\
&\overset{\text{(E)}}{\leq} 2^{2p-1}pd^{\frac{p-2}{2}}L^{2p}R^{2p-2}\left(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J\right)^2,
\end{aligned}
$$

where the inequality (A) follows from $|a|^p - |b|^p \leq p(|a|^{p-1} + |b|^{p-1})|a-b|$, (B) follows from triangle inequality and $(a+b)^2 \leq 2(a^2+b^2)$, (C) follows from (4.5) and (E) is obtained by $\|v\|_p \leq \|SAx_1\|_p + \|SAx_2\|_p \leq 2R$.

For $1 \leq p \leq 2$, we have

$$
\begin{aligned}
&\delta^2((x_1,x_2),(x_1',x_2')) \\
&\leq \sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k - \lambda_k^{-\frac{1}{p}}u_k'|^2(|\lambda_k^{-\frac{1}{p}}u_k|^{p-1} + |\lambda_k^{-\frac{1}{p}}u_k'|^{p-1})^2 \\
&\leq \max_{k\in J}|\lambda_k^{-\frac{1}{p}}u_k - \lambda_k^{-\frac{1}{p}}u_k'|^p\cdot\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k - \lambda_k^{-\frac{1}{p}}u_k'|^{2-p}(|\lambda_k^{-\frac{1}{p}}u_k|^{2p-2} + |\lambda_k^{-\frac{1}{p}}u_k'|^{2p-2}) \\
&\leq L^p(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)^p(\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k - \lambda_k^{-\frac{1}{p}}u_k'|^p)^{\frac{2-p}{p}} \\
&\qquad\qquad \cdot\left[(\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k|^p)^{\frac{2p-2}{p}} + (\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k'|^p)^{\frac{2p-2}{p}}\right] \\
&\leq L^p(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)^p(\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k|^p + \lambda_k|\lambda_k^{-\frac{1}{p}}u_k'|^p)^{\frac{2-p}{p}} \\
&\qquad\qquad \cdot\left[(\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k|^p)^{\frac{2p-2}{p}} + (\sum_{k\in J}\lambda_k|\lambda_k^{-\frac{1}{p}}u_k'|^p)^{\frac{2p-2}{p}}\right] \\
&\leq 2^pL^{2p}R^p(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)^p,
\end{aligned}
$$

where we use Hölder's inequality $\|fg\|_1 \leq \|f\|_\alpha\|g\|_\beta$ with $\alpha = \frac{p}{2-p}$ and $\beta = \frac{p}{2p-2}$ in the third line.

For $p \geq 2$, the diameter of T is upper bounded by

$$
\begin{aligned}
&\max_{(x_1,x_2)\in T,(x_1',x_2')\in T}\delta((x_1,x_2),(x_1',x_2')) \\
&\leq 2^{\frac{2p-1}{2}}pd^{\frac{p-2}{4}}L^pR^{p-1}(\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)
\end{aligned}
$$

$$\leq 2^{\frac{2p-1}{2}} p d^{\frac{p}{4}} L^p R^p,$$

where we use the fact that $\|x_1 - x_1'\|_J \leq d^{\frac{1}{2}} R$ from (4.5). For $1 \leq p \leq 2$, the diameter of T is upper bounded by $L^p R^{\frac{p}{2}} (\|x_1 - x_1'\|_J + \|x_2 - x_2'\|_J)^{\frac{p}{2}} \leq d^{\frac{1}{2}} L^p R^p$ where we obtain $\|x_1 - x_1'\|_J \leq d^{\frac{1}{p}} R$ from (4.5). □

The next lemma proves the Lewis weights of SA are always within by $[\frac{1}{2\beta}, \frac{2}{\beta}]$. This is already established in Lemma 3.3.4 for rescaled sampling matrices of the first kind for $p \leq 2$. Here, for completeness, we shall use the same idea to establish the result for rescaled sampling matrices of the second kind and for general p.

We first need an analogous result of Lemma 3.3.3, taken from [16, Lemmata 5.3 and 5.4]. This is a much harder result than Lemma 3.3.3.

Lemma 4.3.4. *Suppose $A \in \mathbb{R}^{n \times d}$ and $\overline{w_1}, \ldots, \overline{w_n}$ are the Lewis weights of A. Let $w_1, \ldots, w_n$ be weights such that*

$$\frac{1}{\alpha} w_i^{2/p} \leq a_i^\top \left(\sum_i w_i^{1-2/p} a_i a_i^\top \right)^{-1} a_i \leq \alpha w_i^{2/p}, \quad \forall i = 1, \ldots, n.$$

Then it holds for all i that

$$\alpha^{-c(p,d)} w_i \leq \overline{w_i} \leq \alpha^{c(p,d)} w_i,$$

where $c(p,d) = (p/2)/(1 - |p/2 - 1|)$ when $p < 4$ or $c(p,d) \sim_p \sqrt{d}$ when $p \geq 4$.

Lemma 4.3.5. *Suppose that $A \in \mathbb{R}^{n \times d}$ and $t_1, \ldots, t_n \in \mathbb{R}$ such that $t_i \geq \beta w_i(A)$ for all i. Let $m = \sum_i t_i$ and $S \in \mathbb{R}^{m \times n}$ be a rescaled sampling matrix of the second kind with sampling probabilities $p_1, \ldots, p_n$, where $p_i = \frac{t_i}{m}$. If $\beta \gtrsim \log \frac{d}{\delta}$ when $p < 4$ or $\beta \gtrsim d \log \frac{d}{\delta}$ when $p \geq 4$, then the ℓ_p Lewis weights of SA are upper bounded by $\frac{2}{\beta}$ with probability at least $1 - \delta$.*

Proof. Let $a_i \in \mathbb{R}^d$ be the i-th row of A as a column vector. Without loss of generality, suppose that $A^\top W^{1-\frac{p}{2}} A = I_d$. Hence, the Lewis weights of A are $w_i^{\frac{2}{p}} = a_i^\top (A^\top W^{1-\frac{p}{2}} A)^{-1} a_i = a_i^\top a_i = \|a_i\|_2^2$. We claim that

$$(1 - \epsilon) I_d \preceq \sum_{k=1}^m \frac{a_{i_k} a_{i_k}^\top}{m p_{i_k}} w_{i_k}^{1-\frac{2}{p}} \preceq (1 + \epsilon) I_d$$

holds with probability at least $1 - \delta$. Let $X_k = \frac{a_{i_k} a_{i_k}^\top}{p_{i_k}} w_{i_k}^{1-\frac{2}{p}}$ and then we have $\mathbb{E}\, X_k = I_d$. First, we have $\mathbb{E}\, X_k = I_d$ and $\|X_k - I_d\|_2 \leq 1 + \frac{\|a_{i_k}\|_2^2}{w_{i_k}/d} w_{i_k}^{1-\frac{2}{p}} = 1 + \frac{m}{\beta}$. Besides, we have that

$$\begin{aligned}
\|\mathbb{E}\,(X_k - I_d)\|_2^2 &= \left\| \mathbb{E} (X_k - I_d)^\top (X_k - I_d) \right\|_2 \\
&= \left\| \mathbb{E}\, X_k^\top X_k - I_d \right\|_2 \\
&= \left\| \frac{w_{i_k}}{p_{i_k}} \cdot \mathbb{E}\, \frac{a_{i_k} a_{i_k}^\top w_{i_k}^{1-\frac{2}{p}}}{p_{i_k}} - I_d \right\|_2 \\
&= \left\| \frac{w_{i_k}}{p_{i_k}} \sum_{i=1}^n a_i a_i^\top w_{i_k}^{1-2/p} + I_d \right\|_2 \\
&\leq 1 + \frac{m}{\beta}.
\end{aligned}$$

By matrix Chernoff bound, it follows that

$$\Pr\left\{\left\|\frac{1}{m}\sum_{k=1}^{m}(X_k - I_d)\right\|_2 \geq \epsilon\right\} \leq 2d\exp\left(\frac{-m\epsilon^2}{1+d+(1+d)\cdot\epsilon/3}\right) \leq 2d\exp\left(-\beta\epsilon^2\right)$$

Setting $\beta \gtrsim \frac{1}{\epsilon^2}\log\frac{d}{\delta}$ guarantees the failure probability to be at most δ, proving the claim. Therefore, we have that

$$(1-\epsilon)\left(\frac{d}{m}\right)^{1-\frac{2}{p}} I_d \preceq \left[\sum_{k=1}^{m}\frac{a_{i_k}}{(mp_{i_k})^{\frac{1}{p}}}\left(\frac{w_{i_k}}{dp_{i_k}}\right)^{1-\frac{2}{p}}\frac{a_{i_k}^\top}{(mp_{i_k})^{\frac{1}{p}}}\right]^{-1} \preceq (1+2\epsilon)\left(\frac{d}{m}\right)^{1-\frac{2}{p}} I_d$$

holds with probability at least $1-\delta$. Hence, it follows that

$$\frac{a_i^\top}{(mp_i)^{1/p}}\cdot\left[\sum_{k=1}^{m}\frac{a_{i_k}}{(mp_{i_k})^{\frac{1}{p}}}\left(\frac{dp_{i_k}}{w_{i_k}}\right)^{\frac{2}{p}-1}\frac{a_{i_k}^\top}{(mp_{i_k})^{\frac{1}{p}}}\right]^{-1}\cdot\frac{a_i}{(mp_i)^{1/p}} \leq (1+2\epsilon)\frac{d}{m}\left(\frac{w_{i_k}}{dp_{i_k}}\right)^{2/p}.$$

Applying Lemma 4.3.4 and setting ϵ to be a constant (depending on p) for $p<4$ and $\epsilon \sim_p 1/\sqrt{d}$ for $p \geq 4$ gives that $w_i(SA) \leq 2\frac{d}{m}\frac{w_i}{dp_i} \leq \frac{2}{\beta}$. □

4.4 Result for Sampling of First Kind

We use sampling of the second kind in our algorithm. However, our main result Theorem 4.2.1 still works for the first kind of rescaled sampling matrices. Specifically, let $p_i = \min\{\beta w_i, 1\}$, where $\beta = \Omega(\frac{d^{\frac{p}{2}-1}}{\epsilon^2}(\log^2 d\log\frac{d}{\epsilon}+\log\frac{1}{\delta}))$, and S be the rescaled sampling matrix of the first kind with sampling probabilities $p_1,\ldots,p_n$. Accordingly, Lines 4–9 of Algorithm 4.1 are changed to the following lines. We remove zero rows in the sample matrix for efficiency.

Algorithm 4.2 Replacement of Lines 4–9 in Algorithm 4.1

```
S ← zero matrix of dimensions n × n
m ← 0
for i = 1, ..., n do
    Generate a random variable X ~ Ber(p_i)
    if X = 1 then
        m ← m + 1
        S_{m,i} = p_i^{-1/p}
        Query b_i
    end if
end for
Truncate S to its first m rows
```

Compared to the proof of Theorem 4.2.1, the following modifications are needed: (1) By Theorem 5.2.2, S is a 1/2-subspace-embedding matrix of A. (2) To show that Lemma 4.3.2 holds, we observe that by Lemma 3.3.4, Lewis weights of SA are uniformly upper bounded by $\frac{2}{\beta}$.

Chapter 5

ℓ_p-Subspace Embedding for $p > 2$

In this chapter, we revisit the ℓ_p-subspace embedding obtained via Lewis weight sampling for $p > 2$. This type of result was first formally presented by Cohen and Peng [16, Theorem 7.1], employing the second kind of sampling with $\tilde{O}(d^{p/2}/\epsilon^5)$ samples. This complexity matches the ℓ_p-subspace embedding dimension due to Bourgain et al. [6], from which Cohen and Peng derive their result. However, Cohen and Peng did not provide a clear proof; they use Lewis weights as lower bounds for sampling probabilities, while [6] uses exact Lewis weights for sampling. Later, Woodruff and Yasuda [82] improved the sample complexity to $\tilde{O}(d^{p/2}/\epsilon^2 \log n)$ under the first kind of sampling scheme, achieving the best possible ϵ^{-2} dependence but introducing an undesirable $\log n$ factor.

In this chapter, we shall first, for completeness, provide a proof that Lewis weight sampling indeed gives an ℓ_p subspace embedding, with a target dimension of $\tilde{O}(d^{p/2}/\epsilon^5)$. We shall then improve this target dimension to $\tilde{O}(d^{p/2}/\epsilon^2)$, removing the undesirable $\log n$ factor in the sample complexity of [82].

Follow-up Work. After our result of $\tilde{O}(d^{p/2}/\epsilon^2)$ dimensions was published in [37], Yasuda [85] independently improved the sample complexity in his PhD thesis to $\tilde{O}(d^{p/2}/\epsilon^2)$ without the $\log n$ factor by following the idea in [16] and using a flattening argument that studies the Lewis weights of the concatenated matrix $\binom{SA}{A}$ for a constant-factor subspace embedding matrix S. Our argument is similar at large, though arguably simpler.

5.1 Sample Complexity of $\tilde{O}(d^{p/2}/\epsilon^5)$

The proof follows from [6, Theorem 7.3], adapted to the usual linear algebraic language. We first restate the theorem from Theorem 1.2.12.

Theorem 5.1.1. *Let $A \in \mathbb{R}^{n\times d}$ and $\epsilon \in (0, 1/2]$. Let $t_1, t_2, \ldots, t_n$ be a sequence of real numbers such that $t_i \geq \beta w_i(A)$, $w(A)$ is the ℓ_p Lewis weight of A as in Definition 1.2.8, $N = \sum_i t_i$ and $p_i = t_i/N$, where*

$$\beta \gtrsim_p \frac{d^{p/2-1}}{\epsilon^5}\left(\log\frac{d}{\epsilon} + \log\frac{1}{\delta}\right).$$

Suppose that $S \in \mathbb{R}^{N\times n}$ is a rescaled sampling matrix of the second kind with respect to $p_1, \ldots, p_n$. Then with probability at least $1-\delta$, S is a $(1+\epsilon)$-subspace-embedding for A in the ℓ_p-norm.

Proof. Consider the sphere $\mathcal{S} = \{Ax : \|Ax\|_p = 1\}$. Our goal is to show that

$$\sup_{u\in\mathcal{S}} \left| \|Su\|_p^p - 1 \right| \le \epsilon. \tag{5.1}$$

Let $\mathcal{F}$ be a standard ϵ-net on $\mathcal{S}$ with respect to the ℓ_p-norm. By a standard volume argument, we can choose $\mathcal{F}$ such that

$$\log |\mathcal{F}| \le d \log(1 + \frac{2}{\epsilon}).$$

We claim that it suffices to show

$$\sup_{u\in\mathcal{F}} \left| \|Su\|_p^p - 1 \right| \le \epsilon. \tag{5.2}$$

Indeed, assume that (5.2) holds and let $\Delta = \sup_{u\in\mathcal{S}} \left| \|Su\|_p^p - 1 \right|$. Suppose that this supremum is attained at $u^* \in \mathcal{S}$. there exists $v \in \mathcal{F}$ such that $\|u^* - v\|_p \le \epsilon$. Then

$$\begin{aligned}
\|Su^*\|_p - 1 &\le \|Sv\|_p + \|S(u^* - v)\|_p - 1 \\
&= (\|Sv\|_p - 1) + \|S(u^* - v)\|_p \\
&\le \epsilon + (1 + \Delta)\|u^* - v\|_p \\
&\le \epsilon + (1 + \Delta)\epsilon.
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
\|Su^*\|_p - 1 &\ge \|Sv\|_p - \|S(u^* - v)\|_p - 1 \\
&= (\|Sv\|_p - 1) - \|S(u^* - v)\|_p \\
&\ge -\epsilon - (1 + \Delta)\|u^* - v\|_p \\
&\ge -\epsilon - (1 + \Delta)\epsilon.
\end{aligned}$$

Combining the inequality of both directions yields that

$$\Delta \le \epsilon + (1 + \Delta)\epsilon,$$

whence we can solve that $\Delta \le 2\epsilon/(1 - \epsilon) \le 4\epsilon$. Rescaling ϵ proves (5.1).

Next, suppose that $Ax \in \mathcal{F}$. Let $w_i = w_i(A)$ and $W = \mathrm{diag}\{w_1, \ldots, w_n\}$. We then have

$$\begin{aligned}
\|SAx\|_p^p - \|Ax\|_p^p &= \sum_{j=1}^{N} \frac{|\langle a_{i_j}, x\rangle|^p}{N p_{i_j}} - \left\| W^{\frac{1}{p}} W^{-\frac{1}{p}} Ax \right\|_p^p \\
&= \frac{1}{N} \sum_{j=1}^{N} \left| \langle p_{i_j}^{-\frac{1}{p}} a_{i_j}, x\rangle \right|^p - \sum_{i=1}^{n} w_i \left| \langle w_i^{-\frac{1}{p}} a_i, x\rangle \right|^p \\
&= \frac{1}{N} \sum_{j=1}^{N} \left| \langle p_{i_j}^{-\frac{1}{p}} a_{i_j}, x\rangle \right|^p - \sum_{i=1}^{n} \frac{w_i}{d} \left| \left\langle (\frac{w_i}{d})^{-\frac{1}{p}} a_i, x \right\rangle \right|^p.
\end{aligned}$$

In the remainder of the proof, we rescale w_i to w_i/d so that $\sum_i w_i = 1$. Then $p_i \ge (\beta d/N) w_i$. Let $\lambda = \beta d/N$ and write $p_i = \lambda_i w_i$, we have $\lambda_i \ge \lambda$ and $\lambda < 1$. Our goal has thus become to show that

$$\left| \frac{1}{N} \sum_{j=1}^{N} \left| p_{i_j}^{-\frac{1}{p}} a_{i_j}^\top x \right|^p - \sum_{i=1}^{n} w_i \left| w_i^{-\frac{1}{p}} a_i^\top x \right|^p \right| \le \epsilon \tag{5.3}$$

for all $Ax \in \mathcal{F}$.

Let $B_{w,p}$ be the unit ball in the column space of $W^{-\frac{1}{p}}A$ with respect to the $\|\cdot\|_{w,p}$ norm. Note that $\|Ax\|_p = \|W^{-1/p}Ax\|_{w,p}$ and thus $\mathcal{F}$ is a net on the boundary of $B_{w,p}$. Let $y = W^{-\frac{1}{p}}Ax \in B_{w,p}$, then Equation (5.3) is equivalent to

$$\left| \frac{1}{N} \sum_{j=1}^{N} \left| \lambda_{i_j}^{-\frac{1}{p}} y_{i_j} \right|^p - \sum_{i=1}^{n} w_i \left| y_{i_j} \right|^p \right| \leq \epsilon \tag{5.4}$$

for all $y \in \mathcal{F}$.

It follows directly from Lemma 1.2.9(3) that $\|y\|_\infty \leq \sqrt{d}$ whenever $y \in B_{w,p}$. By [6, Proposition 7.2], there exists a subset $\mathcal{A}_k \subseteq \mathcal{F}$ such that

$$B_{w,p} \subseteq \bigcup_{g \in \mathcal{A}_k} \left(g + \frac{\epsilon(1+\epsilon)^k}{3} \cdot B_\infty \right), \tag{5.5}$$

for any $k = 1, 2, \ldots, l = \lceil \frac{\log d^{\frac{1}{2}}}{\log(1+\epsilon)} \rceil + 1$ with the size $|\mathcal{A}_k|$ satisfying

$$\log |\mathcal{A}_k| \lesssim p \frac{d}{\epsilon^2 (1+\epsilon)^{2k}} \log \frac{d}{\epsilon}.$$

For every $y \in \mathcal{F}$, there exists a corresponding vector $y^{(k)} \in \mathcal{A}_k$ with $\|y - y^{(k)}\|_\infty \leq \epsilon(1+\epsilon)^k/3$. Next, we split the coordinates of $y^{(k)}$ into different levels. Define

$$\begin{aligned} C_{0,y} &= [n], \\ C_{k,y} &= \{i \in [n] : \left|(y^{(k)})_i\right| \geq (1+\epsilon)^{k-1}\}, \quad k \in [l] \\ D_{k,y} &= C_{k,y} \setminus (\cup_{h>k} C_{h,y}), \quad k = 0, 1, \ldots, l \end{aligned}$$

and a new vector $\tilde{y}$ such that its i-th coordinate is defined to be

$$\tilde{y}_i = |y_i| \cdot \chi_{D_{0,y}}(i) + \sum_{k=1}^{l} (1+\epsilon)^k \cdot \chi_{D_{k,y}}(i),$$

where $\chi_{D_{k,y}}$ is an indicator function such that $\chi_{D_{k,y}}(i) = 1$ if $i \in D_{k,y}$ for all $i \in [n]$ and $k \in \{0\} \cup [l]$.

For all $i \in C_{k,y}$, it holds that

$$|y_i| \geq \left|(y^{(k)})_i\right| - \left|y_i - (y^{(k)})_i\right| \geq (1+\epsilon)^{k-1} - \frac{\epsilon(1+\epsilon)^k}{3} > (1+\epsilon)^{k-2},$$

and for all $i \notin C_{k,y}$, it holds that

$$|y_i| \leq \left|(y^{(k)})_i\right| + \left|y_i - (y^{(k)})_i\right| < (1+\epsilon)^{k-1} + \frac{\epsilon(1+\epsilon)^k}{3} \leq (1+\epsilon)^k.$$

In the preceding two chains of inequalities, both first steps follow from the definition (5.5) of $\mathcal{A}_k$. It follows that

$$(1+\epsilon)^{k-2} \leq |y_i| \leq (1+\epsilon)^{k+1}, \quad \forall i \in D_{k,y}$$

and that $|y_i| \leq 1$ for all $i \in D_{0,y}$. It follows that

$$\frac{\tilde{y}_i}{|y_i|} \in [(1+\epsilon)^{-1}, (1+\epsilon)^2] \qquad \text{for all } i \in [n] \text{ and } y \in \mathbb{R}^d.$$

Next, we claim that to show (5.4), it suffices to show the following inequality for all $y \in \mathcal{F}$:

$$\left| \frac{1}{N} \sum_{j=1}^{N} \left| \lambda_{i_j}^{-\frac{1}{p}} \tilde{y}_i \right|^p - \sum_{i=1}^{n} w_i \left| \tilde{y}_i \right|^p \right| \leq \epsilon. \tag{5.6}$$

Indeed, suppose that (5.6) holds and it follows that

$$\begin{aligned}
&\frac{1}{N} \sum_{j=1}^{N} \lambda_{i_j}^{-1} |y_i|^p - \sum_{i=1}^{n} w_i |y_i|^p \\
&\leq (1+\epsilon) \frac{1}{N} \sum_{j=1}^{N} \lambda_{i_j}^{-1} \left| \tilde{y}_{i_j} \right|^p - \frac{1}{(1+\epsilon)^2} \sum_{i=1}^{n} w_i \left| \tilde{y}_i \right|^p \\
&\leq \epsilon(1+\epsilon) + \left(1 + \epsilon - \frac{1}{(1+\epsilon)^2} \right) \sum_{i=1}^{n} w_i \left| \tilde{y}_i \right|^p \qquad \text{(by (5.6))} \\
&\leq \epsilon(1+\epsilon) + (1 + \epsilon - (1-\epsilon)^2)(1+\epsilon)^2 \sum_{i=1}^{n} w_i |y_i|^p \\
&\leq \epsilon(1+\epsilon) + 3\epsilon(1+\epsilon)^2 \|y\|_{w,p} \\
&\leq 9\epsilon,
\end{aligned}$$

and similarly

$$\begin{aligned}
&\frac{1}{N} \sum_{j=1}^{N} \lambda_{i_j}^{-1} \left| y_{i_j} \right|^p - \sum_{i=1}^{n} w_i |y_i|^p \\
&\geq \frac{1}{(1+\epsilon)^2} \cdot \frac{1}{N} \sum_{i_j=1}^{N} \lambda_{i_j}^{-1} \left| \tilde{y}_{i_j} \right|^p - (1+\epsilon) \sum_{i=1}^{n} w_i \left| \tilde{y}_i \right|^p \\
&\geq -\frac{\epsilon}{(1+\epsilon)^2} + \left(\frac{1}{(1+\epsilon)^2} - (1+\epsilon) \right) \sum_{i=1}^{n} w_i \left| \tilde{y}_i \right|^p \\
&\geq -\frac{\epsilon}{(1+\epsilon)^2} + \left(\frac{1}{(1+\epsilon)^2} - (1+\epsilon) \right) \frac{1}{1+\epsilon} \sum_{i=1}^{n} w_i |y_i|^p \\
&\geq -4\epsilon.
\end{aligned}$$

Rescaling ϵ by a constant factor gives (5.3).

Now we turn to prove inequality (5.6). According to the definition of $\tilde{y}$, it suffices to prove that for all $y \in \mathcal{F}$

$$\left| \frac{1}{N} \sum_{j=1}^{N} \lambda_{i_j}^{-1} \left| \tilde{y}_{i_j} \right|^p \cdot \chi_{D_{0,y}}(i_j) - \sum_{i \in D_{0,y}} w_i \left| \tilde{y}_i \right|^p \right| \leq \epsilon_0 \tag{5.7}$$

and

$$\left|\sum_{j=1}^{N} N^{-1}(1+\epsilon)^{pk}\left(\lambda_{i_j}^{-1}\chi_{D_{k,y}}(i_j) - \sum_{i\in D_{k,y}} w_i\right)\right| \le \epsilon_k \tag{5.8}$$

where $\epsilon_0, \ldots, \epsilon_l$ are chosen such that

$$\sum_{k=0}^{l} \epsilon_k \le \epsilon.$$

We prove the inequalities (5.7) and (5.8) by Bernstein's inequality (Lemma 1.2.1). Let $X_j = \lambda_{i_j}^{-1}\left|\tilde{y}_{i_j}\right|^p \cdot \chi_{D_{0,y}}(i_j) - \sum_{i\in D_{0,y}} w_i \left|\tilde{y}_i\right|^p$. Then we have

$$\begin{aligned}
|X_j| &\le \lambda^{-1}\left|\tilde{y}_{i_j}\right|^p \chi_{D_{0,y}}(i_j) + \sum_{i\in D_{0,y}} w_i \left|\tilde{y}_i\right|^p \\
&\le \lambda^{-1}(1+\epsilon)^{2p}\left|y_{i_j}\right|^p \chi_{D_{0,y}}(i_j) + (1+\epsilon)^{2p}\sum_{i\in D_{0,y}} w_i \left|y_i\right|^p \\
&\le \frac{(1+\epsilon)^{2p}}{\lambda} + (1+\epsilon)^{2p} \\
&\le \frac{2}{\lambda}(1+\epsilon)^{2p}
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}\, X_j^2 &\le \mathbb{E}(\lambda_{i_j}^{-1}|\tilde{y}_{i_j}|^p \chi_{D_{0,y}}(i_j))^2 \\
&\le \lambda^{-1}\,\mathbb{E}\,\lambda_{i_j}^{-1}|\tilde{y}_{i_j}|^{2p}\chi_{D_{0,y}}(i_j) \\
&\le \lambda^{-1}\sum_{i=1}^{N} |\tilde{y}_i|^{2p}\chi_{D_{0,y}}(i)\frac{p_i}{\lambda_i} \\
&= \lambda^{-1}\sum_{i\in D_{0,y}} |\tilde{y}_i|^{2p} w_i \\
&\le \lambda^{-1}\sum_{i\in D_{0,y}} (1+\epsilon)^{2p}|y_i|^{2p} w_i \\
&\le \lambda^{-1}(1+\epsilon)^{2p}\sum_{i\in D_{0,y}} |y_i|^p w_i \\
&\le \frac{1}{\lambda}(1+\epsilon)^{2p},
\end{aligned}$$

where we used the fact that $|y_i| \le 1$ for $i \in D_{0,y}$ and $\sum_{i\in D_{0,y}} w_i |y_i|^p \le \|y\|_{w,p}^p = 1$.

Applying Bernstein's inequality (Lemma 1.2.1) gives that the failure probability of (5.7) for an arbitrary fixed y is at most

$$2\exp\left(-\frac{\frac{1}{2}N\lambda\epsilon_0^2}{(1+\epsilon)^{2p} + \frac{2}{3}(1+\epsilon)^{2p}\epsilon_0}\right) \le 2\exp\left(-\frac{\beta d\epsilon_0^2}{3(1+\epsilon)^{2p}}\right).$$

Similarly, the failure probability of (5.8) for an arbitrary fixed y is at most

$$2\exp\left(-\frac{\beta d\epsilon_k^2}{3(1+\epsilon)^{pk}}\right).$$

Let $\mathcal{B}_k = \{D_{k,y} : y \in \mathcal{F}\}$ for $k \in [l]$. It follows from the definition of $D_{k,y}$ that

$$\log |\mathcal{B}_k| \leq \sum_{h=k}^{l} \log |\mathcal{A}_h| \lesssim p \frac{d}{\epsilon^3 (1+\epsilon)^{2k}} \log \frac{d}{\epsilon}.$$

Therefore, for both (5.7) and (5.8) to hold with probability at least $1 - \delta$, it suffices to have that

$$|\mathcal{F}| \cdot 2 \exp\left(-\frac{\beta d \epsilon_0^2}{3(1+\epsilon)^{2p}}\right) + \sum_{k=1}^{l} |\mathcal{B}_k| \cdot 2 \exp\left(-\frac{\beta d \epsilon_k^2}{3(1+\epsilon)^{pk}}\right) < \delta. \tag{5.9}$$

Take $\epsilon_0 = c_1 \epsilon$, $\epsilon_l = c_2 \epsilon$ and $\epsilon_k = (1+\epsilon)^{\frac{(2-p)(l-k)}{2}} \epsilon_l$ for $1 \leq k < l$, where c_1 is an absolute constant and c_2 is a constant that depends only on p. Then when

$$\beta \gtrsim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^5} \left(\log \frac{d}{\epsilon} + \log \frac{1}{\delta}\right),$$

we can verify that

$$\begin{aligned}
|\mathcal{B}_k| \cdot 2 \exp\left(-\frac{\beta d \epsilon_k^2}{3(1+\epsilon)^{pk}}\right) &\leq \exp\left(C_1 \frac{d}{\epsilon^3 (1+\epsilon)^{2k}} \log \frac{d}{\epsilon} - \frac{c_3 \beta d \epsilon^2}{d^{p/2-1} (1+\epsilon)^{2k}}\right) \\
&\leq \exp\left(C_2 \frac{1}{\epsilon^3} \log \frac{d}{\epsilon} - \frac{c_4 \beta \epsilon^2}{d^{p/2-1}}\right) \\
&\leq \exp\left(-\frac{C_3}{\epsilon^3} \log \frac{d}{\epsilon} - \frac{C_3}{\epsilon^3} \log \frac{1}{\delta}\right) \\
&< \frac{\epsilon \delta}{C_4 d}
\end{aligned}$$

(where we use the fact that $(1+\epsilon)^{2k} \leq (1+\epsilon)^{2l} \sim d$ in the second inequality) and

$$\begin{aligned}
|\mathcal{F}| \cdot 2 \exp\left(-\frac{\beta d \epsilon_0^2}{3(1+\epsilon)^{2p}}\right) &\leq \exp\left(C_3 d \log \frac{1}{\epsilon} - c_5 \beta d \epsilon^2\right) \\
&\leq \exp\left(-C_5 \frac{d}{\epsilon^3} \log \frac{d}{\epsilon} - C_5 \frac{d}{\epsilon^3} \log \frac{1}{\delta}\right) \\
&< \frac{\epsilon \delta}{C_4 d}.
\end{aligned}$$

We see that (5.9) indeed holds, noticing that there are only $l+1 \sim (\log d)/\epsilon$ summands. □

5.2 Sample Complexity of $\tilde{O}(d^{p/2}/\epsilon^2)$

We first state and prove the result for the second kind of sampling scheme (i.i.d. row sampling).

Theorem 5.2.1. *Let $A \in \mathbb{R}^{n \times d}$, $2 < p < \infty$ and $0 < \epsilon, \delta < 1$. Suppose that $t_i \sim \beta w_i(A)$ and $w_i(A)$ is the ℓ_p Lewis weight of A as in Definition 1.2.8 for all $i \in [n]$, where*

$$\beta \sim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2} \left((\log d)^2 \log\left(\frac{d}{\epsilon} \log \frac{1}{\delta}\right) + \log \frac{1}{\delta}\right)$$

is the oversampling parameter. Let $m = \sum_i t_i$ and $p_i = t_i/m$. Suppose that $S \in \mathbb{R}^{m \times d}$ is a rescaled sampling matrix of the second kind with sampling probabilities $p_1, \ldots, p_n$, then S is an $(1+\epsilon)$-subspace-embedding for A in the ℓ_p-norm with probability at least $1 - \delta$.

Proof. In order to prove S is an $(1+\epsilon)$-subspace-embedding matrix for A, we prove

$$\max_{\|Ax\|_p \le 1} \left| \|SAx\|_p^p - \|Ax\|_p^p \right| \le \epsilon. \tag{5.10}$$

If the following inequality holds

$$\mathop{\mathbb{E}}_{S} \left(\max_{\|Ax\|_p \le 1} \left| \|SAx\|_p^p - \|Ax\|_p^p \right| \right)^{\ell} \le \epsilon^{\ell} \delta \tag{5.11}$$

for $\ell = \log \frac{1}{\delta}$, then by Markov's inequality, Equation (5.10) holds with probability at least $1-\delta$.

By the symmetrization trick,

$$\mathop{\mathbb{E}}_{S} \left(\max_{\|Ax\|_p \le 1} \left| \|SAx\|_p^p - \|Ax\|_p^p \right| \right)^{\ell} \le 2^{\ell} \mathop{\mathbb{E}}_{S,\sigma} \left(\max_{\|Ax\|_p \le 1} \left| \sum_{k=1}^{m} \sigma_k |(SA)_k x|^p \right| \right)^{\ell},$$

where σ_k's are independent Rademacher variables and $(SA)_k$ denotes the k-th row of SA. Then

(1) By Theorem 1.2.12, S is a $\frac{1}{2}$-subspace embedding matrix of A with probability at least $1-\delta/2$;

(2) By Lemma 4.3.5, Lewis weights of SA are uniformly upper bounded by $\frac{2}{\beta}$ with probability at least $1-\delta/2$.

Let $\mathcal{E}$ denote the event that (1) and (2) both happen, then $\Pr(\mathcal{E}) \ge 1-\delta$. It follows from (1) that

$$\begin{aligned} \left(\max_{\|Ax\|_p \le 1} \left| \sum_{k=1}^{m} \sigma_k |(SA)_k x|^p \right| \right)^{\ell} &\le \left(\max_{\|SAx\|_p \le 2} \left| \sum_{k=1}^{m} \sigma_k |(SA)_k x|^p \right| \right)^{\ell} \\ &= 2^p \cdot \left(\max_{\|SAx\|_p \le 1} \left| \sum_{k=1}^{m} \sigma_k |(SA)_k x|^p \right| \right)^{\ell}. \end{aligned}$$

The remainder of the proof is near-identical to [47, Proposition 15.18]. We define

$$\Psi = \max_{x:\|SAx\|_p \le 1} \left| \sum_{k=1}^{m} \sigma_k |(SA)_k x|^p \right|$$

Using the fact that the Lewis weights of SA are upper bounded by $2/\beta$, we have similarly to [47, (15.17)] that

$$\mathop{\mathbb{E}}_{\sigma}(\Psi^{\ell} \big| \mathcal{E}) \le \left(\frac{3d^{\frac{p}{2}}}{2m} \right)^{\ell} + \left(\frac{2}{\beta d} \right)^{\frac{\ell}{2}} \mathop{\mathbb{E}}_{\sigma} \left[\left(\max_{x:\|SAx\|_p \le 1} \left| \sum_{k \in J} \sigma_k \lambda_k^{\frac{1}{2}} |\lambda_k^{-\frac{1}{p}} (SA)_k x|^p \right| \right)^{\ell} \middle| \mathcal{E} \right],$$

where $\lambda_k = w_k(SA)/d$ is the rescaled Lewis weight of SA and $J = \{k : \lambda_k > m^{-2}\}$ is the same as in [47, Propsition 15.18]. The task is then to upper bound $\mathbb{E}_{\sigma}(\Psi^{\ell}|\mathcal{E})$, where $\Xi = \Xi(\sigma)$ is defined as

$$\Xi = \max_{x:\|SAx\|_p \le 1} \left| \sum_{k \in J} \sigma_k \lambda_k^{\frac{1}{2}} \left| \lambda_k^{-\frac{1}{p}} (SA)_k x \right|^p \right|.$$

We define

$$Z(x) = \sum_{k\in J} \sigma_k \lambda_k^{\frac{1}{2}} \left| \lambda_k^{-\frac{1}{p}} (SA)_k x \right|^p,$$

$$T = \{\Lambda^{-\frac{1}{p}} SAx : \|SAx\|_p \le 1\}, \quad \Lambda = \mathrm{diag}\{\lambda_1, \lambda_2, \dots, \lambda_m\},$$

and thus $\Xi = \max_{x\in T} |Z(x)|$. Now the problem is reduced to study the Rademacher process $Z(x)$ over the metric space T endowed with pseudo-metric

$$d(x,y) = \left(\sum_{i\in J} \lambda_i (|x_i|^p - |y_i|^p)^2 \right)^{\frac{1}{2}}.$$

By [47, (15.18)], the diameter of T is

$$\mathrm{diam}(T) \lesssim_p d^{\frac{p}{4}}.$$

Let us show how to deduce the tail bound for Ξ. First, by [47, (15.18)], it holds that

$$\int_0^\infty \sqrt{\ln N(T,d,\epsilon)} \lesssim_p \left(d^{\frac{p}{2}} (\log d)^2 \log m \right)^{\frac{1}{2}}.$$

Now, recall the tail bound version of Dudley's integral in Lemma 1.2.4 that

$$\Pr\left\{ \sup_{x,y\in T} |Z(x) - Z(y)| \ge C \left(\int_0^\infty \sqrt{\ln N(T,d,\epsilon)} + z \cdot \mathrm{diam}(T) \right) \right\} \le C' \exp(-z^2),$$

and note that $Z(0) = 0$, we have therefore

$$\Pr\left\{ \Xi \gtrsim d^{\frac{p}{4}} \log d \sqrt{\log m} + z \cdot d^{\frac{p}{4}} \,\middle|\, \mathcal{E} \right\} \le C' \exp(-z^2).$$

By the moment bound of subgaussian variables (Proposition 1.2.5), it holds that

$$\mathop{\mathbb{E}}_{\sigma}(\Xi^\ell \,|\, \mathcal{E}) \le K^\ell (\sqrt{\ell} d^{\frac{p}{4}} + d^{\frac{p}{4}} \log d \sqrt{\log m})^\ell,$$

for some constant K. Hence

$$\mathop{\mathbb{E}}_{\sigma}(\Psi^\ell \,|\, \mathcal{E}) \le \left(\frac{3d^{\frac{p}{2}}}{2m} \right)^\ell + \left(\frac{2}{d\beta} \right)^{\frac{\ell}{2}} \mathop{\mathbb{E}}_{\sigma}(\Xi^\ell \,|\, \mathcal{E}) \lesssim_p \left(\frac{\epsilon}{2} \right)^\ell \delta,$$

provided that

$$\beta \gtrsim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2} \left((\log d)^2 \log m + \log \frac{1}{\delta} \right),$$

Since $m \sim \beta d$, this condition is satisfied when

$$\beta \sim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2} \left((\log d)^2 \log \left(\frac{d}{\epsilon} \log \frac{1}{\delta} \right) + \log \frac{1}{\delta} \right).$$

Unconditioning on $\mathcal{E}$ and rescaling δ completes the proof. □

We remark that for a constant δ, Theorem 5.2.1 achieves, up to a constant factor, an embedding dimension that matches the upper bound of $N_p(d,\epsilon)$ in Lemma 1.2.7, which was previously achieved by an iterative argument.

Next, we state and prove the result for the first kind of sampling scheme.

Theorem 5.2.2. *Let $A \in \mathbb{R}^{n\times d}$, $2 < p < \infty$ and $0 < \epsilon, \delta < 1$. Suppose that the oversampling parameter*

$$\beta \sim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2}\left((\log d)^2 \log\left(\frac{d}{\epsilon}\log\frac{1}{\delta}\right) + \log\frac{1}{\delta}\right).$$

Let $p_i = \min\{\beta w_i(A), 1\}$ and $S \in \mathbb{R}^{n\times n}$ be the rescaled sampling matrix of the first kind with probabilities $p_1, \ldots, p_n$. With probability at least $1-\delta$, S is a $(1+\epsilon)$-subspace-embedding for A and has

$$m \lesssim_p \frac{d^{\frac{p}{2}}}{\epsilon^2}\left((\log d)^2 \log\left(\frac{d}{\epsilon}\log\frac{1}{\delta}\right) + \log\frac{1}{\delta}\right)$$

nonzero rows.

Proof. The proof is nearly identical to Theorem 5.2.1 except that the supporting theorems for the events on S differ. Specifically,

(1) By the remark after Theorem 1.2.12, S is a $(1/2)$-subspace-embedding matrix of A with probability at least $1 - \delta/2$;

(2) By Lemma 3.3.2 and Lemma 4.3.4, Lewis weights of SA are uniformly upper bounded by $2/\beta$ with probability at least $1 - \delta/2$.

All other parts of the proof are identical to the proof of Theorem 5.2.1, and we shall obtain that

$$\beta \gtrsim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2}\left(\log^2 d \log m + \log\frac{1}{\delta}\right).$$

By a Chernoff bound, we know that $m \lesssim \beta d$ with probability at least $1 - \delta$ and therefore it suffices to have

$$\beta \sim_p \frac{d^{\frac{p}{2}-1}}{\epsilon^2}\left(\log^2 d \log\left(\frac{d}{\epsilon}\log\frac{1}{\delta}\right) + \log\frac{1}{\delta}\right). \qquad \square$$

Bibliography

[1] Morteza Alamgir and Ulrike von Luxburg. Phase transition in the family of p-resistances. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 379–387, 2011.

[2] Josh Alman and Virginia Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 522–539. SIAM, 2021.

[3] Alexandr Andoni, Khanh Do Ba, Piotr Indyk, and David P. Woodruff. Efficient sketches for earth-mover distance, with applications. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*, pages 324–330, 2009.

[4] Alexandr Andoni, Robert Krauthgamer, and Krzysztof Onak. Streaming algorithms via precision sampling. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 363–372, 2011.

[5] J Bourgain and J Lindenstrauss. Distribution of points on spheres and approximation by zonotopes. *Israel Journal of Mathematics*, 64:25–31, 1988.

[6] Jean Bourgain, Joram Lindenstrauss, and Vitali Milman. Approximation of zonoids by zonotopes. *Acta Mathematica*, 162(1):73–141, 1989.

[7] Christos Boutsidis, David P. Woodruff, and Peilin Zhong. Optimal principal component analysis in distributed and streaming models. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 236–249, 2016.

[8] Vladimir Braverman, Stephen R. Chestnut, David P. Woodruff, and Lin F. Yang. Streaming space complexity of nearly all functions of one variable on frequency vectors. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS '16, page 261–276, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341912.

[9] Vladimir Braverman, Petros Drineas, Cameron Musco, Christopher Musco, Jalaj Upadhyay, David P. Woodruff, and Samson Zhou. Near optimal linear algebra in the online and sliding window models. In Sandy Irani, editor, *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 517–528. IEEE, 2020.

[10] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

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

[12] Xue Chen and Michal Derezinski. Query complexity of least absolute deviation regression via robust uniform convergence. In Mikhail Belkin and Samory Kpotufe, editors, *Proceedings of the 34th Annual Conference on Learning Theory, 2021*, volume 134 of *Proceedings of Machine Learning Research*, pages 1144–1179. PMLR, 2021.

[13] Xue Chen and Eric Price. Active regression via linear-sample sparsification. In Alina Beygelzimer and Daniel Hsu, editors, *Proceedings of the 32nd Annual Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 663–695. PMLR, 2019.

[14] Kenneth L. Clarkson and David P. Woodruff. Numerical linear algebra in the streaming model. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 205–214, 2009.

[15] 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.

[16] Michael B Cohen and Richard Peng. ℓ_p row sampling by Lewis weights. In *Proceedings of the 47th annual ACM symposium on Theory of computing*, pages 183–192, 2015.

[17] Michael B. Cohen, T.S. Jayram, and Jelani Nelson. Simple Analyses of the Sparse Johnson-Lindenstrauss Transform. In Raimund Seidel, editor, *1st Symposium on Simplicity in Algorithms (SOSA 2018)*, volume 61 of *OpenAccess Series in Informatics (OASIcs)*, pages 15:1–15:9, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-064-4.

[18] Michael B Cohen, Cameron Musco, and Jakub Pachocki. Online row sampling. *Theory of Computing*, 16(15):1–25, 2020. APPROX-RANDOM 2016 Special Issue.

[19] Graham Cormode and S. Muthukrishnan. An improved data stream summary: The count-min sketch and its applications. In *Proceedings of the Latin American Symposium on Theoretical Informatics (LATIN)*, pages 29–38. Springer, 2004.

[20] Anirban Dasgupta, Petros Drineas, Boulos Harb, Ravi Kumar, and Michael W Mahoney. Sampling algorithms and coresets for ℓ_p regression. *SIAM Journal on Computing*, 38(5):2060–2078, 2009.

[21] Lei Deng, Guoqi Li, Song Han, Luping Shi, and Yuan Xie. Model compression and hardware acceleration for neural networks: A comprehensive survey. *Proceedings of the IEEE*, 108(4):485–532, 2020.

[22] Petros Drineas and Michael W Mahoney. Lectures on randomized numerical linear algebra. *The Mathematics of Data*, 25(1), 2018.

[23] Petros Drineas, Michael W Mahoney, and Shan Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1127–1136, 2006.

[24] Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, and David P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *J. Mach. Learn. Res.*, 13(1):3475–3506, 2012.

[25] Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, and David P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(111):3475–3506, 2012.

[26] Benjamin Van Durme and Ashwin Lall. Streaming pointwise mutual information. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NIPS'09, page 1892–1900, Red Hook, NY, USA, 2009. Curran Associates Inc. ISBN 9781615679119.

[27] Abderrahim Elmoataz, Matthieu Toutain, and Daniel Tenbrinck. On the p-laplacian and ∞-laplacian on graphs with applications in image and data processing. *SIAM Journal on Imaging Sciences*, 8(4):2412–2451, 2015.

[28] Maryam Fazel, Yin Tat Lee, Swati Padmanabhan, and Aaron Sidford. Computing lewis weights to high precision. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2723–2742, 2022.

[29] David A. Freedman. On Tail Probabilities for Martingales. *The Annals of Probability*, 3(1):100–118, 1975.

[30] Alan M. Frieze, Ravi Kannan, and Santosh S. Vempala. Fast monte-carlo algorithms for finding low-rank approximations. *J. ACM*, 51(6):1025–1041, 2004.

[31] Aarshvi Gajjar, Christopher Musco, and Chinmay Hegde. Active learning for single neuron models with Lipschitz non-linearities. In *Procedings of the 26th International Conference on Artificial Intelligence and Statistics*, pages 4101–4113. PMLR, 2023.

[32] Aarshvi Gajjar, Xingyu Xu, Chinmay Hegde, and Christopher Musco. Improved bounds for agnostic active learning of single index models. In *RealML Workshop NeurIPS*, 2023.

[33] Aarshvi Gajjar, Wai Ming Tai, Xu Xingyu, Chinmay Hegde, Christopher Musco, and Yi Li. Agnostic active learning of single index models with linear sample complexity. In *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247, pages 1715–1754. PMLR, 2024.

[34] René Gonin and Arthur H. Money. *Nonlinear L_p-Norm Estimation*. CRC Press, 1989.

[35] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021.

[36] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[37] Sheng-Jun Huang, Yi Li, Yiming Sun, and Ying-Peng Tang. One-shot active learning based on Lewis weight sampling for multiple deep models. In *Proceedings of the International Conference on Learning Representations*, 2024.

[38] Arun Jambulapati, James R. Lee, Yang P. Liu, and Aaron Sidford. Sparsifying sums of norms. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1953–1962, 2023. doi: 10.1109/FOCS57990.2023.00119.

[39] Arun Jambulapati, James R Lee, Yang P Liu, and Aaron Sidford. Sparsifying generalized linear models. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 1665–1675, 2024.

[40] Shunhua Jiang, Binghui Peng, and Omri Weinstein. Dynamic least-squares regression. arXiv:2201.00228 [cs.DS], 2022.

[41] Qiuye Jin, Mingzhi Yuan, Qin Qiao, and Zhijian Song. One-shot active learning for image segmentation via contrastive learning and diversity-based sampling. *Knowledge-Based Systems*, 241:108278, 2022.

[42] William B Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26, 1984.

[43] William B. Johnson and Gideon Schechtman. Chapter 19 – Finite dimensional subspaces of L_p. In W. B. Johnson and J. Lindenstrauss, editors, *Handbook of the Geometry of Banach Spaces*, volume 1 of *Handbook of the Geometry of Banach Spaces*, pages 837–870. Elsevier Science B.V., 2001.

[44] Daniel M. Kane and Jelani Nelson. Sparser Johnson-Lindenstrauss transforms. *J. ACM*, 61(1), jan 2014. ISSN 0004-5411.

[45] Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. *SIAM J. Comput.*, 46(1):456–477, 2017.

[46] Alexander Koldobsky and Hermann König. Chapter 21 – Aspects of the isometric theory of Banach spaces. In W. B. Johnson and J. Lindenstrauss, editors, *Handbook of the Geometry of Banach Spaces*, volume 1 of *Handbook of the Geometry of Banach Spaces*, pages 837–870. Elsevier Science B.V., 2001.

[47] Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: isoperimetry and processes*, volume 23. Springer Science & Business Media, 1991.

[48] Roie Levin, Anish Prasad Sevekari, and David P. Woodruff. Robust subspace approximation in a stream. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 10706–10716, 2018.

[49] Yi Li, Ruosong Wang, Lin Yang, and Hanrui Zhang. Nearly linear row sampling algorithm for quantile regression. In *International Conference on Machine Learning*, pages 5979–5989. PMLR, 2020.

[50] Yi Li, Ruosong Wang, and David P. Woodruff. Tight bounds for the subspace sketch problem with applications. *SIAM J. Comput.*, 50(4):1287–1335, 2021.

[51] Yi Li, David Woodruff, and Taisuke Yasuda. Exponentially improved dimensionality reduction for ℓ_1: Subspace embeddings and independence testing. Accepted to *COLT* 2021. Full version available at arXiv:2104.12946 [cs.DS], 2021.

[52] Yi Li, Honghao Lin, and David P. Woodruff. The ℓ_p-subspace sketch problem in small dimensions with applications to support vector machines. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 850–877. SIAM, 2023.

[53] Yingyu Liang, Zhao Song, Mengdi Wang, Lin Yang, and Xin Yang. Sketching transformed matrices with applications to natural language processing. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, pages 467–481, 2020.

[54] Sepideh Mahabadi, Ilya P. Razenshteyn, David P. Woodruff, and Samson Zhou. Non-adaptive adaptive sampling on turnstile streams. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1251–1264, 2020.

[55] Ivan Markovsky. *Low-Rank Approximation: Algorithms, Implementation, Applications*. Communications and Control Engineering. Springer International Publishing AG, 2nd edition, 2019.

[56] Jiří Matoušek. Improved upper bounds for approximation by zonotopes. *Acta Mathematica*, pages 55–73, 1996.

[57] Gaurav Menghani. Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Computing Surveys*, 55(12):1–37, 2023.

[58] Riley Murray, James Demmel, Michael W Mahoney, N Benjamin Erichson, Maksim Melnichenko, Osman Asif Malik, Laura Grigori, Piotr Luszczek, Michał Dereziński, Miles E Lopes, et al. Randomized numerical linear algebra: A perspective on the field with an eye to software. *arXiv preprint arXiv:2302.11474*, 2023.

[59] Cameron Musco, Christopher Musco, David P. Woodruff, and Taisuke Yasuda. Active linear regression for ℓ_p norms and beyond. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022*, pages 744–753, 2022.

[60] Cameron Musco, Christopher Musco, David P. Woodruff, and Taisuke Yasuda. Active sampling for linear regression beyond the l_2 norm. arXiv:2111.04888v1 [cs.LG], 2022.

[61] Cameron Musco, Christopher Musco, David P. Woodruff, and Taisuke Yasuda. Active linear regression for ℓ_p norms and beyond. arXiv:2111.04888v4 [cs.LG], 2022.

[62] Jelani Nelson and Huy L. Nguyen. OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *2013 ieee 54th annual symposium on foundations of computer science*, pages 117–126. IEEE, 2013.

[63] Aditya Parulekar, Advait Parulekar, and Eric Price. L1 regression with Lewis weights subsampling. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2021, August 16-18, 2021*, volume 207, pages 49:1–49:21, 2021.

[64] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys*, 54(9):1–40, 2021.

[65] Carlos Riquelme, Ramesh Johari, and Baosen Zhang. Online active linear regression via thresholding. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 2506–2512, 2017.

[66] Sivan Sabato and Rémi Munos. Active regression by stratification. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 469–477, 2014.

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

[68] Gideon Schechtman. Tight embedding of subspaces of L_p in ℓ_p^n for even p. *Proceedings of the American Mathematical Society*, 139(12):4419–4421, 2011.

[69] Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison, 2009.

[70] Neta Shoham and Haim Avron. Experimental design for overparameterized learning with application to single shot deep active learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[71] Michel Talagrand. Embedding subspaces of L_1 into ℓ_1^N. *Proceedings of the American Mathematical Society*, 108(2):363–369, 1990.

[72] Michel Talagrand. Embedding subspaces of L_p in ℓ_p^N. In J. Lindenstrauss and V. Milman, editors, *Geometric Aspects of Functional Analysis*, pages 311–326, 1995.

[73] Ying-Peng Tang and Sheng-Jun Huang. Active learning for multiple target models. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, 2022.

[74] Joel A Tropp. Improved analysis of the subsampled randomized hadamard transform. *Advances in Adaptive Data Analysis*, 3(01n02):115–126, 2011.

[75] Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015. ISSN 1935-8237.

[76] Jalaj Upadhyay. Differentially private linear algebra in the streaming model. *arXiv preprint arXiv:1409.5414*, 2014.

[77] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*, volume 47. Cambridge University Press, 2018.

[78] Tom J Viering, Jesse H Krijthe, and Marco Loog. Nuclear discrepancy for single-shot batch active learning. *Machine Learning*, 108(8-9):1561–1599, 2019.

[79] Przemysław Wojtaszczyk. *Banach Spaces for Analysts*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1991.

[80] David P. Woodruff. Low rank approximation lower bounds in row-update streams. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 1781–1789, 2014.

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

[82] David P. Woodruff and Taisuke Yasuda. Online Lewis weight sampling. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 4622–4666. SIAM, 2023.

[83] David P Woodruff and Peilin Zhong. Distributed low rank approximation of implicit functions of a matrix. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, pages 847–858. IEEE, 2016.

[84] Yazhou Yang and Marco Loog. Single shot active learning using pseudo annotators. *Pattern Recognition*, 89:22–31, 2019.

[85] Taisuke Yasuda. *Algorithms for Matrix Approximation: Sketching, Sampling, and Sparse Optimization*. PhD thesis, Carnegie Mellon University, 2024.