Sketching as a Tool for Numerical Linear Algebra *

David P. Woodruff
IBM Research Almaden
dpwoodru@us.ibm.com

November 18, 2014

Abstract

This survey highlights the recent advances in algorithms for numerical linear algebra that have come from the technique of linear sketching, whereby given a matrix, one first compressed it to a much smaller matrix by multiplying it by a (usually) random matrix with certain properties. Much of the expensive computation can then be performed on the smaller matrix, thereby accelerating the solution for the original problem. In this survey we consider least squares as well as robust regression problems, low rank approximation, and graph sparsification. We also discuss a number of variants of these problems. Finally, we discuss the limitations of sketching methods.

*Version appearing as a monograph in NOW Publishers "Foundations and Trends in Theoretical Computer Science" series, Vol 10, Issue 1–2, 2014, pp 1–157

Contents

1 Introduction

To give the reader a flavor of results in this survey, let us first consider the classical linear regression problem. In a special case of this problem one attempts to "fit" a line through a set of given points as best as possible.

For example, the familiar Ohm's law states that the voltage V is equal to the resistance R times the electrical current I, or $V = R \cdot I$. Suppose one is given a set of n example volate-current pairs (v_j, i_j) but does not know the underlying resistance. In this case one is attempting to find the unknown slope of a line through the origin which best fits these examples, where best fits can take on a variety of different meanings.

More formally, in the standard setting there is one *measured variable* b, in the above example this would be the voltage, and a set of d *predictor variables* $a_1, \ldots, a_d$. In the above example $d = 1$ and the single predictor variable is the electrical current. Further, it is assumed that the variables are linearly related up to a noise variable, that is $b = x_0 + a_1 x_1 + \cdots + a_d x_d + \gamma$, where $x_0, x_1, \ldots, x_d$ are the coefficients of a hyperplane we are trying to learn (which does not go through the origin if $x_0 \neq 0$), and γ is a random variable which may be adversarially chosen, or may come from a distribution which we may have limited or no information about. The x_i are also known as the *model parameters*. By introducing an additional predictor variable a_0 which is fixed to 1, we can in fact assume that the unknown hyperplane goes through the origin, that is, it is an unknown subspace of codimension 1. We will thus assume that $b = a_1 x_1 + \cdots + a_d x_d + \gamma$ and ignore the affine component throughout.

In an experiment one is often given n observations, or n $(d+1)$-tuples $(a_{i,1}, \ldots, a_{i,d}, b_i)$, for $i = 1, 2, \ldots, n$. It is more convenient now to think of the problem in matrix form, where one is given an $n \times d$ matrix $\mathbf{A}$ whose rows are the values of the predictor variables in the d examples, together with an $n \times 1$ column vector $\mathbf{b}$ whose entries are the corresponding observations, and the goal is to output the coefficient vector $\mathbf{x}$ so that $\mathbf{Ax}$ and $\mathbf{b}$ are close in whatever the desired sense of closeness may mean. Notice that as one ranges over all $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{Ax}$ ranges over all linear combinations of the d columns of $\mathbf{A}$, and therefore defines a d-dimensional subspace of $\mathbb{R}^n$, which we refer to as the column space of $\mathbf{A}$. Therefore the regression problem is equivalent to finding the vector $\mathbf{x}$ for which $\mathbf{Ax}$ is the closest point in the column space of $\mathbf{A}$ to the observation vector $\mathbf{b}$.

Much of the focus of this survey will be on the over-constrained case, in which the number n of examples is much larger than the number d of predictor variables. Note that in this case there are more constraints than

unknowns, and there need not exist a solution $\mathbf{x}$ to the equation $\mathbf{Ax} = \mathbf{b}$.

Regarding the measure of fit, or closeness of $\mathbf{Ax}$ to $\mathbf{b}$, one of the most common is the least squares method, which seeks to find the closest point in Euclidean distance, i.e.,

$$\text{argmin}_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2 = \sum_{i=1}^{n} (b_i - \langle \mathbf{A}_{i,*}, \mathbf{x} \rangle)^2,$$

where $\mathbf{A}_{i,*}$ denotes the i-th row of $\mathbf{A}$, and b_i the i-th entry of the vector $\mathbf{b}$. This error measure has a clear geometric interpretation, as the optimal $\mathbf{x}$ satisfies that $\mathbf{Ax}$ is the standard Euclidean projection of $\mathbf{b}$ onto the column space of $\mathbf{A}$. Because of this, it is possible to write the solution for this problem in a closed form. That is, necessarily one has $\mathbf{A}^T\mathbf{Ax}^* = \mathbf{A}^T\mathbf{b}$ for the optimal solution $\mathbf{x}^*$ by considering the gradient at a point $\mathbf{x}$, and observing that in order for it to be 0, that is for $\mathbf{x}$ to be a minimum, the above equation has to hold. The equation $\mathbf{A}^T\mathbf{Ax}^* = \mathbf{A}^T\mathbf{b}$ is known as the *normal equation*, which captures that the line connecting $\mathbf{Ax}^*$ to $\mathbf{b}$ should be perpendicular to the columns spanned by $\mathbf{A}$. If the columns of $\mathbf{A}$ are linearly independent, $\mathbf{A}^T\mathbf{A}$ is a full rank $d \times d$ matrix and the solution is therefore given by $\mathbf{x}^* = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. Otherwise, there are multiple solutions and a solution $\mathbf{x}^*$ of minimum Euclidean norm is given by $\mathbf{x}^* = \mathbf{A}^\dagger\mathbf{b}$, where $\mathbf{A}^\dagger$ is the Moore-Penrose pseudoinverse of $\mathbf{A}$. Recall that if $\mathbf{A} = \mathbf{U\Sigma V}^T$ is the singular value decomposition (SVD) of $\mathbf{A}$, where $\mathbf{U}$ is $n \times d$ with orthonormal columns, $\mathbf{\Sigma}$ is a diagonal $d \times d$ matrix with non-negative non-increasing diagonal entries, and $\mathbf{V}^T$ is a $d \times d$ matrix with orthonormal rows, then the Moore-Penrose pseudoinverse of $\mathbf{A}$ is the $d \times n$ matrix $\mathbf{V\Sigma}^\dagger\mathbf{U}^T$, where $\mathbf{\Sigma}^\dagger$ is a $d \times d$ diagonal matrix with $\mathbf{\Sigma}^\dagger_{i,i} = 1/\mathbf{\Sigma}_{i,i}$ if $\mathbf{\Sigma}_{i,i} > 0$, and is 0 otherwise.

The least squares measure of closeness, although popular, is somewhat arbitrary and there may be better choices depending on the application at hand. Another popular choice is the method of least absolute deviation, or ℓ_1-regression. Here the goal is to instead find $\mathbf{x}^*$ so as to minimize

$$\|\mathbf{Ax} - \mathbf{b}\|_1 = \sum_{i=1}^{n} |\mathbf{b}_i - \langle \mathbf{A}_{i,*}, \mathbf{x} \rangle|.$$

This measure is known to be less sensitive to outliers than the least squares measure. The reason for this is that one squares the value $\mathbf{b}_i - \langle \mathbf{A}_{i,*}, \mathbf{x} \rangle$ in the least squares cost function, while one only takes its absolute value in the least absolute deviation cost function. Thus, if $\mathbf{b}_i$ is significantly

larger (or smaller) than $\langle \mathbf{A}_{i,*}, \mathbf{x} \rangle$ for the i-th observation, due, e.g., to large measurement noise on that observation, this requires the sought hyperplane $\mathbf{x}$ to be closer to the i-th observation when using the least squares cost function than when using the least absolute deviation cost function. While there is no closed-form solution for least absolute deviation regression, one can solve the problem up to machine precision in polynomial time by casting it as a linear programming problem and using a generic linear programming algorithm.

The problem with the above solutions is that on massive data sets, they are often too slow to be of practical value. Using näive matrix multiplication, solving the normal equations for least squares would take at least $n \cdot d^2$ time. For least absolute deviation regression, when casting the problem as a linear program one needs to introduce $O(n)$ variables (these are needed to enforce the absolute value constraints) and $O(n)$ constraints, and generic solvers would take poly(n) time for an polynomial in n which is at least cubic. While these solutions are polynomial time, they are prohibitive for large values of n.

The starting point of this survey is a beautiful work by Tamás Sarlós [105] which observed that one could use *sketching techniques* to improve upon the above time complexities, if one is willing to settle for a randomized approximation algorithm. Here, one relaxes the problem to finding a vector $\mathbf{x}$ so that $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_p \leq (1+\varepsilon)\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_p$, where $\mathbf{x}^*$ is the optimal hyperplane, with respect to the p-norm, for p either 1 or 2 as in the discussion above. Moreover, one allows the algorithm to fail with some small probability δ, which can be amplified by independent repetition and taking the best hyperplane found.

While sketching techniques will be described in great detail in the following chapters, we give a glimpse of what is to come below. Let $r \ll n$, and suppose one chooses a $r \times n$ random matrix $\mathbf{S}$ from a certain distribution on matrices to be specified. Consider the following algorithm for least squares regression:

1. Sample a random matrix $\mathbf{S}$.
2. Compute $\mathbf{S} \cdot \mathbf{A}$ and $\mathbf{S} \cdot \mathbf{b}$.
3. Output the exact solution x to the regression problem $\min_{\mathbf{x}} \|(\mathbf{S}\mathbf{A})\mathbf{x} - (\mathbf{S}\mathbf{b})\|_2$.

Let us highlight some key features of this algorithm. First, notice that it is a *black box* reduction, in the sense that after computing $\mathbf{S} \cdot \mathbf{A}$ and

$\mathbf{S} \cdot \mathbf{b}$, we then solve a smaller instance of least squares regression, replacing the original number n of observations with the smaller value of r. For r sufficiently small, we can then afford to carry out step 3, e.g., by computing and solving the normal equations as described above.

The most glaring omission from the above algorithm is which random familes of matrices $\mathbf{S}$ will make this procedure work, and for what values of r. Perhaps one of the simplest arguments is the following. Suppose $r = \Theta(d/\varepsilon^2)$ and $\mathbf{S}$ is a $r \times n$ matrix of i.i.d. normal random variables with mean zero and variance $1/r$, denoted $N(0, 1/r)$. Let $\mathbf{U}$ be an $n \times (d+1)$ matrix with orthonormal columns for which the column space of $\mathbf{U}$ is equal to the column space of $[\mathbf{A}, \mathbf{b}]$, that is, the space spanned by the columns of $\mathbf{A}$ together with the vector $\mathbf{b}$.

Consider the product $\mathbf{S} \cdot \mathbf{U}$. By 2-stability of the normal distribution, i.e., if $\mathbf{A} \sim N(0, \sigma_1^2)$ and $\mathbf{B} \sim N(0, \sigma_2^2)$, then $\mathbf{A} + \mathbf{B} \sim N(0, \sigma_1^2 + \sigma_2^2)$, each of the entries of $\mathbf{S} \cdot \mathbf{U}$ is distributed as $N(0, 1/r)$ (recall that the column norms of $\mathbf{U}$ are equal to 1). The entries in different rows of $\mathbf{S} \cdot \mathbf{U}$ are also independent since the rows of $\mathbf{S}$ are independent. The entries in a row are also independent by rotational invarance of the normal distribution, that is, if $\mathbf{g} \sim N(0, \mathbf{I}_n/r)$ is an n-dimensional vector of normal random variables and $\mathbf{U}_{*,1}, \ldots, \mathbf{U}_{*,d}$ are orthogonal vectors, then $\langle \mathbf{g}, \mathbf{U}_{*,1} \rangle, \langle \mathbf{g}, \mathbf{U}_{*,2} \rangle, \ldots, \langle \mathbf{g}, \mathbf{U}_{*,d+1} \rangle$ are independent. Here $\mathbf{I}_n$ is the $n \times n$ identity matrix (to see this, by rotational invariance, these $d+1$ random variables are equal in distribution to $\langle \mathbf{g}, \mathbf{e}_1 \rangle, \langle \mathbf{g}, \mathbf{e}_2 \rangle, \ldots, \langle \mathbf{g}, \mathbf{e}_{d+1} \rangle$, where $\mathbf{e}_1, \ldots, \mathbf{e}_{d+1}$ are the standard unit vectors, from which independence follows since the coordinates of g are independent).

It follows that $\mathbf{S} \cdot \mathbf{U}$ is an $r \times (d+1)$ matrix of i.i.d. $N(0, 1/r)$ random variables. For $r = \Theta(d/\varepsilon^2)$, it is well-known that with probability $1 - \exp(-d)$, all the singular values of $\mathbf{S} \cdot \mathbf{U}$ lie in the interval $[1-\varepsilon, 1+\varepsilon]$. This can be shown by arguing that for any fixed vector $\mathbf{x}$, $\|\mathbf{S} \cdot \mathbf{U}\mathbf{x}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{x}\|_2^2$ with probability $1 - \exp(-d)$, since, by rotational invariance of the normal distribution, $\mathbf{S} \cdot \mathbf{U}\mathbf{x}$ is a vector of r i.i.d. $N(0, \|x\|_2^2)$ random variables, and so one can apply a tail bound for $\|\mathbf{S} \cdot \mathbf{U}\mathbf{x}\|_2^2$, which itself is a χ^2-random variable with r degrees of freedom. The fact that all singular values of $\mathbf{S} \cdot \mathbf{U}$ lie in $[1-\varepsilon, 1+\varepsilon]$ then follows by placing a sufficiently fine net on the unit sphere and applying a union bound to all net points; see, e.g., Theorem 2.1 of [104] for further details.

Hence, for all vectors $\mathbf{y}$, $\|\mathbf{S}\mathbf{U}\mathbf{y}\|_2 = (1 \pm \varepsilon)\|\mathbf{U}\mathbf{y}\|_2$. But now consider the regression problem $\min_{\mathbf{x}} \|(\mathbf{S}\mathbf{A})\mathbf{x} - (\mathbf{S}\mathbf{b})\|_2 = \min_{\mathbf{x}} \|\mathbf{S}(\mathbf{A}\mathbf{x} - \mathbf{b})\|_2$. For each vector x, $\mathbf{A}\mathbf{x} - \mathbf{b}$ is in the column space of $\mathbf{U}$, and therefore by the previous paragraph, $\|\mathbf{S}(\mathbf{A}\mathbf{x} - \mathbf{b})\|_2 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$. It follows that by solving the

regression problem $\min_x \|(\mathbf{SA})\mathbf{x}-(\mathbf{Sb})\|_2$, we obtain a $(1+\varepsilon)$-approximation to the original regression problem with probability $1-\exp(-d)$.

The above technique of replacing $\mathbf{A}$ by $\mathbf{S}\cdot\mathbf{A}$ is known as a sketching technique and $\mathbf{S}\cdot\mathbf{A}$ is referred to as a (linear) sketch of $\mathbf{A}$. While the above is perhaps the simplest instantiation of sketching, notice that it does not in fact give us a faster solution to the least squares regression problem. This is because, while solving the regression problem $\min_{\mathbf{x}} \|(\mathbf{SA})\mathbf{x}-(\mathbf{Sb})\|_2$ can now be done näively in only $O(rd^2)$ time, which no longer depends on the large dimension n, the problem is that $\mathbf{S}$ is a dense matrix and computing $\mathbf{S}\cdot\mathbf{A}$ may now be too slow, taking $\Theta(nrd)$ time.

Thus, the bottleneck in the above algorithm is the time for matrix-matrix multiplication. Tamás Sarlós observed [105] that one can in fact choose $\mathbf{S}$ to come from a much more structured random family of matrices, called fast Johnson-Lindenstrauss transforms [2]. These led to roughly $O(nd\log d)+\text{poly}(d/\varepsilon)$ time algorithms for the least squares regression problem. Recently, Clarkson and Woodruff [27] improved upon the time complexity of this algorithm to obtain *optimal* algorithms for approximate least squares regression, obtaining $O(\text{nnz}(\mathbf{A}))+\text{poly}(d/\varepsilon)$ time, where $\text{nnz}(\mathbf{A})$ denotes the number of non-zero entries of the matrix $\mathbf{A}$. We call such algorithms input-sparsity algorithms, as they exploit the number of non-zero entries of $\mathbf{A}$. The $\text{poly}(d/\varepsilon)$ factors were subsequently optimized in a number of papers [92, 97, 18], leading to optimal algorithms even when $\text{nnz}(\mathbf{A})$ is not too much larger than d.

In parallel, work was done on reducing the dependence on ε in these algorithms from polynomial to polylogarithmic. This started with work of Rokhlin and Tygert [103] (see also the Blendenpik algorithm [8]), and combined with the recent input sparsity algorithms give a running time of $O(\text{nnz}(\mathbf{A})\log(1/\varepsilon))+\text{poly}(d)$ for least squares regression [27]. This is significant for high precision applications of least squares regression, for example, for solving an equation of the form $\mathbf{A}^T\mathbf{Ax}=\mathbf{A}^T\mathbf{b}$. Such equations frequently arise in interior point methods for linear programming, as well as iteratively reweighted least squares regression, which is a subroutine for many important problems, such as logistic regression; see [94] for a survey of such techniques for logistic regression. In these examples $\mathbf{A}$ is often formed from the Hessian of a Newton step in an iteration. It is clear that such an equation is just a regression problem in disguise (in the form of the normal equations), and the (exact) solution of $\text{argmin}_{\mathbf{x}}\|\mathbf{Ax}-\mathbf{b}\|_2$ provides such a solution. By using high precision approximate regression one can speed up the iterations in such algorithms.

Besides least squares regression, related sketching techniques have also

been instrumental in providing better robust ℓ_1-regression, low rank approximation, and graph sparsifiers, as well as a number of variants of these problems. We will cover these applications each in more detail.

Roadmap: In the next chapter we will discuss least squares regression in full detail, which includes applications to constrained and structured regression. In Chapter 3, we will then discuss ℓ_p-regression, including least absolute deviation regression. In Chapter 4 we will dicuss low rank approximation, while in Chapter 5, we will discuss graph sparsification. In Chapter 6, we will discuss the limitations of sketching techniques. In Chapter 7, we will conclude and briefly discuss a number of other directions in this area.

2 Subspace Embeddings and Least Squares Regression

We start with the classical least squares regression problem, which is the following. We are given an $n \times d$ matrix $\mathbf{A}$, which is typically overconstrained, that is, $n \gg d$, together with an $n \times 1$ vector $\mathbf{b}$, and we would like to find an $\mathbf{x} \in \mathbb{R}^d$ which minimizes $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$. Since the problem is overconstrained, there need not exist a vector $\mathbf{x}$ for which $\mathbf{A}\mathbf{x} = \mathbf{b}$. We relax the problem and instead allow for outputting a vector $\mathbf{x}'$ for which with probability .99,

$$\|\mathbf{A}\mathbf{x}' - \mathbf{b}\|_2 \leq (1+\varepsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2.$$

We are interested in fast solutions to this problem, which we present in §2.5.

Chapter Overview: In §2.1 we introduce the notion of an ℓ_2-subspace embedding, which is crucial to many of the applications in this book. In this chapter we will focus on its application to least squares regression. We show several different randomized constructions which vary in the time it takes to construct and apply them, as well as the dimension which they embed into. These constructions turn out to be oblivious to the data set. In §2.2 we introduce a primitive called matrix product, which is a primitive for performing approximate matrix multiplication. Using this primitive we will show how to construct an ℓ_2-subspace embedding. The primitive will also play a role in §2.5 in solving regression with a linear in $1/\varepsilon$ dependence on the accuracy parameter ε, as well as in §4.1 on low rank matrix approximation. In §2.3 we present a trick which takes any constant probability of success subspace embedding and shows how to obtain a high probability success subspace embedding. Thus, to some extent our earlier treatment of constant subspace embeddings is justified. In §2.4 we present a completely different way of achieving a subspace embedding, which is non-oblivious and is obtained by sampling rows of a matrix proportional to their so-called leverage scores. In §2.5 we present a black box application of subspace embeddings to the least squares regression problem. In §2.6 we show how to use subspace embeddings in a different way to solve least squares regression, leading to an algorithm with only a logarithmic dependence on the error parameter $1/\varepsilon$. This method, while it has a much better dependence on $1/\varepsilon$, does require multiple passes over the data unlike the method of §2.5. Finally, in §2.7 we show that for regression instances which possess additional structure, such as those that arise in polynomial fitting problems, one can apply subspace embeddings even faster than via generic ways presented before.

2.1 Subspace embeddings

We start with the basic notion of an ℓ_2-subspace embedding for the column space of an $n \times d$ matrix $\mathbf{A}$. As we will see, this will be a powerful hammer for solving least squares regression. Throughout, for non-negative real numbers a and b, we use the notation $a = (1 \pm \varepsilon)b$ if $a \in [(1-\varepsilon)b, (1+\varepsilon)b]$.

Definition 1 *A $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for the column space of an $n \times d$ matrix $\mathbf{A}$ is a matrix $\mathbf{S}$ for which for all $\mathbf{x} \in \mathbb{R}^d$*

$$\|\mathbf{SAx}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{Ax}\|_2^2.$$

We will often abuse notation and say that $\mathbf{S}$ is an ℓ_2-subspace embedding for $\mathbf{A}$ itself, even though it should be understood from the definition that this property does not depend on a particular basis for the representation of the column space of $\mathbf{A}$.

Notice that if $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$, then it is also a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{U}$, where $\mathbf{U}$ is an orthonormal basis for the column space of $\mathbf{A}$. This is because the sets $\{\mathbf{Ax} \mid \mathbf{x} \in \mathbb{R}^d\}$ and $\{\mathbf{Uy} \mid \mathbf{y} \in \mathbb{R}^t\}$ are equal, where t is the rank of $\mathbf{A}$. Hence, we could without loss of generality assume that $\mathbf{A}$ has orthonormal columns. With this interpretation, the requirement of Definition 1 becomes

$$\|\mathbf{SUy}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{Uy}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{y}\|_2^2,$$

where the final equality holds since $\mathbf{U}$ has orthonormal columns. If this requirement is satisfied for unit vectors $\mathbf{y}$, then it is satisfied for all vectors $\mathbf{y}$ by scaling (since $\mathbf{S}$ is a linear map), so the requirement of Definition 1 can be further simplified to

$$\|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU}\|_2 \leq \varepsilon, \tag{1}$$

that is, the operator norm $\sup_{y \text{ such that } \|y\|_2=1} \|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU}\|_2$, should be at most ε. Here, $\mathbf{I}_d$ is the $d \times d$ identity matrix.

There are various goals of subspace embeddings. Two of the main goals are finding a matrix $\mathbf{S}$ with a small number of rows. Another goal is to be able to compute $\mathbf{S} \cdot \mathbf{A}$ quickly, as this is often a bottleneck in applications.

There are a number of ways of constructing ℓ_2-subspace embeddings which achieve various tradeoffs. One particularly useful form of an ℓ_2-subspace embedding is an *oblivious* ℓ_2-subspace embedding.

Definition 2 *Suppose Π is a distribution on $r \times n$ matrices $\mathbf{S}$, where r is a function of n, d, ε, and δ. Suppose that with probability at least $1 - \delta$, for*

any fixed $n \times d$ matrix $\mathbf{A}$, a matrix $\mathbf{S}$ drawn from distribution Π has the property that $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$. Then we call $\mathbf{\Pi}$ an (ε, δ) oblivious ℓ_2-subspace embedding.

Definition 2 will be used in applications throughout this book, and sometimes for convenience we will drop the word oblivious.

We do want to note that there are other ways of constructing subspace embeddings though, such as through sampling the rows of $\mathbf{A}$ via a certain distribution and reweighting them. This is called Leverage Score Sampling [42, 46, 45, 43], which will be discussed later in the chapter. This also turns out to have a number of applications, for example to CUR decompositions of a matrix discussed in §4.2. Note that this way of constructing subspace embeddings is desirable in that it gives an actual "representative" subset of rows of $\mathbf{A}$ which form a subspace embedding - this is often called a *coreset*. Such representations can sometimes lead to better data interpretability, as well as preserving sparsity. While we do discuss this kind of sampling to some extent, our main focus will be on sketching. The reader is encouraged to look at the survey by Mahoney for more details on sampling-based approaches [85]. See also [78] and [31] for state of the art subspace embeddings based on this approach.

Returning to Definition 2, the first usage of this in the numerical linear algebra community, to the best of our knowledge, was done by Sárlos, who proposed using Fast Johnson Lindenstrauss transforms to provide subspace embeddings. We follow the exposition in Sarlós for this [105].

Definition 3 *A random matrix $\mathbf{S} \in \mathbb{R}^{k \times n}$ forms a Johnson-Lindenstrauss transform with parameters ε, δ, f, or JLT(ε, δ, f) for short, if with probability at least $1 - \delta$, for any f-element subset $V \subset \mathbb{R}^n$, for all $\mathbf{v}, \mathbf{v}' \in V$ it holds that $|\langle \mathbf{Sv}, \mathbf{Sv}' \rangle - \langle \mathbf{v}, \mathbf{v}' \rangle| \leq \varepsilon \|\mathbf{v}\|_2 \|\mathbf{v}'\|_2$.*

Note when $\mathbf{v} = \mathbf{v}'$ we obtain the usual statement that $\|\mathbf{Sv}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{v}\|_2^2$. It turns out that if we scale all $\mathbf{v}, \mathbf{v}' \in V$ so that they are unit vectors, we could alternatively require $\|\mathbf{Sv}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{v}\|_2^2$ and $\|\mathbf{S}(\mathbf{v} + \mathbf{v}')\|_2^2 = (1 \pm \varepsilon)\|\mathbf{v} + \mathbf{v}'\|_2^2$ for all $\mathbf{v}, \mathbf{v}' \in V$. That is, the requirement of the definition could be based on norms rather than inner products. To see that this impies the statement above, we have

$$\begin{aligned} \langle \mathbf{Sv}, \mathbf{Sv}' \rangle &= (\|\mathbf{S}(\mathbf{v} + \mathbf{v}')\|_2^2 - \|\mathbf{Sv}\|_2^2 - \|\mathbf{Sv}'\|_2^2)/2 \\ &= ((1 \pm \varepsilon)\|\mathbf{v} + \mathbf{v}'\|_2^2 - (1 \pm \varepsilon)\|\mathbf{v}\|_2^2 - (1 \pm \varepsilon)\|\mathbf{v}'\|_2^2) \\ &= \langle \mathbf{v}, \mathbf{v}' \rangle \pm O(\varepsilon), \end{aligned}$$

which implies all inner products are preserved up to ε by rescaling ε by a constant.

There are many constructions of Johnson-Lindenstrauss transforms, possibly the simplest is given by the following theorem.

Theorem 4 *(see e.g., [62]) Let $0 < \varepsilon, \delta < 1$ and $\mathbf{S} = \frac{1}{\sqrt{k}}\mathbf{R} \in \mathbb{R}^{k \times n}$ where the entries $\mathbf{R}_{i,j}$ of $\mathbf{R}$ are independent standard normal random variables. Then if $k = \Omega(\varepsilon^{-2}\log(f/\delta))$, then $\mathbf{S}$ is a JLT(ε, δ, f).*

We will see a proof of Theorem 4 in Lemma 18.

We show how Theorem 4 can be used to provide an ℓ_2-subspace embedding. To do so, we need the concept of an ε-net. Let $\mathcal{S} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = \mathbf{A}\mathbf{x}$ for some $\mathbf{x} \in \mathbb{R}^d$ and $\|\mathbf{y}\|_2 = 1\}$. We seek a finite subset of $\mathcal{S}$, denoted $\mathcal{N}$ so that if

$$\langle \mathbf{S}\mathbf{w}, \mathbf{S}\mathbf{w}' \rangle = \langle \mathbf{w}, \mathbf{w}' \rangle \pm \varepsilon \text{ for all } \mathbf{w}, \mathbf{w}' \in \mathcal{N}, \tag{2}$$

then $\|\mathbf{S}\mathbf{y}\|_2 = (1 \pm \varepsilon)\|\mathbf{y}\|_2$ for all $\mathbf{y} \in \mathcal{S}$.

By an argument of [6, 47, 84], it suffices to choose $\mathcal{N}$ so that for all $\mathbf{y} \in \mathcal{S}$, there exists a vector $\mathbf{w} \in \mathcal{N}$ for which $\|\mathbf{y} - \mathbf{w}\|_2 \leq 1/2$. We will refer to $\mathcal{N}$ as a (1/2)-net for $\mathcal{S}$.

To see that $\mathcal{N}$ suffices, if $\mathbf{y}$ is a unit vector, then we can write

$$\mathbf{y} = \mathbf{y}^0 + \mathbf{y}^1 + \mathbf{y}^2 + \cdots, \tag{3}$$

where $\|\mathbf{y}^i\| \leq \frac{1}{2^i}$ and $\mathbf{y}^i$ is a scalar multiple of a vector in $\mathcal{N}$. This is because we can write $\mathbf{y} = \mathbf{y}^0 + (\mathbf{y} - \mathbf{y}^0)$ where $\mathbf{y}^0 \in \mathcal{N}$ and $\|\mathbf{y} - \mathbf{y}^0\| \leq 1/2$ by the definition of $\mathcal{N}$. Then, $\mathbf{y} - \mathbf{y}^0 = \mathbf{y}^1 + ((\mathbf{y} - \mathbf{y}^0) - \mathbf{y}^1)$ where $\mathbf{y}^1 \in \mathcal{N}$ and

$$\|\mathbf{y} - \mathbf{y}^0 - \mathbf{y}^1\|_2 \leq \frac{\|\mathbf{y} - \mathbf{y}^0\|_2}{2} \leq \frac{1}{4}.$$

The expansion in (3) then follows by induction. But then,

$$\begin{aligned}
\|\mathbf{S}\mathbf{y}\|_2^2 &= \|\mathbf{S}(\mathbf{y}^0 + \mathbf{y}^1 + \mathbf{y}^2 + \cdots)\|_2^2 \\
&= \sum_{0 \leq i < j < \infty} \|\mathbf{S}\mathbf{y}^i\|_2^2 + 2\langle \mathbf{S}\mathbf{y}^i, \mathbf{S}\mathbf{y}^j \rangle \\
&= \left(\sum_{0 \leq i < j < \infty} \|\mathbf{y}^i\|_2^2 + 2\langle \mathbf{y}^i, \mathbf{y}^j \rangle \right) \pm 2\varepsilon \left(\sum_{0 \leq i \leq j < \infty} \|\mathbf{y}^i\|_2 \|\mathbf{y}^j\|_2 \right) \\
&= 1 \pm O(\varepsilon),
\end{aligned}$$

where the first equality follows by (3), the second equality follows by expanding the square, the third equality follows from (2), and the fourth equality is what we want (after rescaling ε by a constant factor).

We show the existence of a small (1/2)-net $\mathcal{N}$ via a standard argument.

Lemma 5 *For any $0 < \gamma < 1$, there exists a γ-net $\mathcal{N}$ of $\mathcal{S}$ for which $|\mathcal{N}| \leq (1 + 4/\gamma)^d$.*

Proof: For $t = \text{rank}(\mathbf{A}) \leq d$, we can equivalently express $\mathcal{S}$ as

$$\mathcal{S} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = \mathbf{U}\mathbf{x} \text{ for some } \mathbf{x} \in \mathbb{R}^t \text{ and } \|\mathbf{y}\|_2 = 1\},$$

where $\mathbf{U}$ has orthonormal columns and the same column space as $\mathbf{A}$.

We choose a $\gamma/2$-net $\mathcal{N}'$ of the unit sphere $\mathcal{S}^{t-1}$, where the $\gamma/2$ net has size $(1 + 4/\gamma)^t$. The intuition for this choice is that $\mathbf{U}$ provides an isometry when operating on $\mathcal{S}^{t-1}$, and so a net for $\mathcal{S}^{t-1}$ will give us a net for the image of $\mathcal{S}^{t-1}$ under $\mathbf{U}$.

This can be done by choosing a maximal set $\mathcal{N}'$ of points on $\mathcal{S}^{t-1}$ so that no two points are within distance $\gamma/2$ from each other. It follows that the balls of radius $\gamma/4$ centered at these points are disjoint, but on the other hand they are all contained in the ball of radius $1 + \gamma/4$ centered at the origin. The volume of the latter ball is a factor $(1+\gamma/4)^t/(\gamma/4)^t$ larger than the smaller balls, which implies $|\mathcal{N}'| \leq (1 + 4/\gamma)^t$. See, e.g., [89] for more details.

Define $\mathcal{N} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = \mathbf{U}\mathbf{x} \text{ for some } \mathbf{x} \in \mathcal{N}'\}$. Since the columns of $\mathbf{U}$ are orthonormal, if there were a point $\mathbf{U}\mathbf{x} \in \mathcal{S}$ for which there were no point $\mathbf{y} \in \mathcal{N}$ with $\|\mathbf{y} - \mathbf{U}\mathbf{x}\|_2 \leq \gamma$, then $\mathbf{x}$ would be a point in $\mathcal{S}^{k-1}$ for which there is no point $\mathbf{z} \in \mathcal{N}'$ with $\|\mathbf{x} - \mathbf{z}\|_2 \leq \gamma$, a contradiction. ■

It follows by setting $V = \mathcal{N}$ and $f = 9^d$ in Theorem 4, we can then apply Lemma 5 and (2) to obtain the following theorem. Note that the net size does not depend on ε, since we just need a 1/2-net for the argument, even though the theorem holds for general ε.

Theorem 6 *Let $0 < \varepsilon, \delta < 1$ and $\mathbf{S} = \frac{1}{\sqrt{k}}\mathbf{R} \in \mathbb{R}^{k \times n}$ where the entries $\mathbf{R}_{i,j}$ of $\mathbf{R}$ are independent standard normal random variables. Then if $k = \Theta((d + \log(1/\delta))\varepsilon^{-2})$, then for any fixed $n \times d$ matrix $\mathbf{A}$, with probability $1 - \delta$, $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$, that is, simultaneously for all $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{S}\mathbf{A}\mathbf{x}\|_2 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{x}\|_2$. Here $C > 0$ is an absolute constant.*

It turns out, as we will see in Chapter 6, that Theorem 6 provides the optimal number of rows of $\mathbf{S}$ up to a constant factor, namely $\Theta(k\varepsilon^{-2})$. This is true of any oblivious $(1 \pm \varepsilon)$ ℓ_2-subspace embedding, even those achieving only a constant probability of providing an ℓ_2-subspace embedding of $\mathbf{A}$ with constant probability.

After Theorem 4 was discovered, there were a number of followups. For instance, it was shown by Achlioptas that one can replace $\mathbf{R}$ in Theorem 4 with a matrix of i.i.d. sign random variables [1], that is, each entry is independently set to 1 or -1 with probability $1/2$. Further, Achlioptas showed that one can change the distribution so that for the same value of k, one can set each entry in $\mathbf{R}$ independently to be 1 with probability $1/6$, -1 with probability $1/6$, and 0 with probability $2/3$. The latter is important since it results in a sparse matrix $\mathbf{S}$, for which one can then compute $\mathbf{S} \cdot \mathbf{x}$ for a vector $\mathbf{x} \in \mathbb{R}^n$ more quickly. A breakthrough was made by Dasgupta, Kumar, and Sárlos [33] who showed that it suffices for each column of $\mathbf{S}$ to have only $\varepsilon^{-1}\text{poly}(\log(f/\delta))$ non-zero entries per column. Note that if the $\text{poly}(\log f/\delta)$ term is much smaller than ε^{-1}, this is a significant improvement over the $\Omega(\varepsilon^{-2}\log(f/\delta))$ number of non-zero entries per column achieved by previous schemes. The $\varepsilon^{-1}\text{poly}(\log(f/\delta))$ sparsity was later optimized by Kane and Nelson [67], who got $O(\varepsilon^{-1}\log(f/\delta))$ non-zero entries per column. The latter was shown to be almost tight by Nelson and Nguyễn [99], who showed that $\Omega(\varepsilon^{-1}\log(f/\delta)/\log(1/\varepsilon))$ column sparsity is required.

In short, the above line of work shows that it is possible to apply a $\text{JLT}(\varepsilon, \delta, f)$ matrix $\mathbf{S}$ to a vector $\mathbf{x}$ in $O(\text{nnz}(\mathbf{x}) \cdot \varepsilon^{-1}\log(f/\delta))$ time, where $\text{nnz}(\mathbf{x})$ denotes the number of non-zero entries of the vector $\mathbf{x}$. This results in a significant speedup over Theorem 4 when ε is small. It also leads to improvements in Theorem 6, though regarding ℓ_2-subspace embeddings, one can do better as discussed below.

A somewhat different line of work also came about in trying to speed up the basic construction in Theorem 4, and this is due to Ailon and Chazelle [2]. Instead of trying to achieve a sparse matrix $\mathbf{S}$, they tried to achieve an $\mathbf{S}$ which could be quickly applied to a vector $\mathbf{x}$. The underlying intuition here is that for a vector $\mathbf{x} \in \mathbb{R}^n$ whose ℓ_2 mass is spread roughly uniformly across its n coordinates, sampling a small number of its coordinates uniformly at random and rescaling results in a good estimate of the ℓ_2-norm of $\mathbf{x}$. However, if $\mathbf{x}$ does not have this property, e.g., it is sparse, then sampling is a very poor way to estimate the ℓ_2-norm of $\mathbf{x}$, as typically most samples will be 0. By the uncertainty principle, though, if $\mathbf{x}$ is sparse, then $\mathbf{Fx}$ cannot be too sparse, where $\mathbf{F}$ is the Fourier transform. This is also true for the Hadamard transform $\mathbf{Hx}$, and for any bounded orthonormal system

(i.e., an orthonormal matrix whose entry of maximum magnitude is bounded by $O(1/\sqrt{n})$). Indeed, from results in signal processing due to Donoho and Stark [36], if $\mathbf{A} = [\mathbf{I}_n \mathbf{B}]^T$ is a $2n \times n$ matrix such that $\mathbf{B}$ has orthonormal rows and columns, and for any distinct rows $\mathbf{B}_{i*}, \mathbf{B}_{j*}$ we have $|\mathbf{B}_{i*}, \mathbf{B}_{j*}| \leq M$, then for any $\mathbf{x} \in \mathbb{R}^n$, it holds that $\|\mathbf{x}\|_0 + \|\mathbf{B}\mathbf{x}\|_0 \geq 1/M$. See, e.g., [64], for algorithmic applications of this uncertainty principle.

Unfortunately $\mathbf{H}\mathbf{x}$ can still be sparse enough that a small number of samples will not work, so the intuition is to re-randomize $\mathbf{H}\mathbf{x}$ by applying a cheap rotation - namely, computing $\mathbf{H}\mathbf{D}\mathbf{x}$ for a diagonal matrix $\mathbf{D}$ with i.i.d. entries $\mathbf{D}_{i,i}$ in which $\mathbf{D}_{i,i} = 1$ with probability $1/2$, and $\mathbf{D}_{i,i} = -1$ with probability $1/2$. If $\mathbf{P}$ is an $k \times n$ matrix which implements coordinate sampling, then $\mathbf{P} \cdot \mathbf{H} \cdot \mathbf{D}\mathbf{x}$ now provides the desired Johnson-Lindenstrauss transform. Since $\mathbf{D}$ is a diagonal matrix, $\mathbf{D}\mathbf{x}$ can be computed in $O(n)$ time. The Hadamard matrix $\mathbf{H}$ can be applied to an n-dimensional vector in $O(n \log n)$ time. Finally, $\mathbf{P}$ can be applied to an n-dimensional vector in $O(k)$ time. Hence, $\mathbf{P} \cdot \mathbf{H} \cdot \mathbf{D}$ can be applied to a vector in $O(n \log n)$ time and to an $n \times d$ matrix in $O(nd \log n)$ time. We call this the *Fast Johnson Lindenstrauss Transform*. We note that this is not quite the same as the construction given by Ailon and Chazelle in [2], who form $\mathbf{P}$ slightly differently to obtain a better dependence on $1/\varepsilon$ in the final dimension.

The Fast Johnson Lindenstrauss Transform is significantly faster than the above $O(\text{nnz}(\mathbf{x}) \cdot \varepsilon^{-1} \log(f/\delta))$ time for many reasonable settings of the parameters, e.g., in a number of numerical linear algebra applications in which $1/\delta$ can be exponentially large in d. Indeed, the Fast Johnson Lindenstrauss Transform was first used by Sárlos to obtain the first speedups for regression and low rank matrix approximation with relative error. Sárlos used a version of the Fast Johnson Lindenstrauss Transform due to [2]. We will use a slightly different version called the *Subsampled Randomized Hadamard Transform*, or SRHT for short. Later we will see a significantly faster transform for sparse matrices.

Theorem 7 *(Subsampled Randomized Hadamard Transform [2, 105, 42, 43, 116, 44, 125]) Let* $\mathbf{S} = \frac{1}{\sqrt{kn}} \mathbf{P}\mathbf{H}_n\mathbf{D}$*, where* $\mathbf{D}$ *is an* $n \times n$ *diagonal matrix with i.i.d. diagonal entries* $\mathbf{D}_{i,i}$ *in which* $\mathbf{D}_{i,i} = 1$ *with probability* $1/2$*, and* $\mathbf{D}_{i,i} = -1$ *with probability* $1/2$*.* $\mathbf{H}_n$ *refers to the Hadamard matrix of size* n*, which we assume is a power of* 2*. Here, the* (i,j)*-th entry of* $\mathbf{H}_n$ *is given by* $(-1)^{\langle i,j \rangle}/\sqrt{n}$*, where* $\langle i,j \rangle = \sum_{z=1}^{\log n} i_z \cdot j_z$*, and where* $(i_{\log n}, \ldots, i_1)$ *and* $(j_{\log n}, \ldots, j_1)$ *are the binary representations of* i *and* j *respectively. The* $r \times n$ *matrix* $\mathbf{P}$ *samples* r *coordinates of an* n*-dimensional vector uniformly*

at random, where

$$r = \Omega(\varepsilon^{-2}(\log d)(\sqrt{d} + \sqrt{\log n})^2).$$

Then with probability at least .99, for any fixed $n \times d$ matrix $\mathbf{U}$ with orthonormal columns,

$$\|\mathbf{I}_d - \mathbf{U}^T\mathbf{\Pi}^T\mathbf{\Pi}\mathbf{U}\|_2 \leq \varepsilon.$$

Further, for any vector $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{Sx}$ can be computed in $O(n \log r)$ time.

We will not present the proof of Theorem 7, instead relying upon the above intuition. The proof of Theorem 7 can be found in the references listed above.

Using Theorem 7, it is possible to compute an oblivious ℓ_2-subspace embedding of a matrix $\mathbf{A}$ in $O(nd \log(d(\log n)/\varepsilon))$ time (see Definition 2.2 and Theorem 2.1 of [3] for details on obtaining this time complexity, which is a slight improvement to the $O(nd \log n)$ time mentioned above), which up to the logarithmic factor, is optimal in the matrix dimensions of $\mathbf{A} \in \mathbb{R}^{n \times d}$. One could therefore ask if this is the end of the road for subspace embeddings. Note that applying Theorem 6 to create an oblivious ℓ_2-subspace embedding $\mathbf{S}$, or also using its optimizations discussed in the paragraphs following Theorem 6 due to Kane and Nelson [67], would require time at least $O(\text{nnz}(\mathbf{A})d\varepsilon^{-1})$, since the number of non-zero entries per column of $\mathbf{S}$ would be $\Theta(\varepsilon^{-1}\log(f)) = \Theta(\varepsilon^{-1}d)$, since the f of Theorem 4 would need to be set to equal $\exp(d)$ to apply a net argument.

It turns out that many matrices $\mathbf{A} \in \mathbb{R}^{n \times d}$ are sparse, that is, the number of non-zero entries, $\text{nnz}(\mathbf{A})$, may be much smaller than $n \cdot d$. One could therefore hope to obtain an oblivious ℓ_2-subspace embedding $\mathbf{S}$ in which $\mathbf{S} \cdot \mathbf{A}$ can be computed in $O(\text{nnz}(\mathbf{A}))$ time and which the number of rows of $\mathbf{S}$ is small.

At first glance this may seem unlikely, since as mentioned above, any Johnson Lindenstrauss Transform requires $\Omega(\varepsilon^{-1}\log(f/\delta)/\log(1/\varepsilon))$ non-zero entries per column. Moreover, the size of any C-net for constant C is at least $2^{\Omega(d)}$, and therefore applying the arguments above we see that the "f" in the lower bound needs to be $\Omega(d)$. Alternatively, we could try to use an SRHT-based approach, but it is unknown how to adapt such approaches to exploit the sparsity of the matrix $\mathbf{A}$.

Nevertheless, in work with Clarkson [27] we show that it is indeed possible to achieve $O(\text{nnz}(\mathbf{A}))$ time to compute $\mathbf{S} \cdot \mathbf{A}$ for an oblivious $(1 \pm \varepsilon)$ ℓ_2 subspace embedding $\mathbf{S}$ with only an $r = \text{poly}(d/\varepsilon)$ number of rows. The key to bypassing the lower bound mentioned above is that $\mathbf{S}$ will *not* be a Johnson Lindenstrauss Transform; instead it will only work for a set of $f = 2^{\Omega(d)}$

specially chosen points rather than an arbitrary set of f points. It turns out if we choose $2^{\Omega(d)}$ points from a d-dimensional subspace, then the above lower bound of $\Omega(\varepsilon^{-1}\log(f/\delta)/\log(1/\varepsilon))$ non-zero entries per column does not apply; that is, this set of f points is far from realizing the worst-case for the lower bound.

In fact $\mathbf{S}$ is nothing other than the CountSkech matrix from the data stream literature [24, 115]. Namely, $\mathbf{S}$ is constructed via the following procedure: for each of the n columns $\mathbf{S}_{*i}$, we first independently choose a uniformly random row $h(i) \in \{1, 2, \ldots, r\}$. Then, we choose a uniformly random element of $\{-1, 1\}$, denoted $\sigma(i)$. We set $\mathbf{S}_{h(i),i} = \sigma(i)$ and set $\mathbf{S}_{j,i} = 0$ for all $j \neq i$. Thus, $\mathbf{S}$ has only a single non-zero entry per column. For example, suppose $\mathbf{S} \in \mathbb{R}^{4\times 5}$. Then an instance of $\mathbf{S}$ could be:

$$\begin{pmatrix} 0 & 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 \end{pmatrix}$$

We refer to such an $\mathbf{S}$ as a *sparse embedding matrix*. Note that since $\mathbf{S}$ has only a single non-zero entry per column, one can compute $\mathbf{S}\cdot\mathbf{A}$ for a matrix A in $O(\text{nnz}(\mathbf{A}))$ time.

Theorem 8 *([27]) For $\mathbf{S}$ a sparse embedding with $r = O(d^2/\varepsilon^2\text{poly}(\log(d/\varepsilon)))$ rows, for any fixed $n \times d$ matrix $\mathbf{A}$, with probability .99, $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$. Further, $\mathbf{S}\cdot\mathbf{A}$ can be computed in $O(\text{nnz}(\mathbf{A}))$ time.*

Although the number of rows of $\mathbf{S}$ is larger than the d/ε^2 using Theorem 6, typically $n \gg d$, e.g., in overconstrained regression problems, and so one can reduce $\mathbf{S}\cdot\mathbf{A}$ to a matrix containing $O(d/\varepsilon^2)$ rows by composing it with a matrix $\mathbf{S}'$ sampled using Theorem 4 or Theorem 7, computing $\mathbf{S}'\mathbf{S}\mathbf{A}$ in time $O(\text{nnz}(A))+\text{poly}(d/\varepsilon)$, and so provided $\text{poly}(d/\varepsilon) < \text{nnz}(\mathbf{A})$, this gives an overall $O(\text{nnz}(\mathbf{A}))$ time algorithm for obtaining an oblivious $(1 \pm \varepsilon)$ ℓ_2-subspace embedding with the optimal $O(d/\varepsilon^2)$ number of rows. Note here we can assume that $\text{nnz}(A) \geq n$, as otherwise we can delete the rows of all zeros in $\mathbf{A}$.

The key intuition behind Theorem 8, given in [27] why a sparse embedding matrix provides a subspace embedding, is that $\mathbf{S}$ need not preserve the norms of an arbitrary subset of $2^{O(d)}$ vectors in $\mathbb{R}^n$, but rather it need only preserve those norms of a subset of $2^{O(d)}$ vectors in $\mathbb{R}^n$ which *all sit in a d-dimensional subspace* of $\mathbb{R}^n$. Such a subset of $2^{O(d)}$ vectors is significantly

different from an arbitrary such set; indeed, the property used in [27] which invented this was the following. If $\mathbf{U} \in \mathbb{R}^{n \times d}$ is a matrix with orthonormal columns with the same column space as $\mathbf{A}$, then as one ranges over all unit $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{Ux}$ ranges over all unit vectors in the column space of $\mathbf{A}$. Note though that for any coordinate i, by the Cauchy-Schwarz inequality,

$$(\mathbf{Ux})_i^2 \leq \|\mathbf{U}_{i*}\|_2^2. \tag{4}$$

As $\sum_{i=1}^n \|\mathbf{U}_{i*}\|_2^2 = d$, since $\mathbf{U}$ has orthonormal columns, there is a subset T of $[n]$ of size at most d^2 for which if $(\mathbf{Ux})_i^2 \geq 1/d$, then $i \in T$. Notice that T does not depend on $\mathbf{x}$, but rather is just equal to those rows $\mathbf{U}_{i*}$ for which $\|\mathbf{U}_{i*}\|_2^2 \geq 1/d$. Hence, (4) implies that as one ranges over all unit vectors $\mathbf{Ux}$, the coordinates of $\mathbf{Ux}$ that are larger than $1/d$, if any, must lie in this relatively small set T. This is in sharp contrast to an arbitrary set of $2^{O(d)}$ unit vectors, for which every coordinate could be larger than $1/d$ for at least one vector in the collection. It turns out that if $\mathbf{Ux}$ has no heavy coordinates, then a sparse subspace embedding does have the Johnson-Lindenstrauss property, as shown by Dasgupta, Kumar, and Sárlos [33]. Hence, provided the set of coordinates of T is perfectly hashed by $\mathbf{S}$, one can handle the remaining coordinates by the analysis of [33].

While the above proof technique has proven useful in generating ℓ_p subspace embeddings for other ℓ_p-norms (as we will see in Chapter 3 for the ℓ_1-norm), and also applies more generally to sets of $2^{O(d)}$ vectors with a fixed small number of heavy coordinates, it turns out for ℓ_2 one can simplify and sharpen the argument by using more direct linear-algebraic methods. In particular, via a simpler second moment calculation, Theorem 8 was improved in [92, 97] to the following.

Theorem 9 *[92, 97] For any $0 < \delta < 1$, and for $\mathbf{S}$ a sparse embedding matrix with $r = O(d^2/(\delta\varepsilon^2))$ rows, then with probability $1 - \delta$, for any fixed $n \times d$ matrix $\mathbf{A}$, $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$. The matrix product $\mathbf{S} \cdot \mathbf{A}$ can be computed in $O(\text{nnz}(\mathbf{A}))$ time. Further, all of this holds if the hash function h defining $\mathbf{S}$ is only pairwise independent, and the sign function σ defining $\mathbf{S}$ is only 4-wise independent.*

The proofs of Theorem 9 given in [92, 97] work by bounding, for even integers $\ell \geq 2$,

$$\begin{aligned} \Pr[\|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU}\|_2 \geq \varepsilon] &= \Pr[\|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU}\|_2^\ell \geq \varepsilon^\ell] \\ &\leq \varepsilon^{-\ell}\mathbf{E}[\|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU}\|_2^\ell] \\ &\leq \varepsilon^{-\ell}\mathbf{E}[\mathtt{tr}((\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{SU})^\ell)], \end{aligned}$$

which is a standard way of bounding operator norms of random matrices, see, e.g., [12]. In the bound above, Markov's inequality is used in the first inequality, while the second inequality uses that the eigenvalues of $(\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U})^\ell$ are non-negative for even integers ℓ, one of those eigenvalues is $\|\mathbf{I}_d - \mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}\|_2^\ell$, and the trace is the sum of the eigenvalues. This is also the technique used in the proof of Theorem 10 below (we do not present the proof of this), though there a larger value of ℓ is used while for Theorem 9 we will see that it suffices to consider $\ell = 2$.

Rather than proving Theorem 9 directly, we will give a alternative proof of it observed by Nguyễn [100] in the next section, showing how it is a consequence of a primitive called approximate matrix multiplication that had been previously studied, and for which is useful for other applications we consider. Before doing so, though, we mention that it is possible to achieve fewer than $O(d^2/\varepsilon^2)$ rows for constant probability subspace embeddings if one is willing to increase the running time of applying the subspace embedding from $O(\text{nnz}(\mathbf{A}))$ to $O(\text{nnz}(\mathbf{A})/\varepsilon)$. This was shown by Nelson and Nguyễn [97]. They show that for any $\gamma > 0$, one can achieve $d^{1+\gamma}/\varepsilon^2\text{poly}(1/\gamma)$ dimensions by increasing the number of non-zero entries in $\mathbf{S}$ to $\text{poly}(1/\gamma)/\varepsilon$. They also show that by increasing the number of non-zero entries in $\mathbf{S}$ to $\text{polylog}(d)/\varepsilon$, one can achieve $d/\varepsilon^2\text{polylog}(d)$ dimensions. These results also generalize to failure probability δ, and are summarized by the following theorem.

Theorem 10 *[97] There are distributions on matrices S with the following properties:*

(1) For any fixed $\gamma > 0$ and any fixed $n \times d$ matrix $\mathbf{A}$, $\mathbf{S}$ is a $(1 \pm \varepsilon)$ oblivious ℓ_2-subspace embedding for $\mathbf{A}$ with $r = d^{1+\gamma}/\varepsilon^2$ rows and error probability $1/\text{poly}(d)$. Further, $\mathbf{S} \cdot \mathbf{A}$ can be computed in $O(\text{nnz}(\mathbf{A})\text{poly}(1/\gamma)/\varepsilon)$ time.

(2) There is a $(1 \pm \varepsilon)$ oblivious ℓ_2-subspace embedding for $\mathbf{A}$ with $r = d \cdot \text{polylog}(d/(\varepsilon\delta))/\varepsilon^2$ rows and error probability δ. Further, $\mathbf{S} \cdot \mathbf{A}$ can be computed in $O(\text{nnz}(\mathbf{A})\text{polylog}(d/(\varepsilon\delta)))/\varepsilon)$ time.

We note that for certain applications, such as least squares regression, one can still achieve a $(1+\varepsilon)$-approximation in $O(\text{nnz}(\mathbf{A}))$ time by applying Theorem 10 with the value of ε in Theorem 10 set to a fixed constant since the application only requires a $(1 \pm O(1))$-subspace embedding in order to achieve a $(1+\varepsilon)$-approximation; see Theorem 23 for further details on this. It is also conjectured in [97] that r can be as small as $O((d + \log(1/\delta))/\varepsilon^2)$ with a time for computing $\mathbf{S} \cdot \mathbf{A}$ of $O(\text{nnz}(\mathbf{A})\log(d/\delta)/\varepsilon)$, though at the time of this writing the polylogarithmic factors in Theorem 10 are somewhat far from achieving this conjecture.

There has been further work on this by Bourgain and Nelson [18], who showed among other things that if the columns of $\mathbf{U}$ form an orthonormal basis for the column space of $\mathbf{A}$, and if the coherence $\max_{i\in[n]} \|\mathbf{U}_{i*}\|_2^2 \leq 1/\text{polylog}(d)$, then a sparse embedding matrix provides a $(1\pm\varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$. Here the column sparsity remains 1 given the incoherence assumption, just as in Theorem 9. The authors also provide results for unions of subspaces.

We note that one can also achieve $1-\delta$ success probability bounds in which the sparsity and dimension depend on $O(\log 1/\delta)$ using these constructions [27, 92, 97]. For our applications it will usually not be necessary, as one can often instead repeat the entire procedure $O(\log 1/\delta)$ times and take the best solution found, such as in regression or low rank matrix approximation. We also state a different way of finding an ℓ_2-subspace embedding with high success probability in §2.3.

2.2 Matrix multiplication

In this section we study the approximate matrix product problem.

Definition 11 *Let $0 < \varepsilon < 1$ be a given approximation parameter. In the* Matrix Product *Problem matrices $\mathbf{A}$ and $\mathbf{B}$ are given, where $\mathbf{A}$ and $\mathbf{B}$ each have n rows and a total of c columns. The goal is to output a matrix $\mathbf{C}$ so that*

$$\|\mathbf{A}^T\mathbf{B} - \mathbf{C}\|_F \leq \varepsilon \|\mathbf{A}\|_F \|\mathbf{B}\|_F.$$

There are other versions of approximate matrix product, such as those that replace the Frobenius norms above with operator norms [83, 82, 29, 30]. Some of these works look at bounds in terms of the so-called stable rank of $\mathbf{A}$ and $\mathbf{B}$, which provides a continuous relaxation of the rank. For our application we will focus on the version of the problem given in Definition 11.

The idea for solving this problem is to compute $\mathbf{A}^T\mathbf{S}^T$ and $\mathbf{SB}$ for a sketching matrix $\mathbf{S}$. We will choose $\mathbf{S}$ so that

$$\mathbf{E}[\mathbf{A}^T\mathbf{S}^T\mathbf{SB}] = \mathbf{A}^T\mathbf{B},$$

and we could hope that the variance of this estimator is small, namely, we could hope that the standard deviation of the estimator is $O(\varepsilon\|\mathbf{A}\|_F\|\mathbf{B}\|_F)$. To figure out which matrices $\mathbf{S}$ are appropriate for this, we use the following theorem of Kane and Nelson [67]. This is a more general result of the

analogous result for sign matrices of Clarkson and the author [28], and a slight strengthening of a result of Sarlós [105].

Before giving the theorem, we need a definition.

Definition 12 *[67] A distribution $\mathcal{D}$ on matrices $\mathbf{S} \in \mathbb{R}^{k \times d}$ has the $(\varepsilon, \delta, \ell)$-JL moment property if for all $\mathbf{x} \in \mathbb{R}^d$ with $\|\mathbf{x}\|_2 = 1$,*

$$\mathbf{E}_{\mathbf{S} \sim \mathcal{D}} |\|\mathbf{S}\mathbf{x}\|_2^2 - 1|^\ell \leq \varepsilon^\ell \cdot \delta.$$

We prove the following theorem for a general value of ℓ, since as mentioned it is used in some subspace embedding proofs including the ones of Theorem 10. However, in this section we will only need the case in which $\ell = 2$.

Theorem 13 *[67] For $\varepsilon, \delta \in (0, 1/2)$, let $\mathcal{D}$ be a distribution over matrices with d columns that satisfies the $(\varepsilon, \delta, \ell)$-JL moment property for some $\ell \geq 2$. Then for $\mathbf{A}, \mathbf{B}$ matrices each with d rows,*

$$\Pr_{\mathbf{S} \sim \mathcal{D}} \left[\|\mathbf{A}^T \mathbf{S}^T \mathbf{S} \mathbf{B} - \mathbf{A}^T \mathbf{B}\|_{\mathrm{F}} > 3\varepsilon \|\mathbf{A}\|_{\mathrm{F}} \|\mathbf{B}\|_{\mathrm{F}} \right] \leq \delta.$$

Proof: We proceed as in the proof of [67]. For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we have

$$\frac{\langle \mathbf{S}\mathbf{x}, \mathbf{S}\mathbf{y} \rangle}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} = \frac{\|\mathbf{S}\mathbf{x}\|_2^2 + \|\mathbf{S}\mathbf{y}\|_2^2 - \|\mathbf{S}(\mathbf{x} - \mathbf{y})\|_2^2}{2}.$$

For a random scalar X, let $\|X\|_p = (\mathbf{E}|X|^p)^{1/p}$. We will sometimes consider $X = \|\mathbf{T}\|_F$ for a random matrix $\mathbf{T}$, in which case X is a random scalar and the somewhat cumbersome notation $\| \|\mathbf{T}\|_F \|_p$ indicates $(\mathbf{E}[\|\mathbf{T}\|_F^p])^{1/p}$.

Minkowski's inequality asserts that the triangle inequality holds for this definition, namely, that $\|\mathbf{X} + \mathbf{Y}\|_p \leq \|\mathbf{X}\|_p + \|\mathbf{Y}\|_p$, and as the other properties of a norm are easy to verify, it follows that $\|.\|_p$ is a norm. Using that it is a norm, we have for unit vectors $\mathbf{x}$, $\mathbf{y}$, that $\|\langle \mathbf{S}\mathbf{x}, \mathbf{S}\mathbf{y} \rangle - \langle \mathbf{x}, \mathbf{y} \rangle\|_\ell$ is equal to

$$\begin{aligned}
&= \frac{1}{2} \cdot \|(\|\mathbf{S}\mathbf{x}\|_2^2 - 1) + (\|\mathbf{S}\mathbf{y}\|_2^2 - 1) - (\|\mathbf{S}(\mathbf{x} - \mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2)\|_\ell \\
&\leq \frac{1}{2} \cdot \left(\| \|\mathbf{S}\mathbf{x}\|_2^2 - 1\|_\ell + \| \|\mathbf{S}\mathbf{y}\|_2^2 - 1\|_\ell + \| \|\mathbf{S}(\mathbf{x} - \mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2 \|_\ell \right) \\
&\leq \frac{1}{2} \cdot \left(\varepsilon \cdot \delta^{1/\ell} + \varepsilon \cdot \delta^{1/\ell} + \|\mathbf{x} - \mathbf{y}\|_2^2 \cdot \varepsilon \cdot \delta^{1/\ell} \right) \\
&\leq 3\varepsilon \cdot \delta^{1/\ell}.
\end{aligned}$$

By linearity, this implies for arbitrary vectors $\mathbf{x}$ and $\mathbf{y}$ that $\frac{\|\langle \mathbf{Sx},\mathbf{Sy}\rangle - \langle \mathbf{x},\mathbf{y}\rangle\|_\ell}{\|\mathbf{x}|_2\|\mathbf{y}\|_2} \leq 3\varepsilon \cdot \delta^{1/\ell}$.

Suppose $\mathbf{A}$ has n columns and $\mathbf{B}$ has m columns. Let the columns of $\mathbf{A}$ be $\mathbf{A}_1, \ldots, \mathbf{A}_n$ and the columns of $\mathbf{B}$ be $\mathbf{B}_1, \ldots, \mathbf{B}_n$. Define the random variable

$$X_{i,j} = \frac{1}{\|A_i\|_2\|B_j\|_2} \cdot \left(\langle \mathbf{SA}_i, \mathbf{SB}_j\rangle - \langle \mathbf{A}_i, \mathbf{B}_j\rangle\right).$$

Then, $\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}}^2 = \sum_{i=1}^n \sum_{j=1}^m \|\mathbf{A}_i\|_2^2 \cdot \|\mathbf{B}_j\|_2^2 \cdot X_{i,j}^2$. Again using Minkowski's inequality and that $\ell/2 \geq 1$,

$$\begin{aligned}
\|\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}}^2\|_{\ell/2} &= \|\sum_{i=1}^n \sum_{j=1}^m \|\mathbf{A}_i\|_2^2 \cdot \|\mathbf{B}_j\|_2^2 \cdot X_{i,j}^2\|_{\ell/2} \\
&\leq \sum_{i=1}^n \sum_{j=1}^m \|\mathbf{A}_i\|_2^2 \cdot \|\mathbf{B}_j\|_2^2 \cdot \|X_{i,j}^2\|_{\ell/2} \\
&= \sum_{i=1}^n \sum_{j=1}^m \|\mathbf{A}_i\|_2^2 \cdot \|\mathbf{B}_j\|_2^2 \cdot \|X_{i,j}\|_\ell^2 \\
&\leq (3\varepsilon\delta^{1/\ell})^2 \cdot \left(\sum_{i=1}^n \sum_{j=1}^m \|\mathbf{A}_i\|_2^2 \cdot \|\mathbf{B}_j\|_2^2\right) \\
&= (3\varepsilon\delta^{1/\ell})^2 \cdot \|\mathbf{A}\|_{\mathrm{F}}^2\|\mathbf{B}\|_{\mathrm{F}}^2.
\end{aligned}$$

Using that $\mathbf{E}\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}}^\ell = \|\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|\|_{\mathrm{F}}^2\|_{\ell/2}^{\ell/2}$, together with Markov's inequality, we have

$$\begin{aligned}
\Pr\left[\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}} > 3\varepsilon\|\mathbf{A}\|_{\mathrm{F}}\|\mathbf{B}\|_{\mathrm{F}}\right] &\leq \left(\frac{1}{3\varepsilon\|\mathbf{A}\|_{\mathrm{F}}\|\mathbf{B}\|_{\mathrm{F}}}\right)^\ell \\
&\cdot \quad \mathbf{E}\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}}^\ell \\
&\leq \delta.
\end{aligned}$$

■

We now show that sparse embeddings matrices satisfy the $(\varepsilon, \delta, 2)$-JL-moment property. This was originally shown by Thorup and Zhang [115].

Theorem 14 *Let $\mathbf{S}$ be a sparse embedding matrix, as defined in §2.1, with at least $2/(\varepsilon^2\delta)$ rows. Then $\mathbf{S}$ satisfies the $(\varepsilon, \delta, 2)$-JL moment property. Further, this holds if the hash function h defining the sparse embedding matrix is only 2-wise independent and the sign function σ is 4-wise independent.*

Proof: As per Definition 12, we need to show for any unit vector $\mathbf{x} \in \mathbb{R}^d$,

$$\mathbf{E}_{\mathbf{S}}[(\|\mathbf{S}\mathbf{x}\|_2^2 - 1)^2] = \mathbf{E}_{\mathbf{S}}[\|\mathbf{S}\mathbf{x}\|_2^4] - 2\mathbf{E}_{\mathbf{S}}[\|\mathbf{S}\mathbf{x}\|_2^2] + 1 \leq \varepsilon^2\delta. \tag{5}$$

For a sparse embedding matrix $\mathbf{S}$, we let $h : [d] \to [r]$ be a random 2-wise independent hash function indicating for each column $j \in [d]$, which row in $\mathbf{S}$ contains the non-zero entry. Further, we let $\sigma : [d] \to \{-1, 1\}$ be a 4-wise independent function, independent of h, indicating whether the non-zero entry in the j-th column is 1 or -1. For an event $\mathcal{E}$, let $\delta(\mathcal{E})$ be an indicator variable which is 1 if $\mathcal{E}$ occurs, and is 0 otherwise. Then,

$$\begin{aligned}
\mathbf{E}[\|\mathbf{S}\mathbf{x}\|_2^2] &= \sum_{i\in[r]} \mathbf{E}\left[\left(\sum_{j\in[d]} \delta(h(j) = i)\mathbf{x}_j\sigma(j)\right)^2\right] \\
&= \sum_{i\in[r]} \sum_{j,j'\in[d]} \mathbf{x}_j\mathbf{x}_{j'}\mathbf{E}\left[\delta(h(j) = i)\delta(h(j') = i)\right]\mathbf{E}\left[\sigma(j)\sigma(j')\right] \\
&= \sum_{i\in[r]} \sum_{j\in[d]} \frac{\mathbf{x}_j^2}{r} \\
&= \|\mathbf{x}\|_2^2 \\
&= 1,
\end{aligned}$$

where the second equality uses that h and σ are independent, while the third equality uses that $\mathbf{E}[\sigma(j)\sigma(j')] = 1$ if $j = j'$, and otherwise is equal to 0.

We also have,

$$\begin{aligned}
\mathbf{E}[\|\mathbf{S}\mathbf{x}\|_2^4] &= \mathbf{E}\left[\left(\sum_{i\in[r]}\left(\sum_{j\in[d]}\delta(h(j)=i)\mathbf{x}_j\sigma(j)\right)^2\right)^2\right] \\
&= \sum_{i,i'\in[r]}\sum_{j_1,j_2,j_1',j_2'\in[d]}\mathbf{x}_{j_1}\mathbf{x}_{j_2}\mathbf{x}_{j_1'}\mathbf{x}_{j_2'} \\
&\quad \cdot \mathbf{E}\left[\delta(h(j_1)=i)\delta(h(j_2)=i)\delta(h(j_1')=i')\delta(h(j_2')=i')\right] \\
&\quad \cdot \mathbf{E}\left[\sigma(j_1)\sigma(j_2)\sigma(j_1')\sigma(j_2')\right] \\
&= \sum_{i\in[r]}\sum_{j\in[d]}\frac{\mathbf{x}_j^4}{r} + \sum_{i,i'\in[r]}\sum_{j_1\neq j_1'\in[d]}\frac{\mathbf{x}_{j_1}^2\mathbf{x}_{j_1'}^2}{r^2} \\
&\quad +2\sum_{i\in[r]}\sum_{j_1\neq j_2\in[d]}\frac{\mathbf{x}_{j_1}^2\mathbf{x}_{j_1'}^2}{r^2} \\
&= \sum_{j,j'\in[d]}\mathbf{x}_j^2\mathbf{x}_{j'}^2 + \frac{2}{r}\sum_{j_1\neq j_2\in[d]}\mathbf{x}_{j_1}^2\mathbf{x}_{j_1'}^2 \\
&\leq \|\mathbf{x}\|_2^4 + \frac{2}{r}\|\mathbf{x}\|_2^4 \\
&\leq 1+\frac{2}{r},
\end{aligned}$$

where the second equality uses the independence of h and σ, and the third equality uses that since σ is 4-wise independent, in order for $\mathbf{E}\left[\sigma(j_1)\sigma(j_2)\sigma(j_1')\sigma(j_2')\right]$ not to vanish, it must be that either

1. $j_1 = j_2 = j_1' = j_2'$ or
2. $j_1 = j_2$ and $j_1' = j_2'$ but $j_1 \neq j_1'$ or
3. $j_1 = j_1'$ and $j_2 = j_2'$ but $j_1 \neq j_2$ or
4. $j_1 = j_2'$ and $j_1' = j_2$ but $j_1 \neq j_2$.

Note that in the last two cases, for $\mathbf{E}\left[\delta(h(j_1)=i)\delta(h(j_2)=i)\delta(h(j_1')=i')\delta(h(j_2')=i')\right]$ not to vanish, we must have $i = i'$. The fourth equality and first inequality are based on regrouping the summations, and the sixth inequality uses that $\|\mathbf{x}\|_2 = 1$.

Plugging our bounds on $\|\mathbf{S}\mathbf{x}\|_2^4$ and $\|\mathbf{S}\mathbf{x}\|_2^2$ into (5), the theorem follows.

■

We now present a proof that sparse embedding matrices provide subspace embeddings, as mentioned in §2.1, as given by Nguyễn [100].

Proof of Theorem 9: By Theorem 14, we have that $\mathbf{S}$ satisfies the $(\varepsilon, \delta, 2)$-JL moment property. We can thus apply Theorem 13.

To prove Theorem 9, recall that if $\mathbf{U}$ is an orthonormal basis for the column space of $\mathbf{A}$ and $\|\mathbf{S}\mathbf{y}\|_2 = (1 \pm \varepsilon)\|\mathbf{y}\|_2$ for all $\mathbf{y}$ in the column space of $\mathbf{U}$, then $\|\mathbf{S}\mathbf{y}\|_2 = (1 \pm \varepsilon)\|\mathbf{y}\|_2$ for all $\mathbf{y}$ in the column space of $\mathbf{A}$, since the column spaces of $\mathbf{A}$ and $\mathbf{U}$ are the same.

We apply Theorem 13 to $\mathbf{S}$ with the $\mathbf{A}$ and $\mathbf{B}$ of that theorem equal to $\mathbf{U}$, and the ε of that theorem equal to ε/d. Since $\mathbf{U}^T\mathbf{U} = \mathbf{I}_d$ and $\|\mathbf{U}\|_{\mathrm{F}}^2 = d$, we have,

$$\Pr_{\mathbf{S}\sim\mathcal{D}}\left[\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d\|_{\mathrm{F}} > 3\varepsilon\right] \leq \delta,$$

which implies that

$$\Pr_{\mathbf{S}\sim\mathcal{D}}\left[\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d\|_2 > 3\varepsilon\right] \leq \Pr_{\mathbf{S}\sim\mathcal{D}}\left[\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d\|_{\mathrm{F}} > 3\varepsilon\right] \leq \delta.$$

Recall that the statement that $\mathbf{x}^T(\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d)\mathbf{x} \leq 3\varepsilon$ for all unit $\mathbf{x} \in \mathbb{R}^d$ is equivalent to the statement that $\|\mathbf{S}\mathbf{U}\mathbf{x}\|_2^2 = 1 \pm 3\varepsilon$ for all unit $\mathbf{x} \in \mathbb{R}^d$, that is, $\mathbf{S}$ is a $(1 \pm 3\varepsilon)$ ℓ_2-subspace embedding. The proof follows by rescaling ε by 3. ■

2.3 High probability

The dependence of Theorem 9 on the error probability δ is linear, which is not completely desirable. One can use Theorem 10 to achieve a logarithmic dependence, but then the running time would be at least $\mathrm{nnz}(\mathbf{A})\mathrm{polylog}(d/(\varepsilon\delta))/\varepsilon$ and the number of non-zeros per column of $\mathbf{S}$ would be at least $\mathrm{polylog}(d/(\varepsilon\delta))/\varepsilon$. Here we describe an alternative way based on [13] which takes only $O(\mathrm{nnz}(\mathbf{A})\log(1/\delta))$ time, and preserves the number of non-zero entries per column of $\mathbf{S}$ to be 1. It is, however, a non-oblivious embedding.

In [13], an approach (Algorithm 1 below) to boost the success probability by computing $t = O(\log(1/\delta))$ independent sparse oblivious subspace embeddings $\mathbf{S}_j\mathbf{A}$ is proposed, $j = 1, 2, \ldots, t$, each with only constant success probability, and then running a cross validation procedure to find one which succeeds with probability $1 - 1/\delta$. More precisely, we compute the SVD of all embedded matrices $\mathbf{S}_j\mathbf{A} = \mathbf{U}_j\mathbf{D}_j\mathbf{V}_j^\top$, and find a $j \in [t]$ such that for at least half of the indices $j' \neq j$, all singular values of $\mathbf{D}_j\mathbf{V}_j^\top\mathbf{V}_{j'}\mathbf{D}_{j'}^\top$ are in $[1 \pm O(\varepsilon)]$.

Algorithm 1 Boosting success probability of embedding

Input: $\mathbf{A} \in \mathbb{R}^{n \times d}$, parameters ε, δ

1. Construct $t = O(\log \frac{1}{\delta})$ independent constant success probability sparse subspace embeddings $\mathbf{S}_j\mathbf{A}$ with accuracy $\varepsilon/6$.
2. Compute SVD $\mathbf{S}_j\mathbf{A} = \mathbf{U}_j\mathbf{D}_j\mathbf{V}_j^\top$ for $j \in [t]$.
3. For $j \in [t]$
   (a) Check if for at least half $j' \neq j$,
   $$\sigma_i(\mathbf{D}_j\mathbf{V}_j^\top\mathbf{V}_{j'}\mathbf{D}_{j'}^\top) \in [1 \pm \varepsilon/2], \forall i.$$
   (b) If so, output $\mathbf{S}_j\mathbf{A}$.

The reason why such an embedding $\mathbf{S}_j\mathbf{A}$ succeeds with high probability is as follows. Any two successful embeddings $\mathbf{S}_j\mathbf{A}$ and $\mathbf{S}_{j'}\mathbf{A}$, by definition, satisfy that $\|\mathbf{S}_j\mathbf{A}\mathbf{x}\|_2^2 = (1 \pm O(\varepsilon))\|\mathbf{S}_{j'}\mathbf{A}\mathbf{x}\|_2^2$ for all x, which we show is equivalent to passing the test on the singular values. Since with probability at least $1 - \delta$, a $9/10$ fraction of the embeddings are successful, it follows that the one we choose is successful with probability $1 - \delta$. One can thus show the following theorem.

Theorem 15 *([13]) Algorithm 1 outputs a subspace embedding with probability at least $1 - \delta$. In expectation step 3 is only run a constant number of times.*

2.4 Leverage scores

We now introduce the concept of leverage scores, which provide alternative subspace embeddings based on sampling a small number of rows of $\mathbf{A}$. We will see that they play a crucial role in various applications in this book, e.g., CUR matrix decompositions and spectral sparsification. Here we use the parameter k instead of d for the dimension of the subspace, as this will match our use in applications. For an excellent survey on leverage scores, we refer the reader to [85].

Definition 16 *(Leverage Score Sampling) Let $\mathbf{Z} \in \mathbb{R}^{n \times k}$ have orthonormal columns, and let $p_i = \ell_i^2/k$, where $\ell_i^2 = \|\mathbf{e}_i^T\mathbf{Z}\|_2^2$ is the i-th leverage score of*

$\mathbf{Z}$. *Note that* $(p_1, \ldots, p_n)$ *is a distribution. Let* $\beta > 0$ *be a parameter, and suppose we have any distribution* $q = (q_1, \ldots, q_n)$ *for which for all* $i \in [n]$, $q_i \geq \beta p_i$.

Let s *be a parameter. Construct an* $n \times s$ *sampling matrix* $\mathbf{\Omega}$ *and an* $s \times s$ *rescaling matrix* $\mathbf{D}$ *as follows. Initially,* $\mathbf{\Omega} = \mathbf{0}^{n \times s}$ *and* $\mathbf{D} = \mathbf{0}^{s \times s}$. *For each column* j *of* $\mathbf{\Omega}, \mathbf{D}$, *independently, and with replacement, pick a row index* $i \in [n]$ *with probability* q_i, *and set* $\mathbf{\Omega}_{i,j} = 1$ *and* $\mathbf{D}_{jj} = 1/\sqrt{q_i s}$. *We denote this procedure* $\mathrm{R}and\mathit{Sampling}(\mathbf{Z}, s, q)$.

Note that the matrices $\mathbf{\Omega}$ and $\mathbf{D}$ in the RandSampling$(\mathbf{Z}, s, q)$ procedure can be computed in $O(nk + n + s \log s)$ time.

Definition 16 introduces the concept of the *leverage scores* $\ell_i^2 = \|\mathbf{e}_i^T \mathbf{Z}\|_2^2$ of a matrix $\mathbf{Z}$ with orthonormal columns. For an $n \times k$ matrix $\mathbf{A}$ whose columns need not be orthonormal, we can still define its leverage scores ℓ_i^2 as $\|e_i^T \mathbf{Z}\|_2^2$, where $\mathbf{Z}$ is an $n \times r$ matrix with orthonormal columns having the same column space of $\mathbf{A}$, where r is the rank of $\mathbf{A}$. Although there are many choices $\mathbf{Z}$ of orthonormal bases for the column space of $\mathbf{A}$, it turns out that they all give rise to the same values ℓ_i^2. Indeed, if $\mathbf{Z}'$ were another $n \times r$ matrix with orthonormal colums having the same column space of $\mathbf{A}$, then $\mathbf{Z}' = \mathbf{ZR}$ for an $r \times r$ invertible matrix $\mathbf{R}$. But since $\mathbf{Z}'$ and $\mathbf{Z}$ have orthonormal columns, $\mathbf{R}$ must be orthonormal. Indeed, for every vector $\mathbf{x}$ we have

$$\|\mathbf{x}\|_2 = \|\mathbf{Z}'\mathbf{x}\|_2 = \|\mathbf{ZRx}\|_2 = \|\mathbf{Rx}\|_2.$$

Hence

$$\|\mathbf{e}_i^T \mathbf{Z}'\|_2^2 = \|\mathbf{e}_i^T \mathbf{ZR}\|_2^2 = \|\mathbf{e}_i^t \mathbf{Z}\|_2^2,$$

so the definition of the leverage scores does not depend on a particular choice of orthonormal basis for the column space of $\mathbf{A}$.

Another useful property, though we shall not need it, is that the leverage scores ℓ_i^2 are at most 1. This follows from the fact that any row $\mathbf{v}$ of $\mathbf{Z}$ must have squared norm at most 1, as otherwise

$$\|\mathbf{Z} \cdot \frac{\mathbf{v}}{\|\mathbf{v}\|_2}\|_2^2 = \frac{1}{\|\mathbf{v}\|_2^2} \cdot \|\mathbf{Zv}\|_2^2 > 1,$$

contradicting that $\|\mathbf{Zx}\|_2 = \|\mathbf{x}\|_2$ for all $\mathbf{x}$ since $\mathbf{Z}$ has orthonormal columns.

The following shows that $\mathbf{D}^T \mathbf{\Omega}^T \mathbf{Z}$ is a subspace embedding of the column space of $\mathbf{Z}$, for s large enough. To the best of our knowledge, theorems of this form first appeared in [40, 39]. Here we give a simple proof along the lines in [81].

Theorem 17 *(see, e.g., similar proofs in [81]) Suppose* $\mathbf{Z} \in \mathbb{R}^{n\times k}$ *has orthonormal columns. Suppose* $s > 144k\ln(2k/\delta)/(\beta\varepsilon^2)$ *and* $\mathbf{\Omega}$ *and* $\mathbf{D}$ *are constructed from the* R*andSampling*$(\mathbf{Z}, s, q)$ *procedure. Then with probability at least* $1-\delta$*, simultaneously for all* i*,*

$$1-\varepsilon \leq \sigma_i^2(\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}) \leq 1+\varepsilon,$$

or equivalently,

$$1-\varepsilon \leq \sigma_i^2(\mathbf{Z}^T\mathbf{\Omega}\mathbf{D}) \leq 1+\varepsilon.$$

Proof: We will use the following matrix Chernoff bound for a sum of random matrices, which is a non-commutative Bernstein bound.

Fact 1 *(Matrix Chernoff) Let* $\mathbf{X}_1, \ldots, \mathbf{X}_s$ *be independent copies of a symmetric random matrix* $\mathbf{X} \in \mathbb{R}^{k\times k}$ *with* $\mathbf{E}[\mathbf{X}] = 0$*,* $\|\mathbf{X}\|_2 \leq \gamma$*, and* $\|\mathbf{E}\mathbf{X}^T\mathbf{X}\|_2 \leq s^2$*. Let* $\mathbf{W} = \frac{1}{s}\sum_{i=1}^s \mathbf{X}_i$*. Then for any* $\varepsilon > 0$*,*

$$\Pr[\|\mathbf{W}\|_2 > \varepsilon] \leq 2k\exp(-s\varepsilon^2/(2s^2 + 2\gamma\varepsilon/3)).$$

Let $\mathbf{U}_i \in \mathbb{R}^{1\times k}$ be the i-th sampled row of $\mathbf{Z}$ by the RandSampling$(\mathbf{Z}, s)$ procedure. Let $\mathbf{z}_j$ denote the j-th row of $\mathbf{Z}$. Let $\mathbf{X}_i = \mathbf{I}_k - \mathbf{U}_i^T\mathbf{U}_i/q_i$. Then the $\mathbf{X}_i$ are independent copies of a matrix random variable, and

$$\mathbf{E}[\mathbf{X}_i] = \mathbf{I}_k - \sum_{j=1}^n q_j\mathbf{z}_j^T\mathbf{z}_j/q_j = \mathbf{I}_k - \mathbf{Z}^T\mathbf{Z} = \mathbf{0}_{k\times k}.$$

For any j, $\mathbf{z}_j^T\mathbf{z}_j/q_j$ is a rank-1 matrix with operator norm bounded by $\|\mathbf{z}_j\|_2^2/q_j \leq k/\beta$. Hence,

$$\|\mathbf{X}_i\|_2 \leq \|\mathbf{I}_k\|_2 + \|\mathbf{U}_i^T\mathbf{U}_i/q_i\|_2 \leq 1 + \frac{k}{\beta}. \tag{6}$$

We also have

$$\begin{aligned}
\mathbf{E}[\mathbf{X}^T\mathbf{X}] &= \mathbf{I}_k - 2\mathbf{E}[\mathbf{U}_i^T\mathbf{U}_i/q_i] + \mathbf{E}[\mathbf{U}_i^T\mathbf{U}_i\mathbf{U}_i^T\mathbf{U}_i/q_i^2] \\
&= \sum_{j=1}^n \mathbf{z}_j^T\mathbf{z}_j\mathbf{z}_j^T\mathbf{z}_j/q_i - \mathbf{I}_k \qquad (7)\\
&\leq (k/\beta)\sum_{j=1}^n \mathbf{z}_j^T\mathbf{z}_j - \mathbf{I}_k \qquad (8)\\
&= (k/\beta - 1)\mathbf{I}_k. \qquad (9)
\end{aligned}$$

It follows that $\|\mathbf{E}\mathbf{X}^T\mathbf{X}\|_2 \leq (k/\beta - 1)$. Note that $\mathbf{W} = \frac{1}{k}\sum_{i=1}^{s}\mathbf{X}_i = \mathbf{I}_k - \mathbf{Z}^T\mathbf{\Omega}\mathbf{D}\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}$. Applying Fact 1,

$$\Pr[\|\mathbf{I}_k - \mathbf{Z}^T\mathbf{\Omega}\mathbf{D}\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}\|_2 > \varepsilon] \leq 2k\exp(-s\varepsilon^2/(2k/\beta + 2k\varepsilon/(3\beta))),$$

and setting $s = \Theta(k\log(k/\delta)/(\beta\varepsilon^2))$ implies that with all but δ probability, $\|\mathbf{I}_k - \mathbf{Z}^T\mathbf{\Omega}\mathbf{D}\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}\|_2 \leq \varepsilon$, that is, all of the singular values of $\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}$ are within $1 \pm \varepsilon$, as desired. ■

To apply Theorem 17 for computing subspace embeddings of an $n \times k$ matrix $\mathbf{A}$, one writes $\mathbf{A} = \mathbf{Z}\mathbf{\Sigma}\mathbf{V}^T$ in its SVD. Then, Theorem 17 guarantees that for all $\mathbf{x} \in \mathbb{R}^k$,

$$\|\mathbf{D}^T\mathbf{\Omega}^T\mathbf{A}\mathbf{x}\|_2 = (1 \pm \varepsilon)\|\mathbf{\Sigma}\mathbf{V}^T\|x\|_2 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{x}\|_2,$$

where the first equality uses the definition of $\mathbf{A}$ and the fact that all singular values of $\mathbf{D}^T\mathbf{\Omega}^T\mathbf{Z}$ are $1 \pm \varepsilon$. The second equality uses that $\mathbf{Z}$ has orthonormal columns, so $\|\mathbf{Z}\mathbf{y}\|_2 = \|\mathbf{y}\|_2$ for all vectors $\mathbf{y}$.

One drawback of RandSampling$(\mathbf{Z}, s, q)$ is it requires as input a distribution q which well-approximates the leverage score distribution p of $\mathbf{Z}$. While one could obtain p exactly by computing the SVD of $\mathbf{A}$, this would naïvely take $O(nk^2)$ time (assuming $k < n$). It turns out, as shown in [44], one can compute a distribution q with the approximation parameter $\beta = 1/2$ in time $O(nk\log n + k^3)$ time. This was further improved in [27] to $O(\text{nnz}(A)\log n + k^3)$ time.

We need a version of the Johnson-Lindenstrauss lemma, as follows. We give a simple proof for completeness.

Lemma 18 *(Johnson-Lindenstrauss) Given n points $q_1, \ldots, q_n \in \mathbb{R}^d$, if $\mathbf{G}$ is a $t \times d$ matrix of i.i.d. $N(0, 1/t)$ random variables, then for $t = O(\log n/\varepsilon^2)$ simultaneously for all $i \in [n]$,*

$$\Pr[\forall i,\ \|\mathbf{G}\mathbf{q}_i\|_2 \in (1 \pm \varepsilon)\|\mathbf{q}_i\|_2] \geq 1 - \frac{1}{n}.$$

Proof: For a fixed $i \in [n]$, $\mathbf{G}\mathbf{q}_i$ is a t-tuple of i.i.d. $N(0, \|\mathbf{q}_i\|_2^2/t)$ random variables. Here we use the fact that for independent standard normal random variables g and h and scalars a and b, the random variable $a \cdot g + b \cdot h$ has the same distribution as that of the random variable $\sqrt{a^2 + b^2}z$, where $z \sim N(0, 1)$.

It follows that $\|\mathbf{G}\mathbf{q}_i\|_2^2$ is equal, in distribution, to $(\|\mathbf{q}_i\|_2^2/t) \cdot \sum_{i=1}^{t} g_i^2$, where $g_1, \ldots, g_t$ are independent $N(0, 1)$ random variables.

The random variable $\sum_{i=1}^{t} g_i^2$ is χ^2 with t degree of freedom. The following tail bounds are known.

Fact 2 *(Lemma 1 of [74]) Let $g_1, \ldots, g_t$ be i.i.d. $N(0,1)$ random variables. Then for any $x \geq 0$,*

$$\Pr[\sum_{i=1}^{t} g_i^2 \geq t + 2\sqrt{tx} + 2x] \leq \exp(-x),$$

and

$$\Pr[\sum_{i=1}^{t} g_i^2 \leq t - 2\sqrt{tx}] \leq \exp(-x).$$

Setting $x = \varepsilon^2 t/16$, we have that

$$\Pr[|\sum_{i=1}^{t} g_i^2 - t| \leq \varepsilon t] \leq 2\exp(-\varepsilon^2 t/16).$$

For $t = O(\log n/\varepsilon^2)$, the lemma follows by a union bound over $i \in [n]$. ■

Theorem 19 *([27]) Fix any constant $\beta \in (0,1)$. If p is the leverage score distribution of an $n \times k$ matrix $\mathbf{Z}$ with orthonormal columns, it is possible to compute a distribution q on the n rows for which with probability $9/10$, simultaneously for all $i \in [n]$, $q_i \geq \beta p_i$. The time complexity is $O(\text{nnz}(A)\log n) + \text{poly}(k)$.*

Proof: Let $\mathbf{S}$ be a sparse embedding matrix with $r = O(k^2/\gamma^2)$ rows for a constant $\gamma \in (0,1)$ to be specified. We can compute $\mathbf{S} \cdot \mathbf{A}$ in $O(\text{nnz}(A))$ time. We then compute a QR-factorization of $\mathbf{SA} = \mathbf{Q} \cdot \mathbf{R}$, where $\mathbf{Q}$ has orthonormal columns. This takes $O(rk^2) = \text{poly}(k/\gamma)$ time. Note that $\mathbf{R}^{-1}$ is $k \times k$, and can be computed from $\mathbf{R}$ in $O(k^3)$ time (or faster using fast matrix multiplication).

For $t = O(\log n/\gamma^2)$, let G by a $k \times t$ matrix of i.i.d. $N(0, 1/t)$ random variables. Set $q_i = \|\mathbf{e}_i^T \mathbf{A}\mathbf{R}^{-1}\mathbf{G}\|_2^2$ for all $i \in [n]$. While we cannot compute $\mathbf{A} \cdot \mathbf{R}^{-1}$ very efficiently, we can first compute $\mathbf{R}^{-1}\mathbf{G}$ in $O(k^2 \log n/\gamma^2)$ by standard matrix multiplication, and then compute $\mathbf{A} \cdot (\mathbf{R}^{-1}\mathbf{G})$ in $O(\text{nnz}(A)\log n/\gamma^2)$ time since $(\mathbf{R}^{-1}\mathbf{G})$ has a small number of columns. Since we will set γ to be a constant, the overall time complexity of the theorem follows.

For correctness, by Lemma 18, with probability $1 - 1/n$, simultaneously for all $i \in [n]$, $q_i \geq (1-\gamma)\|\mathbf{e}_i^T \mathbf{A}\mathbf{R}^{-1}\|_2^2$, which we condition on. We now show that $\|\mathbf{e}_i^T \mathbf{A}\mathbf{R}^{-1}\|_2^2$ is approximately p_i. To do so, first consider $\mathbf{A}\mathbf{R}^{-1}$. The claim is that all of the singular values of $\mathbf{A}\mathbf{R}^{-1}$ are in the range $[1-\gamma, 1+\gamma]$.

To see this, note that for any $\mathbf{x} \in \mathbb{R}^k$,

$$\begin{aligned} \|\mathbf{A}\mathbf{R}^{-1}x\|_2^2 &= (1 \pm \gamma)\|\mathbf{S}\mathbf{A}\mathbf{R}^{-1}\mathbf{x}\|_2^2 \\ &= (1 \pm \gamma)\|\mathbf{Q}\mathbf{x}\|_2^2 \\ &= (1 \pm \gamma)\|\mathbf{x}\|_2^2, \end{aligned}$$

where the first equality follows since with probability 99/100, $\mathbf{S}$ is a $(1 \pm \gamma)$ ℓ_2-subspace embedding for $\mathbf{A}$, while the second equality uses the definition of $\mathbf{R}$, and the third equality uses that $\mathbf{Q}$ has orthonormal columns.

Next, if $\mathbf{U}$ is an orthonormal basis for the column space of $\mathbf{A}$, since $\mathbf{A}\mathbf{R}^{-1}$ and $\mathbf{U}$ have the same column space, $\mathbf{U} = \mathbf{A}\mathbf{R}^{-1}\mathbf{T}$ for a $k \times k$ change of basis matrix $\mathbf{T}$. The claim is that the minimum singular value of $\mathbf{T}$ is at least $1-2\gamma$. Indeed, since all of the singular values of $\mathbf{A}\mathbf{R}^{-1}$ are in the range $[1-\gamma, 1+\gamma]$, if there were a singular value of $\mathbf{T}$ smaller than $1-2\gamma$ with corresponding right singular vector $\mathbf{v}$, then $\|\mathbf{A}\mathbf{R}^{-1}\mathbf{T}\mathbf{v}\|_2^2 \leq (1-2\gamma)(1+\gamma) < 1$, but $\|\mathbf{A}\mathbf{R}^{-1}\mathbf{T}\mathbf{v}\|_2^2 = \|\mathbf{U}\mathbf{v}|_2^2 = 1$, a contradiction.

Finally, it follows that for all $i \in [n]$,

$$\|\mathbf{e}_i^T\mathbf{A}\mathbf{R}^{-1}\|_2^2 = \|e_i^T\mathbf{U}\mathbf{T}^{-1}\|_2^2 \geq (1-2\gamma)\|e_i^T\mathbf{U}\|_2^2 = (1-2\gamma)p_i.$$

Hence, $q_i \geq (1-\gamma)(1-2\gamma)p_i$, which for an appropriate choice of constant $\gamma \in (0,1)$, achieves $q_i \geq \beta p_i$, as desired. ■

2.5 Regression

We formally define the regression problem as follows.

Definition 20 *In the ℓ_2-Regression Problem, an $n \times d$ matrix $\mathbf{A}$ and an $n \times 1$ column vector $\mathbf{b}$ are given, together with an approximation parameter $\varepsilon \in [0,1)$. The goal is to output a vector $\mathbf{x}$ so that*

$$\|\mathbf{A}\mathbf{x} - \mathbf{b}\| \leq (1+\varepsilon) \min_{\mathbf{x}' \in \mathbb{R}^d} \big\|\mathbf{A}\mathbf{x}' - \mathbf{b}\big\|.$$

The following theorem is an immediate application of ℓ_2-subspace embeddings. The proof actually shows that there is a direct relationship between the time complexity of computing an ℓ_2-subspace embedding and the time complexity of approximately solving ℓ_2-regression. We give one instantiation of this relationship in the following theorem statement.

Theorem 21 *The ℓ_2-Regression Problem can be solved with probability .99 in $O(\mathrm{nnz}(A)) + \mathrm{poly}(d/\varepsilon)$ time.*

Proof: Consider the at most $(d+1)$-dimensional subspace L of $\mathbb{R}^n$ spanned by the columns of $\mathbf{A}$ together with the vector $\mathbf{b}$. Suppose we choose $\mathbf{S}$ to be a sparse embedding matrix with $r = d^2/\varepsilon^2 \text{poly}(\log(d/\varepsilon))$ rows. By Theorem 8, we have that with probability .99,

$$\forall \mathbf{y} \in L,\ \|\mathbf{S}\mathbf{y}\|_2^2 = (1 \pm \varepsilon)\|\mathbf{y}\|_2^2. \tag{10}$$

It follows that we can compute $\mathbf{S} \cdot \mathbf{A}$ followed by $\mathbf{S} \cdot \mathbf{b}$, and then let

$$\mathbf{x} = \operatorname*{argmin}_{\mathbf{x}' \in \mathbb{R}^d} \|\mathbf{S}\mathbf{A}\mathbf{x}' - \mathbf{S}\mathbf{b}\|_2.$$

By (10), it follows that $\mathbf{x}$ solves the ℓ_2-Regression Problem. The number of rows of $\mathbf{S}$ can be improved to $r = O(d/\varepsilon^2)$ by applying Theorem 9. ■

We note that Theorem 21 can immediately be generalized to other versions of regression, such as *constrained regression*. In this problem there is a constraint subset $\mathcal{C} \subseteq \mathbb{R}^d$ and the goal is, given an $n \times d$ matrix $\mathbf{A}$ and an $n \times 1$ column vector $\mathbf{b}$, to output a vector $\mathbf{x}$ for which

$$\|\mathbf{A}\mathbf{x} - \mathbf{b}\| \leq (1+\varepsilon) \min_{\mathbf{x}' \in \mathcal{C}} \|\mathbf{A}\mathbf{x}' - \mathbf{b}\|.$$

Inspecting the simple proof of Theorem 21 we see that (10) in particular implies

$$\forall \mathbf{x} \in \mathcal{C},\ \|\mathbf{S}(\mathbf{A}\mathbf{x} - \mathbf{b})\|_2 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2, \tag{11}$$

from which we have the following corollary. This corollary follows by replacing $\mathbf{A}$ and $\mathbf{b}$ with $\mathbf{S}\mathbf{A}$ and $\mathbf{S}\mathbf{b}$, where $\mathbf{S}$ has $O(d^2/\varepsilon^2)$ rows using Theorem 9.

Corollary 22 *The constrained ℓ_2 Regression Problem with constraint set $\mathcal{C}$ can be solved with probability .99 in $O(\text{nnz}(A)) + T(d, \varepsilon)$ time, where $T(d, \varepsilon)$ is the time to solve constrained ℓ_2 regression with constraint set $\mathcal{C}$ when $\mathbf{A}$ has $O(d^2/\varepsilon^2)$ rows and d columns.*

It is also possible to obtain a better dependence on ε than given by Theorem 21 and Corollary 22 in both the time and space, due to the fact that it is possible to choose the sparse subspace embedding $\mathbf{S}$ to have only $O(d^2/\varepsilon)$ rows. We present this as its own separate theorem. We only state the time bound for unconstrained regression.

The proof is due to Sarlós [105]. The key concept in the proof is that of the *normal equations*, which state for the optimal solution $\mathbf{x}$, $\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}$,

or equivalently, $\mathbf{A}^T(\mathbf{A}\mathbf{x}-\mathbf{b}) = \mathbf{0}$, that is, $\mathbf{A}\mathbf{x}-\mathbf{b}$ is orthogonal to the column space of $\mathbf{A}$. This is easy to see from the fact that the optimal solution $\mathbf{x}$ is such that $\mathbf{A}\mathbf{x}$ is the projection of $\mathbf{b}$ onto the column space of $\mathbf{A}$, which is the closest point of $\mathbf{b}$ in the column space of $\mathbf{A}$ in Euclidean distance.

Theorem 23 *If $\mathbf{S}$ is a sparse subspace embedding with $O(d^2/\varepsilon)$ rows, then with probability .99, the solution $\min_{\mathbf{x}'\in\mathbb{R}^d} \|\mathbf{S}\mathbf{A}\mathbf{x}'-\mathbf{S}\mathbf{b}\|_2 = (1\pm\varepsilon)\min_{\mathbf{x}\in\mathbb{R}^d} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2$.*

Proof: Let $\mathbf{x}'$ be $\text{argmin}_{\mathbf{x}'\in\mathbb{R}^d}\|\mathbf{S}(\mathbf{A}\mathbf{x}'-\mathbf{b})\|_2$, and let $\mathbf{x}$ be $\text{argmin}_{\mathbf{x}\in\mathbb{R}^d}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2$. It will be useful to reparameterize the problem in terms of an orthonormal basis $\mathbf{U}$ for the column space of $\mathbf{A}$. Let $\mathbf{U}\mathbf{y}' = \mathbf{A}\mathbf{x}'$ and $\mathbf{U}\mathbf{y} = \mathbf{A}\mathbf{x}$.

Because of the normal equations, we may apply the Pythagorean theorem,

$$\|\mathbf{A}\mathbf{x}' - \mathbf{b}\|_2^2 = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \|\mathbf{A}\mathbf{x}' - \mathbf{A}\mathbf{x}\|_2^2,$$

which in our new parameterization is,

$$\|\mathbf{U}\mathbf{y}' - \mathbf{b}\|_2^2 = \|\mathbf{U}\mathbf{y} - \mathbf{b}\|_2^2 + \|\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2^2.$$

It suffices to show $\|\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2^2 = O(\varepsilon)\|\mathbf{U}\mathbf{y} - \mathbf{b}\|_2^2$, as then the theorem will follow by rescaling ε by a constant factor. Since $\mathbf{U}$ has orthonormal columns, it suffices to show $\|\mathbf{y}' - \mathbf{y}\|_2^2 = O(\varepsilon)\|\mathbf{U}\mathbf{y} - \mathbf{b}\|_2^2$.

Conditioned on $\mathbf{S}$ being a $(1 \pm 1/2)$ ℓ_2-subspace embedding, which by Theorem 9 occurs with probability .999 for an $\mathbf{S}$ with an appropriate $O(d^2/\varepsilon)$ number of rows, we have

$$\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d\|_2 \leq \frac{1}{2}. \tag{12}$$

Hence,

$$\begin{aligned}
\|\mathbf{y}' - \mathbf{y}\|_2 &\leq \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2 + \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}(\mathbf{y}' - \mathbf{y}) - \mathbf{y}' - \mathbf{y}\|_2 \\
&\leq \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2 + \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U} - \mathbf{I}_d\|_2 \cdot \|(\mathbf{y}' - \mathbf{y})\|_2 \\
&\leq \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2 + \frac{1}{2} \cdot \|\mathbf{y}' - \mathbf{y}\|_2,
\end{aligned}$$

where the first inequality is the triangle inequality, the second inequality uses the sub-multiplicativity of the spectral norm, and the third inequality uses (12). Rearranging, we have

$$\|\mathbf{y}' - \mathbf{y}\|_2 \leq 2\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}(\mathbf{y}' - \mathbf{y})\|_2. \tag{13}$$

By the normal equations in the sketch space,

$$\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{U}\mathbf{y}' = \mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{b},$$

and so plugging into (13),

$$\|\mathbf{y}' - \mathbf{y}\|_2 \leq 2\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{U}\mathbf{y} - \mathbf{b})\|_2. \tag{14}$$

By the normal equations in the original space, $\mathbf{U}^T(\mathbf{U}\mathbf{y} - \mathbf{b}) = \mathbf{0}_{\mathrm{rank}(\mathbf{A})\times 1}$. By Theorem 14, $\mathbf{S}$ has the $(\varepsilon, \delta, 2)$-JL moment property, and so by Theorem 13,

$$\Pr_{\mathbf{S}}\left[\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{U}\mathbf{y} - \mathbf{b})\|_{\mathrm{F}} > 3\frac{\sqrt{\varepsilon}}{d}\|\mathbf{U}\|_{\mathrm{F}}\|\mathbf{U}\mathbf{y} - \mathbf{b}\|_{\mathrm{F}}\right] \leq \frac{1}{1000}.$$

Since $\|\mathbf{U}\|_{\mathrm{F}} \leq \sqrt{d}$, it follows that with probability .999, $\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{U}\mathbf{y} - \mathbf{b})\|_{\mathrm{F}} \leq 3\sqrt{\varepsilon}\|\mathbf{U}\mathbf{y} - \mathbf{b}\|_2$, and plugging into (14), together with a union bound over the two probability .999 events, completes the proof. ■

2.6 Machine precision regression

Here we show how to reduce the dependence on ε to logarithmic in the regression application, following the approaches in [103, 8, 27].

A classical approach to finding $\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|$ is to solve the normal equations $\mathbf{A}^\top\mathbf{A}\mathbf{x} = \mathbf{A}^\top\mathbf{b}$ via Gaussian elimination; for $\mathbf{A} \in \mathbb{R}^{n\times r}$ and $\mathbf{b} \in \mathbb{R}^{n\times 1}$, this requires $O(\mathrm{nnz}(\mathbf{A}))$ time to form $\mathbf{A}^\top\mathbf{b}$, $O(r\,\mathrm{nnz}(\mathbf{A}))$ time to form $\mathbf{A}^\top\mathbf{A}$, and $O(r^3)$ time to solve the resulting linear systems. (Another method is to factor $\mathbf{A} = \mathbf{Q}\mathbf{W}$, where $\mathbf{Q}$ has orthonormal columns and $\mathbf{W}$ is upper triangular; this typically trades a slowdown for a higher-quality solution.)

Another approach to regression is to apply an iterative method from the general class of Krylov or conjugate-gradient type algorithms to a preconditioned version of the problem. In such methods, an estimate $\mathbf{x}^{(m)}$ of a solution is maintained, for iterations $m = 0, 1\ldots$, using data obtained from previous iterations. The convergence of these methods depends on the *condition number* $\kappa(\mathbf{A}^\top\mathbf{A}) = \frac{\sup_{\mathbf{x},\|\mathbf{x}\|=1}\|\mathbf{A}\mathbf{x}\|^2}{\inf_{\mathbf{x},\|\mathbf{x}\|=1}\|\mathbf{A}\mathbf{x}\|^2}$ from the input matrix. A classical result ([80] via [91] or Theorem 10.2.6,[53]), is that

$$\frac{\left\|\mathbf{A}(\mathbf{x}^{(m)} - \mathbf{x}^*)\right\|^2}{\left\|\mathbf{A}(\mathbf{x}^{(0)} - \mathbf{x}^*)\right\|^2} \leq 2\left(\frac{\sqrt{\kappa(\mathbf{A}^\top\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A}^\top\mathbf{A})} + 1}\right)^m. \tag{15}$$

Thus the running time of CG-like methods, such as `CGNR` [53], depends on the (unknown) condition number. The running time per iteration is the time needed to compute matrix vector products $\mathbf{v} = \mathbf{A}\mathbf{x}$ and $\mathbf{A}^T\mathbf{v}$, plus $O(n+d)$ for vector arithmetic, or $O(\text{nnz}(\mathbf{A}))$.

Pre-conditioning reduces the number of iterations needed for a given accuracy: suppose for a non-singular matrix $\mathbf{R}$, the condition number $\kappa(\mathbf{R}^\top\mathbf{A}^\top\mathbf{A}\mathbf{R})$ is small. Then a conjugate gradient method applied to $\mathbf{A}\mathbf{R}$ would converge quickly, and moreover for iterate $\mathbf{y}^{(m)}$ that has error $\alpha^{(m)} \equiv \left\|\mathbf{A}\mathbf{R}\mathbf{y}^{(m)} - \mathbf{b}\right\|$ small, the corresponding $\mathbf{x} \leftarrow \mathbf{R}\mathbf{y}^{(m)}$ would have $\|\mathbf{A}\mathbf{x} - \mathbf{b}\| = \alpha^{(m)}$. The running time per iteration would have an additional $O(d^2)$ for computing products involving $\mathbf{R}$.

Suppose we apply a sparse subspace embedding matrix $\mathbf{S}$ to $\mathbf{A}$, and $\mathbf{R}$ is computed so that $\mathbf{S}\mathbf{A}\mathbf{R}$ has orthonormal columns, e.g., via a QR-decomposition of $\mathbf{S}\mathbf{A}$. If $\mathbf{S}$ is an ℓ_2-subspace embedding matrix to constant accuracy ε_0, for all unit $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{A}\mathbf{R}\mathbf{x}\|^2 = (1 \pm \varepsilon_0)\|\mathbf{S}\mathbf{A}\mathbf{R}\mathbf{x}\|^2 = (1 \pm \varepsilon_0)^2$. It follows that the condition number

$$\kappa(\mathbf{R}^\top\mathbf{A}^\top\mathbf{A}\mathbf{R}) \leq \frac{(1+\varepsilon_0)^2}{(1-\varepsilon_0)^2}.$$

That is, $\mathbf{A}\mathbf{R}$ is well-conditioned. Plugging this bound into (15), after m iterations $\left\|\mathbf{A}\mathbf{R}(\mathbf{x}^{(m)} - \mathbf{x}^*)\right\|^2$ is at most $2\varepsilon_0^m$ times its starting value.

Thus starting with a solution $\mathbf{x}^{(0)}$ with relative error at most 1, and applying $1 + \log(1/\varepsilon)$ iterations of a CG-like method with $\varepsilon_0 = 1/e$, the relative error is reduced to ε and the work is $O((\text{nnz}(\mathbf{A}) + d^2)\log(1/\varepsilon))$, plus the work to find $\mathbf{R}$. We have

Theorem 24 *The ℓ_2-regression problem can be solved up to a $(1+\varepsilon)$-factor with probability at least $99/100$ in*

$$O(\text{nnz}(\mathbf{A})\log(n/\varepsilon) + d^3\log^2 d + d^2\log(1/\varepsilon))$$

time.

The matrix $\mathbf{A}\mathbf{R}$ is so well-conditioned that a simple iterative improvement scheme has the same running time up to a constant factor. Again start with a solution $\mathbf{x}^{(0)}$ with relative error at most 1, and for $m \geq 0$, let $\mathbf{x}^{(m+1)} \leftarrow \mathbf{x}^{(m)} + \mathbf{R}^\top\mathbf{A}^\top(\mathbf{b} - \mathbf{A}\mathbf{R}\mathbf{x}^{(m)})$. Then using the normal equations,

$$\begin{aligned}
\mathbf{A}\mathbf{R}(\mathbf{x}^{(m+1)} - \mathbf{x}^*) &= \mathbf{A}\mathbf{R}(\mathbf{x}^{(m)} + \mathbf{R}^\top\mathbf{A}^\top(\mathbf{b} - \mathbf{A}\mathbf{R}\mathbf{x}^{(m)}) - \mathbf{x}^*) \\
&= (\mathbf{A}\mathbf{R} - \mathbf{A}\mathbf{R}\mathbf{R}^\top\mathbf{A}^\top\mathbf{A}\mathbf{R})(\mathbf{x}^{(m)} - \mathbf{x}^*) \\
&= \mathbf{U}(\mathbf{\Sigma} - \mathbf{\Sigma}^3)\mathbf{V}^\top(\mathbf{x}^{(m)} - \mathbf{x}^*),
\end{aligned}$$

where $\mathbf{AR} = \mathbf{U\Sigma V}^\top$ is the SVD of $\mathbf{AR}$.

For all unit $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{ARx}\|^2 = (1 \pm \varepsilon_0)^2$, and so we have that all singular values σ_i of $\mathbf{AR}$ are $1 \pm \varepsilon_0$, and the diagonal entries of $\mathbf{\Sigma} - \mathbf{\Sigma}^3$ are all at most $\sigma_i(1 - (1 - \varepsilon_0)^2) \leq \sigma_i 3\varepsilon_0$ for $\varepsilon_0 \leq 1$. Hence

$$\left\|\mathbf{AR}(\mathbf{x}^{(m+1)} - \mathbf{x}^*)\right\| \leq 3\varepsilon_0 \left\|\mathbf{AR}(\mathbf{x}^{(m)} - \mathbf{x}^*)\right\|,$$

and by choosing $\varepsilon_0 = 1/2$, say, $O(\log(1/\varepsilon))$ iterations suffice for this scheme also to attain ε relative error.

2.7 Polynomial fitting

A natural question is if *additional structure in* $\mathbf{A}$ can be non-trivially exploited to further accelerate the running time of ℓ_2-regression. Given that $\mathbf{A}$ is structured, perhaps we can run in time even faster than $O(\text{nnz}(\mathbf{A}))$. This was studied in [10, 9], and we shall present the result in [10].

Perhaps one of the oldest regression problems is polynomial fitting. In this case, given a set of samples $(z_i, b_i) \in \mathbb{R} \times \mathbb{R}$, for $i = 1, 2, \ldots, n$, we would like to choose coefficients $\beta_0, \beta_1, \ldots, \beta_q$ of a degree-q univariate polynomial $b = \sum_{i=1}^q \beta_i z^i$ which best fits our samples. Setting this up as a regression problem, the corresponding matrix $\mathbf{A}$ is $n \times (q+1)$ and is a *Vandermonde matrix*. Despite the fact that $\mathbf{A}$ may be dense, we could hope to solve regression in time faster than $O(\text{nnz}(\mathbf{A})) = O(nq)$ using its Vandermonde structure.

We now describe the problem more precisely, starting with a definition.

Definition 25 *(Vandermonde Matrix) Let $x_0, x_1, \ldots, x_{n-1}$ be real numbers. The Vandermonde matrix, denoted $\mathbf{V}_{n,q}(x_0, x_1, \ldots, x_{n-1})$, has the form:*

$$\mathbf{V}_{n,q}(x_1, x_1, \ldots, x_{n-1}) = \begin{pmatrix} 1 & x_0 & \ldots & x_0^{q-1} \\ 1 & x_1 & \ldots & x_1^{q-1} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & x_{n-1} & \ldots & x_{n-1}^{q-1} \end{pmatrix}$$

Vandermonde matrices of dimension $n \times q$ require only $O(n)$ implicit storage and admit $O(n \log^2 q)$ matrix-vector multiplication time (see, e.g., Theorem 2.11 of [114]). It is also possible to consider block-Vandermonde matrices as in [10]; for simplicity we will only focus on the simplest polynomial fitting problem here, in which Vandermonde matrices suffice for the discussion.

We consider regression problems of the form $\min_{\mathbf{x}\in\mathbb{R}^q} \|\mathbf{V}_{n,q}\mathbf{x} - \mathbf{b}\|_2$, or the approximate version, where we would like to output an $\mathbf{x}' \in \mathbb{R}^q$ for which

$$\|\mathbf{V}_{n,q}\mathbf{x}' - \mathbf{b}\|_2 \leq (1+\varepsilon) \min_{\mathbf{x}\in\mathbb{R}^d} \|\mathbf{V}_{n,q}\mathbf{x} - \mathbf{b}\|_2.$$

We call this the *ℓ_2-Polynomial Fitting Problem.*

Theorem 26 *(ℓ_2-Polynomial Fitting)[10] There is an algorithm that solves the ℓ_2-Polynomial Fitting Problem in time $O(n \log^2 q) + \text{poly}(q\varepsilon^{-1})$. By combining sketching methods with preconditioned iterative solvers, we can also obtain logarithmic dependence on ε.*

Note that since $\text{nnz}(\mathbf{V}_{n,q}) = nq$ and the running time of Theorem 26 is $O(n \log^2 q)$, this provides a sketching approach that operates faster than "input-sparsity" time. It is also possible to extend Theorem 26 to ℓ_1-regression, see [10] for details.

The basic intuition behind Theorem 26 is to try to compute $\mathbf{S} \cdot \mathbf{V}_{n,q}$ for a sparse embedding matrix $\mathbf{S}$. Naively, this would take $O(nq)$ time. However, since $\mathbf{S}$ contains a single non-zero entry per column, we can actually think of the product $\mathbf{S} \cdot \mathbf{V}_{n,q}$ as r *vector-matrix products* $x^1 \cdot \mathbf{V}^1_{n,q}, \ldots, x^r \cdot \mathbf{V}^r_{n,q}$, where x^i is the vector with coordinates $j \in [n]$ for which $h(j) = i$, and $\mathbf{V}^i_{n,q}$ is the row-submatrix of $\mathbf{V}_{n,q}$ consisting only of those rows $j \in [n]$ for which $h(j) = i$. To compute each of these vector-matrix products, we can now appeal to the fast matrix-vector multiplication algorithm associated with Vandermonde matrices, which is similar to the Fast Fourier Transform. Thus, we can compute each $x^i \cdot \mathbf{V}^i_{n,q}$ in time proportional to the number of rows of $\mathbf{V}^i_{n,q}$, times a factor of $\log^2 q$. In total we can compute all matrix-vector products in $O(n \log^2 q)$ time, thereby computing $\mathbf{S}\mathbf{V}_{n,q}$, which we know is an ℓ_2-subspace embedding. We can also compute $\mathbf{S}\mathbf{b}$ in $O(n)$ time, and now can solve the sketched problem $\min_x \|\mathbf{S}\mathbf{V}_{n,q}x - \mathbf{S}\mathbf{b}\|_2$ in $\text{poly}(q/\varepsilon)$ time.

3 Least Absolute Deviation Regression

While least squares regression is arguably the most used form of regression in practice, it has certain non-robustness properties that make it unsuitable for some applications. For example, oftentimes the noise in a regression problem is drawn from a normal distribution, in which case least squares regression would work quite well, but if there is noise due to measurement error or a different underlying noise distribution, the least squares regression solution may overfit this noise since the cost function squares each of its summands.

A more robust alternative is least absolute deviation regression, or ℓ_1-regression, $\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1 = \sum_{i=1}^{n} |\mathbf{b}_i - \langle \mathbf{A}_{i,*}, \mathbf{x} \rangle|$. The ℓ_1-norm is much less well-behaved than the ℓ_2-norm, e.g., it is not invariant under rotation, not everywhere differentiable, etc. There is also no closed-form solution for an ℓ_1-regression problem in general, as a special case of it is the geometric median or Fermat-Weber problem, for which there is no closed form solution.

Nevertheless, ℓ_1-regression is much less sensitive to outliers. It is also the maximum likelihood estimator (MLE) when the noise in the regression problem is i.i.d. Laplacian of zero median. In this chapter we will focus on recent advances in solving ℓ_1-regression using sketching. To do so, we first describe a sampling-based solution. We note that many of the results in this section generalize to ℓ_p-regression for $p > 1$. See [27, 92, 124] for works on this. This general line of work was introduced by Clarkson [25], though our exposition will mostly follow that of [32] and the sketching speedups built on top of it [108, 26, 92, 124].

Chapter Overview: In §3.1 we show how one can adapt the idea of leverage score sampling in §2.4 for ℓ_2 to provide an initial sampling-based algorithm for ℓ_1-regression. In §3.2 we introduce the notion of a subspace embedding for the ℓ_1-norm and show how if we had such an object, it could be used in the context of ℓ_1-regression. We postpone one technical detail in this application to §3.3, which shows how to combine ℓ_1-subspace embeddings together with Gaussian sketching to make the technique of using ℓ_1-subspace embeddings in §3.2 efficient. In §3.4 we turn to the task of constructing ℓ_1-subspace embeddings. We do this using Cauchy random variables. This leads to an ℓ_1-regression algorithm running in $O(nd^2 \log d) + \text{poly}(d/\varepsilon)$. In §3.5 we then speed this up even further by replacing the dense matrix of Cauchy random variables in the previous section with a product of a sparse ℓ_2-subspace embedding and a diagonal matrix of exponential random variables. This leads to an overall time of $O(\text{nnz}(A) \log) + \text{poly}(d/\varepsilon)$. Finally, in §3.6 we discuss one application of ℓ_1-regression to ℓ_1-Hyperplane Approximation.

3.1 Sampling-Based solution

One of the most natural ways of solving a regression problem is by sampling. Let us augment the $n \times d$ design matrix $\mathbf{A}$ in the regression problem to an $n \times (d+1)$ matrix by including the $\mathbf{b}$ vector as the $(d+1)$-st column.

Let $p \in [0,1]^n$. Suppose we form a submatrix of $\mathbf{A}$ by including each row of $\mathbf{A}$ in the submatrix independently with probability p_i. Let us write this as $\mathbf{S} \cdot \mathbf{A}$, where $\mathbf{S}$ is a diagonal $n \times n$ matrix with $\mathbf{S}_{i,i} = 1/p_i$ if row i was included in the sample, and $\mathbf{S}_{i,i} = 0$ otherwise. Then $\mathbf{E}[\mathbf{S} \cdot \mathbf{A}] = \mathbf{A}$, and so for any fixed $\mathbf{x}$, $\mathbf{E}[\mathbf{S} \cdot \mathbf{A}\mathbf{x}] = \mathbf{A}\mathbf{x}$.

What we would like is that for all

$$\forall \mathbf{x} \in \mathbb{R}^{d+1},\ \|\mathbf{S} \cdot \mathbf{A}\mathbf{x}\|_1 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{x}\|_1, \tag{16}$$

that is, $\mathbf{S}$ is an oblivious subspace embedding for $\mathbf{A}$. Note that although $\mathbf{S} \cdot \mathbf{A}$ is an $n \times d$ matrix, in expectation it has only $r = \sum_{i=1}^n p_i$ non-zero rows, and so we can throw away all of the zero rows. It follows that if r is small, one could then afford to directly solve the *constrained* regression problem:

$$\min_{\mathbf{x} \in \mathbb{R}^{d+1}, \mathbf{x}_{d+1} = -1} \|\mathbf{S}\mathbf{A}\mathbf{x}\|_1,$$

using linear programming. This would now take time $\text{poly}(r, d)$, which is a significant savings over solving the problem directly, e.g., if r is much smaller or independent of n. Note that the constraint $\mathbf{x}_{d+1} = -1$ can be directly incorporated into a linear program for solving this problem, and only slightly increases its complexity.

We are left with the task of showing (16). To do so, fix a particular vector $\mathbf{x} \in \mathbb{R}^d$. Define the random variable $Z_i = |\mathbf{A}_{i,*}\mathbf{x}|/p_i$, so that for $Z = \sum_{i=1}^n Z_i$, we have $\mathbf{E}[Z] = \|\mathbf{A}\mathbf{x}\|_1$. We would like to understand how large each Z_i can be, and what the variance of Z is. We would like these quantities to be small, which at first glance seems hard since p cannot depend on $\mathbf{x}$.

One way of bounding Z_i is to write $\mathbf{A} = \mathbf{U} \cdot \boldsymbol{\tau}$ for an $n \times d$ matrix $\mathbf{U}$ and a $d \times d$ change of basis matrix $\boldsymbol{\tau}$. Since $\mathbf{U}$ does not depend on any particular vector $\mathbf{x}$, one could hope to define p in terms of $\mathbf{U}$ for a particularly good choice of basis $\mathbf{U}$ for the column space of $\mathbf{A}$. Note that one has

$$|\mathbf{A}_{i,*}, \mathbf{x}|/p_i = |\mathbf{U}_{i,*}\boldsymbol{\tau}\mathbf{x}|/p_i \leq \|\mathbf{U}_{i,*}\|_1 \cdot \|\boldsymbol{\tau}\mathbf{x}\|_\infty/p_i, \tag{17}$$

where the inequality follows by Hölder's inequality.

A natural choice at this point to bound the RHS of (17) is to define $p_i = \min(1, r \cdot \frac{\|\mathbf{U}_{i,*}\|}{\sum_{j=1}^n \|\mathbf{U}_{j,*}\|_1})$, where recall r is about the expected number of

rows we wish to sample (the expected number of rows sampled may be less than r since p_i is a probability and so is upper-bounded by 1). For later purposes, it will be helpful to instead allow

$$p_i \geq \min(1, \zeta \cdot r \cdot \frac{\|\mathbf{U}_{i,*}\|}{\sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1}),$$

where $\zeta \in (0, 1]$ can be thought of as a relaxation parameter which will allow for more efficient algorithms.

Note that for those i for which $\zeta \cdot r \cdot \frac{\|\mathbf{U}_{i,*}\|}{\sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1} \geq 1$, the i-th row $\mathbf{A}_{i,*}$ will always be included in the sample, and therefore will not affect the variance of the sampling process.

Let us now consider those i for which $\zeta \cdot r \cdot \frac{\|\mathbf{U}_{i,*}\|}{\sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1} < 1$. For such i one has

$$Z_i = |\mathbf{A}_{i,*}, \mathbf{x}|/p_i \leq (\sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1) \cdot \|\boldsymbol{\tau}\mathbf{x}\|_\infty/(r\zeta) = \alpha \cdot \beta \|\mathbf{A}\mathbf{x}\|_1/(r\zeta), \quad (18)$$

where $\alpha = \sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1$ and $\beta = \sup_{\mathbf{x}} \frac{\|\boldsymbol{\tau}\mathbf{x}\|_\infty}{\|\mathbf{A}\mathbf{x}\|_1}$.

In order for Z_i to never be too large, we would like to choose a $\mathbf{U}$ so that α and β are as small as possible. This motivates the following definition of a well-conditioned basis for the 1-norm. For ease of notation, let $\|\mathbf{U}\|_1 = \sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1$.

Definition 27 *(Well-conditioned basis for the 1-norm)(see [32]) Let $\mathbf{A}$ be an $n \times d$ matrix. An $n \times d$ matrix $\mathbf{U}$ is an $(\alpha, \beta, 1)$-well conditioned basis for the column space of $\mathbf{A}$ if (1) $\|\mathbf{U}\|_1 \leq \alpha$, and (2) for all $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_\infty \leq \beta \|\mathbf{U}\mathbf{x}\|_1$.*

Note that our definition of α and β above coincide with that in Definition 27, in particular the definition of β, since $\|\mathbf{U}(\boldsymbol{\tau}\mathbf{x})\|_1 = \|\mathbf{A}\mathbf{x}\|_1$ by definition of $\mathbf{U}$ and $\boldsymbol{\tau}$.

Fortunately, well-conditioned bases with $\alpha, \beta \leq \text{poly}(d)$ exist and can be efficiently computed. We will sometimes simply refer to $\mathbf{U}$ as a well-conditioned basis if α and β are both bounded by $\text{poly}(d)$. That such bases exist is due to a theorem of Auerbach [7, 11], which shows that $\alpha = d$ and $\beta = 1$ suffice. However, we are not aware of an efficient algorithm which achieves these values. The first efficient algorithm for finding a well-conditioned basis is due to Clarkson [25], who achieved a running time of $O(nd^5 \log n) + \text{poly}(d)$. The same running time was achieved by Dasgupta et

al. [32], who improved the concrete values of α and β. We will see that one can in fact compute such bases much faster using sketching techniques below, but let us first see how these results already suffice to solve ℓ_1-regression in $O(nd^5 \log n) + \text{poly}(d/\varepsilon)$ time.

Returning to (18), we have the bound

$$Z_i = |\mathbf{A}_{i,*}, \mathbf{x}|/p_i \leq \text{poly}(d)\|\mathbf{A}\mathbf{x}\|_1/(r\zeta).$$

Using this bound together with independence of the sampled rows,

$$\begin{aligned}
\mathbf{Var}[Z] &= \sum_{i=1}^{n} \mathbf{Var}[Z_i] = \sum_{i|p_i<1} \mathbf{Var}[Z_i] \leq \sum_{i|p_i<1} \mathbf{E}[Z_i^2] = \sum_{i|p_i<1} \frac{|\mathbf{A}_{i,*}, \mathbf{x}|^2}{p_i} \\
&\leq \max_{i|p_i<1} \frac{|\mathbf{A}_{i,*}, \mathbf{x}|}{p_i} \sum_{i|p_i<1} |\mathbf{A}_{i,*}, \mathbf{x}| \\
&\leq \frac{\text{poly}(d)\|\mathbf{A}\mathbf{x}\|_1^2}{r\zeta}.
\end{aligned}$$

We have computed $\mathbf{E}[Z]$ and bounded $\mathbf{Var}[Z]$ as well as $\max_{i|p_i<1} Z_i$, and can now use strong tail bounds to bound the deviation of Z from its expectation. We use the following tail inequalities.

Theorem 28 (Bernstein inequality [90]) *Let $Z_i \geq 0$ be independent random variables with $\sum_i \mathrm{E}[Z_i^2] < \infty$, and define $Z = \sum_i Z_i$. Then, for any $t > 0$,*

$$\Pr[Z \leq \mathrm{E}[Z] - t] \leq \exp\left(\frac{-t^2}{2\sum_i \mathrm{E}[Z_i^2]}\right).$$

Moreover, if $Z_i - \mathbf{E}[Z_i] \leq \Delta$ for all i, we have

$$\Pr[Z \geq \mathbf{E}[Z] + \gamma] \leq \exp\left(\frac{-\gamma^2}{2\mathbf{Var}[Z] + 2\gamma\Delta/3}\right).$$

Plugging our bounds into 28, we have

$$\begin{aligned}
\Pr[Z \leq \|\mathbf{A}\mathbf{x}\|_1 - \varepsilon\|\mathbf{A}\mathbf{x}\|_1] &\leq \exp\left(\frac{-\varepsilon^2\|\mathbf{A}\mathbf{x}\|_1^2 r\zeta}{2\text{poly}(d)\|\mathbf{A}\mathbf{x}\|_1^2}\right) \\
&\leq \exp\left(\frac{-\varepsilon^2 r\zeta}{2\text{poly}(d)}\right),
\end{aligned}$$

and also

$$\begin{aligned}\Pr[Z \geq \|\mathbf{Ax}\|_1 + \varepsilon\|\mathbf{Ax}\|_1] &\leq \exp\left(\frac{-\varepsilon^2\|\mathbf{Ax}\|_1^2}{2\frac{\mathrm{poly}(d)\|\mathbf{Ax}\|_1^2}{r\zeta} + 2\varepsilon\frac{\|\mathbf{Ax}\|_1^2\mathrm{poly}(d)}{3r\zeta}}\right) \\ &\leq \exp\left(\frac{-\varepsilon^2 r\zeta}{2\mathrm{poly}(d) + 2\varepsilon\mathrm{poly}(d)/3}\right).\end{aligned}$$

Setting $r = \varepsilon^{-2}\mathrm{poly}(d)/\zeta$ for a large enough polynomial in d allows us to conclude that for any fixed $\mathbf{x} \in \mathbb{R}^d$,

$$\Pr[Z \in (1 \pm \varepsilon)\|\mathbf{Ax}\|_1] \geq 1 - (\varepsilon/4)^d. \tag{19}$$

While this implies that Z is a $(1+\varepsilon)$-approximation with high probability for a fixed $\mathbf{x}$, we now need an argument for all $\mathbf{x} \in \mathbb{R}^d$. To prove that $\|\mathbf{SAx}\|_1 = (1 \pm \varepsilon)\|\mathbf{Ax}\|_1$ for all $\mathbf{x}$, it suffices to prove the statement for all $\mathbf{y} \in \mathbb{R}^n$ for which $\mathbf{y} = \mathbf{Ax}$ for some $\mathbf{x} \in \mathbb{R}^d$ and $\|\mathbf{y}\|_1 = 1$. Indeed, since $\mathbf{SA}$ is a linear map, it will follow that $\|\mathbf{SAx}\|_1 = (1 \pm \varepsilon)\|\mathbf{Ax}\|_1$ for all $\mathbf{x}$ by linearity.

Let $\mathcal{B} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = \mathbf{Ax} \text{ for some } \mathbf{x} \in \mathbb{R}^d \text{ and } \|\mathbf{y}\|_1 = 1\}$. We seek a finite subset of $\mathcal{B}$, denoted $\mathcal{N}$, which is an ε-net, so that if $\|\mathbf{Sw}\|_1 = (1 \pm \varepsilon)\|\mathbf{w}\|_1$ for all $\mathbf{w} \in \mathcal{N}$, then it implies that $\|\mathbf{Sy}\|_1 = (1 \pm \varepsilon)\|\mathbf{y}\|_1$ for all $\mathbf{y} \in \mathcal{B}$. The argument will be similar to that in §2.1 for the ℓ_2 norm, though the details are different.

It suffices to choose $\mathcal{N}$ so that for all $\mathbf{y} \in \mathcal{B}$, there exists a vector $\mathbf{w} \in \mathcal{N}$ for which $\|\mathbf{y} - \mathbf{w}\|_1 \leq \varepsilon$. Indeed, in this case note that

$$\|\mathbf{Sy}\|_1 = \|\mathbf{Sw} + \mathbf{S}(\mathbf{y} - \mathbf{w})\|_1 \leq \|\mathbf{Sw}\|_1 + \|\mathbf{S}(\mathbf{y} - \mathbf{w})\|_1 \leq 1 + \varepsilon + \|\mathbf{S}(\mathbf{y} - \mathbf{w})\|_1.$$

If $\mathbf{y} - \mathbf{w} = 0$, we are done. Otherwise, suppose α is such that $\alpha\|\mathbf{y} - \mathbf{w}\|_1 = 1$. Observe that $\alpha \geq 1/\varepsilon$, since, $\|\mathbf{y} - \mathbf{w}\|_1 \leq \varepsilon$ yet $\alpha\|\mathbf{y} - \mathbf{w}\|_1 = 1$.

Then $\alpha(\mathbf{y} - \mathbf{w}) \in \mathcal{B}$, and we can choose a vector $\mathbf{w}^2 \in \mathcal{N}$ for which $\|\alpha(\mathbf{y} - \mathbf{w}) - \mathbf{w}^2\|_1 \leq \varepsilon$, or equivalently, $\|\mathbf{y} - \mathbf{w} - \mathbf{w}^2/\alpha\|_1 \leq \varepsilon/\alpha \leq \varepsilon^2$. Hence,

$$\begin{aligned}\|\mathbf{S}(\mathbf{y} - \mathbf{w})\|_1 &= \|\mathbf{S}\mathbf{w}^2/\alpha + \mathbf{S}(\mathbf{y} - \mathbf{w} - \mathbf{w}^2/\alpha)\|_1 \\ &\leq (1+\varepsilon)/\alpha + \|\mathbf{S}(\mathbf{y} - \mathbf{w} - \mathbf{w}^2/\alpha)\|_1.\end{aligned}$$

Repeating this argument, we inductively have that

$$\|\mathbf{Sy}\|_1 \leq \sum_{i \geq 0}(1+\varepsilon)\varepsilon^i \leq (1+\varepsilon)/(1-\varepsilon) \leq 1 + O(\varepsilon).$$

By a similar argument, we also have that

$$\|\mathbf{Sy}\|_1 \geq 1 - O(\varepsilon).$$

Thus, by rescaling ε by a constant factor, we have that $\|\mathbf{Sy}\|_1 = 1 \pm \varepsilon$ for all vectors $\mathbf{y} \in \mathcal{B}$.

Lemma 29 *There exists an ε-net $\mathcal{N}$ for which $|\mathcal{N}| \leq (2/\varepsilon)^d$.*

Proof: For a parameter γ and point $\mathbf{p} \in \mathbb{R}^n$, define

$$B(\mathbf{p}, \gamma, \mathbf{A}) = \{\mathbf{q} = \mathbf{Ax} \text{ for some } \mathbf{x} \text{ and } \|\mathbf{p} - \mathbf{q}\|_1 \leq \gamma\}.$$

Then $B(\varepsilon, 0)$ is a d-dimensional polytope with a (d-dimensional) volume denoted $|B(\varepsilon, 0)|$. Moreover, $B(1, 0)$ and $B(\varepsilon/2, 0)$ are similar polytopes, namely, $B(1, 0) = (2/\varepsilon)B(\varepsilon/2, 0)$. As such, $|B(1, 0)| = (2/\varepsilon)^d |B(\varepsilon/2, 0)|$.

Let $\mathcal{N}$ be a maximal subset of $\mathbf{y} \in \mathbb{R}^n$ in the column space of $\mathbf{A}$ for which $\|\mathbf{y}\|_1 = 1$ and for all $\mathbf{y} \neq \mathbf{y}' \in \mathcal{N}$, $\|\mathbf{y} - \mathbf{y}'\|_1 > \varepsilon$. Since $\mathcal{N}$ is maximal, it follows that for all $\mathbf{y} \in \mathcal{B}$, there exists a vector $\mathbf{w} \in \mathcal{N}$ for which $\|\mathbf{y} - \mathbf{w}\|_1 \leq \varepsilon$. Moreover, for all $\mathbf{y} \neq \mathbf{y}' \in \mathcal{N}$, $B(\mathbf{y}, \varepsilon/2, \mathbf{A})$ and $B(\mathbf{y}', \varepsilon/2, \mathbf{A})$ are disjoint, as otherwise by the triangle inequality, $\|\mathbf{y} - \mathbf{y}'\|_1 \leq \varepsilon$, a contradicition. It follows by the previous paragraph that $\mathcal{N}$ can contain at most $(2/\varepsilon)^d$ points. ■

By applying (19) and a union bound over the points in $\mathcal{N}$, and rescaling ε by a constant factor, we have thus shown the following theorem.

Theorem 30 *The above sampling algorithm is such that with probability at least $1 - 2^{-d}$, simultaneously for all $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{SAx}\|_1 = (1 \pm \varepsilon)\|\mathbf{Ax}\|_1$. The expected number of non-zero rows of $\mathbf{SA}$ is at most $r = \varepsilon^{-2}\text{poly}(d)/\zeta$. The overall time complexity is $T_{wcb} + \text{poly}(d/\varepsilon)$, where T_{wcb} is the time to compute a well-conditioned basis. Setting $T_{wcb} = O(nd^5 \log n)$ suffices.*

3.2 The Role of subspace embeddings for L1-Regression

The time complexity of the sampling-based algorithm for ℓ_1-Regression in the previous section is dominated by the computation of a well-conditioned basis. In this section we will design subspace embeddings with respect to the ℓ_1-norm and show how they can be used to speed up this computation. Unlike for ℓ_2, the distortion of our vectors in our subspace will not be $1 + \varepsilon$, but rather a larger factor that depends on d. Still, the distortion does not depend on n, and this will be sufficient for our applications. This will be

because, with this weaker distortion, we will still be able to form a well-conditioned basis, and then we can apply Theorem 30 to obtain a $(1+\varepsilon)$-approximation to ℓ_1-regression.

Definition 31 *(Subspace Embedding for the ℓ_1-Norm) We will say a matrix $\mathbf{S}$ is an ℓ_1-subspace embedding for an $n \times d$ matrix $\mathbf{A}$ if there are constants $c_1, c_2 > 0$ so that for all $\mathbf{x} \in \mathbb{R}^d$,*

$$\|\mathbf{Ax}\|_1 \leq \|\mathbf{SAx}\|_1 \leq d^{c_1}\|\mathbf{Ax}\|_1,$$

and $\mathbf{S}$ has at most d^{c_2} rows.

Before discussing the existence of such embeddings, let us see how they can be used to speed up the computation of a well-conditioned basis.

Lemma 32 *([108]) Suppose $\mathbf{S}$ is an ℓ_1-subspace embedding for an $n \times d$ matrix $\mathbf{A}$. Let $\mathbf{Q} \cdot \mathbf{R}$ be a QR-decomposition of $\mathbf{SA}$, i.e., $\mathbf{Q}$ has orthonormal columns (in the standard ℓ_2 sense) and $\mathbf{Q} \cdot \mathbf{R} = \mathbf{SA}$. Then $\mathbf{AR}^{-1}$ is a $(d^{c_1+c_2/2+1}, 1, 1)$-well-conditioned basis.*

Proof: We have

$$\begin{aligned}
\alpha &= \|\mathbf{AR}^{-1}\|_1 = \sum_{i=1}^{d} \|\mathbf{AR}^{-1}e_i\|_1 \leq d^{c_1} \sum_{i=1}^{d} \|\mathbf{SAR}^{-1}\mathbf{e}_i\|_1 \\
&\leq d^{c_1+c_2/2} \sum_{i=1}^{d} \|\mathbf{SAR}^{-1}\mathbf{e}_i\|_2 \\
&\leq d^{c_1+c_2/2} \sum_{i=1}^{d} \|\mathbf{Qe}_i\|_2 \\
&= d^{c_1+c_2/2+1}.
\end{aligned}$$

Recall we must bound β, where β is minimal for which for all $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_\infty \leq \beta\|\mathbf{AR}^{-1}\mathbf{x}\|_1$. We have

$$\begin{aligned}
\|\mathbf{AR}^{-1}\mathbf{x}\|_1 &\geq \|\mathbf{SAR}^{-1}\mathbf{x}\|_1 \\
&\geq \|\mathbf{SAR}^{-1}\mathbf{x}\|_2 \\
&= \|\mathbf{Qx}\|_2 \\
&= \|\mathbf{x}\|_2 \\
&\geq \|\mathbf{x}\|_\infty,
\end{aligned}$$

and so $\beta = 1$. ■

Note that $\mathbf{SA}$ is a $d^{c_2} \times d$ matrix, and therefore its QR decomposition can be computed in $O(d^{c_2+2})$ time. One can also compute $\mathbf{A} \cdot \mathbf{R}^{-1}$ in $O(nd^2)$ time, which could be sped up with fast matrix multiplication, though we will see a better way of speeding this up below. By Lemma 32, provided $\mathbf{S}$ is a subspace embedding for $\mathbf{A}$ with constants $c_1, c_2 > 0$, $\mathbf{AR}^{-1}$ is a $(d^{c_1+c_2/2+1}, 1, 1)$-well-conditioned basis, and so we can improve the time complexity of Theorem 30 to $T_{mm} + O(nd^2) + \text{poly}(d/\varepsilon)$, where T_{mm} is the time to compute the matrix-matrix product $\mathbf{S} \cdot \mathbf{A}$.

We are thus left with the task of producing an ℓ_1-subspace embedding for $\mathbf{A}$. There are many ways to do this non-obliviously [77, 106, 17, 113, 66], but they are slower than the time bounds we can achieve using sketching.

We show in §3.4 that by using sketching we can achieve $T_{mm} = O(nd^2 \log d)$, which illustrates several main ideas and improves upon Theorem 30. We will then show how to improve this to $T_{mm} = O(\text{nnz}(\mathbf{A}))$ in §3.5. Before doing so, let us first see how, given $\mathbf{R}$ for which $\mathbf{AR}^{-1}$ is well-conditioned, we can improve the $O(nd^2)$ time for computing a representation of $\mathbf{AR}^{-1}$ which is sufficient to perform the sampling in Theorem 30.

3.3 Gaussian sketching to speed up sampling

Lemma 32 shows that if $\mathbf{S}$ is an ℓ_1-subspace embedding for an $n \times d$ matrix $\mathbf{A}$, and $\mathbf{Q} \cdot \mathbf{R}$ is a QR-decomposition of $\mathbf{SA}$, then $\mathbf{AR}^{-1}$ is a well-conditioned basis.

Computing $\mathbf{AR}^{-1}$, on the other hand, naively takes $O(\text{nnz}(\mathbf{A})d)$ time. However, observe that to do the sampling in Theorem 30, we just need to be able to compute the probabilities p_i, for $i \in [n]$, where recall

$$p_i \geq \min(1, \zeta \cdot r \cdot \frac{\|\mathbf{U}_{i,*}\|_1}{\sum_{j=1}^{n} \|\mathbf{U}_{j,*}\|_1}), \tag{20}$$

where $\zeta \in (0, 1]$, and $\mathbf{U} = \mathbf{AR}^{-1}$ is the well-conditioned basis. This is where ζ comes in to the picture.

Instead of computing the matrix product $\mathbf{A} \cdot \mathbf{R}^{-1}$ directly, one can choose a $d \times t$ matrix $\mathbf{G}$ of i.i.d. $N(0, 1/t)$ random variables, for $t = O(\log n)$ and first compute $\mathbf{R}^{-1} \cdot \mathbf{G}$. This matrix can be computed in $O(td^2)$ time and only has t columns, and so now computing $\mathbf{AR}^{-1}\mathbf{G} = \mathbf{A} \cdot (\mathbf{R}^{-1} \cdot \mathbf{G})$ can be computed in $O(\text{nnz}(\mathbf{A})t) = O(\text{nnz}(\mathbf{A}) \log n)$ time. By choosing the parameter $\varepsilon = 1/2$ of Lemma 18 we have for all $i \in [n]$, that $\frac{1}{2}\|(\mathbf{AR}^{-1})_i\|_2 \leq$

$\|(\mathbf{AR}^{-1}\mathbf{G})_i\|_2 \leq 2\|(\mathbf{AR}^{-1})_i\|_2$. Therefore,

$$\sum_{j=1}^{n} \|(\mathbf{AR}^{-1}\mathbf{G})_j\|_1 \leq \sqrt{d}\sum_{j=1}^{n} \|(\mathbf{AR}^{-1}\mathbf{G})_j\|_2 \leq 2\sqrt{d}\sum_{j=1}^{n} \|(\mathbf{AR}^{-1})_j\|_1,$$

and also

$$\|(\mathbf{AR}^{-1}\mathbf{G})_j\|_1 \geq \|(\mathbf{AR}^{-1}\mathbf{G})_j\|_2 \geq \frac{1}{2}\|(\mathbf{AR}^{-1})_j\|_2 \geq \frac{1}{2\sqrt{d}}\|(\mathbf{AR}^{-1})_j\|_1.$$

It follows that for

$$p_i = \min(1, r \cdot \frac{\|(\mathbf{AR}^{-1}\mathbf{G})_i\|_1}{\sum_{j=1}^{n} \|(\mathbf{AR}^{-1}\mathbf{G})_j\|_1}),$$

we have that (20) holds with $\zeta = 1/(4d)$.

We note that a tighter anaylsis is possible, in which $\mathbf{G}$ need only have $O(\log(d\varepsilon^{-1}\log n))$ columns, as shown in [26].

3.4 Subspace embeddings using cauchy random variables

The Cauchy distribution, having density function $p(x) = \frac{1}{\pi} \cdot \frac{1}{1+x^2}$, is the unique 1-stable distribution. That is to say, if $C_1, \ldots, C_M$ are independent Cauchys, then $\sum_{i\in[M]} \gamma_i C_i$ is distributed as a Cauchy scaled by $\gamma = \sum_{i\in[M]} |\gamma_i|$.

The absolute value of a Cauchy distribution has density function $f(x) = 2p(x) = \frac{2}{\pi}\frac{1}{1+x^2}$. The cumulative distribution function $F(z)$ of it is

$$F(z) = \int_0^z f(z)dz = \frac{2}{\pi}\arctan(z).$$

Note also that since $\tan(\pi/4) = 1$, we have $F(1) = 1/2$, so that 1 is the median of this distribution.

Although Cauchy random variables do not have an expectation, and have infinite variance, some control over them can be obtained by clipping them. The first use of such a truncation technique in algorithmic applications that we are aware of is due to Indyk [63].

Lemma 33 *Consider the event $\mathcal{E}$ that a Cauchy random variable X satisfies $|X| \leq M$, for some parameter $M \geq 2$. Then there is a constant $c > 0$ for which $\Pr[\mathcal{E}] \geq 1 - \frac{2}{\pi M}$ and $\mathbf{E}[|X| \mid \mathcal{E}] \leq c\log M$, where $c > 0$ is an absolute constant.*

Proof:

$$\Pr[\mathcal{E}] = F(M) = \frac{2}{\pi}\tan^{-1}(M) = 1 - \frac{2}{\pi}\tan^{-1}\left(\frac{1}{M}\right) \geq 1 - \frac{2}{\pi M}. \qquad (21)$$

Hence, for $M \geq 2$,

$$\mathbf{E}[|X| \mid \mathcal{E}] = \frac{1}{\Pr[\mathcal{E}]}\int_0^M \frac{2}{\pi}\frac{x}{1+x^2} = \frac{1}{\Pr[\mathcal{E}]}\frac{1}{\pi}\log(1+M^2) \leq C\log M,$$

where the final bound uses (21). ■

We will show in Theorem 36 below that a matrix of i.i.d. Cauchy random variables is an ℓ_1-subspace embedding. Interestingly, we will use the existence of a well-conditioned basis in the proof, though we will not need an algorithm for constructing it. This lets us use well-conditioned bases with slightly better parameters. In particular, we will use the following Auerbach basis.

Definition 34 *(see "Connection to Auerbach bases" in Section 3.1 of [32]) There exists a $(d, 1, 1)$-well-conditioned basis.*

For readability, it is useful to separate out the following key lemma that is used in Theorem 36 below. This analysis largely follows that in [108].

Lemma 35 *(Fixed Sum of Dilations) Let $\mathbf{S}$ be an $r \times n$ matrix of i.i.d. Cauchy random variables, and let $\mathbf{y}_1, \ldots, \mathbf{y}_d$ be d arbitrary vectors in $\mathbb{R}^n$. Then*

$$\Pr[\sum_{i=1}^d \|\mathbf{S}\mathbf{y}_i\|_1 \geq Cr\log(rd)\sum_{i=1}^d \|\mathbf{y}_i\|_1] \leq \frac{1}{100},$$

where $C > 0$ is an absolute constant.

Proof: Let the rows of $\mathbf{S}$ be denoted $\mathbf{S}_{1*}, \ldots, \mathbf{S}_{r*}$. For $i = 1, \ldots, r$, let $\mathcal{F}_i$ be the event that

$$\forall j \in [d],\ |\langle \mathbf{S}_{i*}, \mathbf{y}_j\rangle| \leq C'rd\|\mathbf{y}_j\|_1,$$

where C' is a sufficiently large positive constant. Note that by the 1-stability of the Cauchy distribution, $\langle \mathbf{S}_{i*}, \mathbf{y}_j\rangle$ is distributed as $\|\mathbf{y}_j\|_1$ times a Cauchy random variable. By Lemma 33 applied to $M = C'rd$, together with a union bound, we have

$$\Pr[\mathcal{F}_i] \geq 1 - d\cdot\frac{2}{\pi C'rd} \geq 1 - \frac{2}{\pi C'r}.$$

Letting $\mathcal{F} = \wedge_{i=1}^{r} \mathcal{F}_i$, we have by another union bound that

$$\Pr[\mathcal{F}] \geq 1 - \frac{2r}{\pi C' r} = 1 - \frac{2}{\pi C'}.$$

Given $\mathcal{F}$, we would then like to appeal to Lemma 33 to bound the expectations $\mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}]$. The issue is that the expectation bound in Lemma 33 cannot be applied, since the condition $\mathcal{F}$ additionally conditions $\mathbf{S}_{i*}$ through the remaining columns $\mathbf{A}_{*j'}$ for $j' \neq j$. A first observation is that by independence, we have

$$\mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}] = \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_i].$$

We also know from Lemma 33 that if $\mathcal{F}_{i,j}$ is the event that $|\langle \mathbf{S}_{i*} \mathbf{y}_j \rangle| \leq C' r d \|\mathbf{y}_j\|_1$, then $\mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_{i,j}] \leq C \log(C' r d) \|\mathbf{y}_j\|_1$, where C is the constant of that lemma.

We can perform the following manipulation (for an event $\mathcal{A}$, we use the notation $\neg \mathcal{A}$ to denote the occurrence of the complement of $\mathcal{A}$):

$$\begin{aligned} C \log(C' r d) \|\mathbf{y}_j\|_1 &= \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_{i,j}] \\ &= \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_i] \cdot \Pr[\mathcal{F}_i \mid \mathcal{F}_{i,j}] \\ &+ \mathbf{E}[\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_{i,j} \wedge \neg \mathcal{F}_i] \cdot \Pr[\neg \mathcal{F}_i \mid \mathcal{F}_{i,j}] \\ &\geq \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_i] \cdot \Pr[\mathcal{F}_i \mid \mathcal{F}_{i,j}]. \end{aligned}$$

We also have

$$\Pr[\mathcal{F}_i \mid \mathcal{F}_{i,j}] \cdot \Pr[\mathcal{F}_{i,j}] = \Pr[\mathcal{F}_i] \geq 1 - O(1/(C'r)),$$

and $\Pr[\mathcal{F}_{i,j}] \geq 1 - O(1/(C' r d))$. Combining these two, we have

$$\Pr[\mathcal{F}_i \mid \mathcal{F}_{i,j}] \geq \frac{1}{2}, \tag{22}$$

for $C' > 0$ a sufficiently large constant. Plugging (22) into the above,

$$C \log(C' r d) \|\mathbf{y}_j\|_1 \geq \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_i] \cdot \Pr[\mathcal{F}_i \mid \mathcal{F}_j] \cdot \frac{1}{2},$$

or equivalently,

$$\mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}] = \mathbf{E}[|\langle \mathbf{S}_{i*}, \mathbf{y}_j \rangle| \mid \mathcal{F}_i] \leq C \log(C' r d) \|\mathbf{y}_j\|_1, \tag{23}$$

as desired.

We thus have, combining (23) with Markov's inequality,

$$\begin{aligned}
& \Pr[\sum_{j=1}^{d} \|\mathbf{S}\mathbf{y}_j\|_1 \geq rC' \log(C'rd) \sum_{ji=1}^{d} \|\mathbf{y}_j\|_1] \\
\leq & \Pr[\neg\mathcal{F}] + \Pr[\sum_{j=1}^{d} \|\mathbf{S}\mathbf{y}_j\|_1 \geq rC' \log(C'rd) \sum_{j=1}^{d} \|\mathbf{y}_j\|_1 \mid \mathcal{F}] \\
\leq & \frac{2}{\pi C'} + \frac{\mathbf{E}[\sum_{j=1}^{d} \|\mathbf{S}\mathbf{y}_j\|_1 \mid \mathcal{F}]}{rC' \log(C'rd) \sum_{j=1}^{d} \|\mathbf{y}_j\|_1} \\
= & \frac{2}{\pi C'} + \frac{\sum_{i=1}^{r} \sum_{j=1}^{d} \mathbf{E}[|\langle \mathbf{S}_{i*}\mathbf{y}_j \rangle| \mid \mathcal{F}]}{rC' \log(C'rd) \sum_{j=1}^{d} \|\mathbf{y}_j\|_1} \\
\leq & \frac{2}{\pi C'} + \frac{rC \log(C'rd)}{rC' \log(C'rd)} \\
\leq & \frac{2}{\pi C'} + \frac{C}{C'}.
\end{aligned}$$

As C' can be chosen sufficiently large, while C is the fixed constant of Lemma 33, we have that

$$\Pr[\sum_{j=1}^{d} \|\mathbf{S}\mathbf{y}_j\|_1 \geq rC' \log(C'rd) \sum_{ji=1}^{d} \|\mathbf{y}_j\|_1] \leq \frac{1}{100}.$$

The lemma now follows by appropriately setting the constant C in the lemma statement. ■

Theorem 36 *A matrix* $\mathbf{S}$ *of i.i.d. Cauchy random variables with* $r = O(d \log d)$ *rows is an* ℓ_1*-subspace embedding with constant probability, that is, with probability at least* $9/10$ *simultaneously for all* x*,*

$$\|\mathbf{A}\mathbf{x}\|_1 \leq 4\|\mathbf{S}\mathbf{A}\mathbf{x}\|_1/r = O(d \log d)\|\mathbf{A}\mathbf{x}\|_1.$$

Proof: Since we will show that with probability $9/10$, for all $\mathbf{x}$ we have $\|\mathbf{A}\mathbf{x}\|_1 \leq 3\|\mathbf{S}\mathbf{A}\mathbf{x}\|_1/r \leq Cd \log d\|\mathbf{A}\mathbf{x}\|_1$, we are free to choose whichever basis of the column space of $\mathbf{A}$ that we like. In particular, we can assume the d columns $\mathbf{A}_{*1}, \ldots, \mathbf{A}_{*d}$ of $\mathbf{A}$ form an Auerbach basis. We will first bound the dilation, and then bound the contraction.

Dilation: We apply Lemma 35 with $\mathbf{y}^i = \mathbf{A}_{*i}$ for $i = 1, 2, \ldots, d$. We have with probability at least 99/100,

$$\sum_{j=1}^{d} \|\mathbf{S}\mathbf{y}_j\|_1 \leq rC\log(rd) \sum_{j=1}^{d} \|\mathbf{y}_j\|_1 = rCd\log(rd), \tag{24}$$

where the last equality used that $\mathbf{y}^1, \ldots, \mathbf{y}^d$ is an Auerbach basis.

Now let $\mathbf{y} = \mathbf{A}\mathbf{x}$ be an arbitrary vector in the column space of $\mathbf{A}$. Then,

$$\begin{aligned}
\|\mathbf{S}\mathbf{y}\|_1 &= \sum_{j=1}^{d} \|\mathbf{S}\mathbf{A}_{*j} \cdot \mathbf{x}_j\|_1 \\
&\leq \sum_{j=1}^{d} \|\mathbf{S}\mathbf{A}_{*j}\|_1 \cdot |\mathbf{x}_j| \\
&\leq \|\mathbf{x}\|_\infty \sum_{j=1}^{d} \|\mathbf{S}\mathbf{A}_{*j}\|_1 \\
&\leq \|\mathbf{x}\|_\infty rCd\log(rd) \\
&\leq \|\mathbf{A}\mathbf{x}\|_1 rCd\log(rd),
\end{aligned}$$

where the third inequality uses (24) and the fourth inequality uses a property of $\mathbf{A}$ being a $(d, 1, 1)$-well-conditioned basis. It follows that $4\|\mathbf{S}\mathbf{A}\mathbf{x}\|_1/r \leq 4Cd\log(rd)\|\mathbf{A}\mathbf{x}\|_1$, as needed in the statement of the theorem.

Contraction: We now argue that no vector's norm shrinks by more than a constant factor. Let $\mathbf{y} = \mathbf{A}\mathbf{x}$ be an arbitrary vector in the column space of $\mathbf{A}$. By the 1-stability of the Cauchy distribution, each entry of $\mathbf{S}\mathbf{y}$ is distributed as a Cauchy scaled by $\|\mathbf{y}\|_1$.

Since the median of the distribution of the absolute value of a Cauchy random variable is 1, we have that with probability at least 1/2, $|\langle \mathbf{S}_i\mathbf{y}\rangle| \geq \|\mathbf{y}\|_1$. Since the entries of $\mathbf{S}\mathbf{y}$ are independent, it follows by a Chernoff bound that the probability that fewer than a 1/3 fraction of the entries are smaller than $\|\mathbf{y}\|_1$ is at most $\exp(-r)$. Hence, with probability $1 - \exp(-r)$, $\|\mathbf{S}\mathbf{y}\|_1$ is at least $r\|\mathbf{y}\|_1/3$, or equivalently, $4\|\mathbf{S}\mathbf{A}\mathbf{x}\|_1/r \geq (4/3)\|\mathbf{A}\mathbf{x}\|_1$.

We now use a net argument as in [108]. By Lemma 29, there exists an ε-net $\mathcal{N} \subset \{\mathbf{A}\mathbf{x} \mid \|\mathbf{A}\mathbf{x}\|_1 = 1\}$ for which $|\mathcal{N}| \leq (24Cd\log(rd))^d$ and for any $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $\|\mathbf{y}\|_1 = 1$, there exists a $\mathbf{w} \in \mathcal{N}$ with $\|\mathbf{y} - \mathbf{w}\|_1 \leq \frac{1}{12Cd\log(rd)}$. Observe that for a sufficiently large $r = O(d\log d)$ number of rows of $\mathbf{S}$, we have by a union bound, that with probability $1 - \exp(-r)|\mathcal{N}| \geq 99/100$, simultaneously for all $\mathbf{z} \in \mathcal{N}$, $4\|\mathbf{S}\mathbf{w}\|_1/r \geq (4/3)\|\mathbf{w}\|_1$.

For an arbitrary $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $\|\mathbf{y}\|_1 = 1$, we can write $\mathbf{y} = \mathbf{w} + (\mathbf{y} - \mathbf{w})$ for a $\mathbf{w} \in \mathcal{N}$ and $\|\mathbf{y} - \mathbf{w}\|_1 \leq \frac{1}{12Cd\log(rd)}$. By the triangle inequality,

$$\begin{aligned}\frac{4\|\mathbf{S}\mathbf{y}\|_1}{r} &\geq \frac{4\|\mathbf{S}\mathbf{w}\|_1}{r} - \frac{4\|\mathbf{S}(\mathbf{y}-\mathbf{w})\|_1}{r} \\ &\geq \frac{4}{3}\|\mathbf{w}\|_1 - \frac{4\|\mathbf{S}(\mathbf{y}-\mathbf{w})\|_1}{r} \\ &= \frac{4}{3} - \frac{4\|\mathbf{S}(\mathbf{y}-\mathbf{w})\|_1}{r}.\end{aligned}$$

Since we have already shown that $4\|\mathbf{S}\mathbf{A}\mathbf{x}\|_1/r \leq 4Cd\log(rd)\|\mathbf{A}\mathbf{x}\|_1$ for all $\mathbf{x}$, it follows that

$$\frac{4\|\mathbf{S}(\mathbf{y}-\mathbf{w})\|_1}{r} \leq 4Cd\log(rd)\|\mathbf{y}-\mathbf{w}\|_1 \leq \frac{4Cd\log(rd)}{12Cd\log(rd)} \leq \frac{1}{3}.$$

It follows now that $4\|\mathbf{S}\mathbf{y}\|_1/r \geq 1 = \|\mathbf{y}\|_1$ for all vectors $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $\|\mathbf{y}\|_1 = 1$.

Hence, the statement of the theorem holds with probability at least $9/10$, by a union bound over the events in the dilation and contraction arguments. This concludes the proof. ■

Corollary 37 *There is an $O(nd^2 + nd\log(d\varepsilon^{-1}\log n)) + \mathrm{poly}(d/\varepsilon)$ time algorithm for solving the ℓ_1-regression problem up to a factor of $(1+\varepsilon)$ and with error probability $1/10$.*

Proof: The corollary follows by combining Theorem 30, Lemma 32 and its optimization in §3.3, and Theorem 36. Indeed, we can compute $\mathbf{S}\cdot\mathbf{A}$ in $O(nd^2)$ time, then a QR-factorization as well as $\mathbf{R}^{-1}$ in $\mathrm{poly}(d)$ time. Then we can compute $\mathbf{A}\mathbf{R}^{-1}\mathbf{G}$ as well as perform the sampling in Theorem 30 in $nd\log(d\varepsilon^{-1}\log n)$ time. Finally, we can solve the ℓ_1-regression problem on the samples in $\mathrm{poly}(d/\varepsilon)$ time. ■

While the time complexity of Corollary 37 can be improved to roughly $O(nd^{1.376}) + \mathrm{poly}(d/\varepsilon)$ using algorithms for fast matrix multiplication, there are better ways of speeding this up, as we shall see in the next section.

3.5 Subspace embeddings using exponential random variables

We now describe a speedup over the previous section using exponential random variables, as in [124]. Other speedups are possible, using [26, 27, 92],

though the results in [124] additionally also slightly improve the sampling complexity. The use of exponential random variables in [124] is inspired by an elegant work of Andoni, Onak, and Krauthgamer on frequency moments [5, 4].

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

Property 1 *If $U_1, \ldots, U_n$ are exponentially distributed, and $\alpha_i > 0$ $(i = 1, \ldots, n)$ are real numbers, then $\max\{\alpha_1/U_1, \ldots, \alpha_n/U_n\} \simeq \left(\sum_{i \in [n]} \alpha_i\right)/U$, where U is exponential.*

The following lemma shows a relationship between the Cauchy distribution and the exponential distribution.

Lemma 38 *Let $y_1, \ldots, y_d \geq 0$ be scalars. Let $U_1, \ldots, U_d$ be d independent exponential random variables, and let $X = (\sum_{i \in [d]} y_i^2/U_i^2)^{1/2}$. Let $C_1, \ldots, C_d$ be d independent Cauchy random variables, and let $Y = (\sum_{i \in [d]} y_i^2 C_i^2)^{1/2}$. There is a constant $\gamma > 0$ for which for any $t > 0$.*

$$\Pr[X \geq t] \leq \Pr[Y \geq \gamma t].$$

Proof: We would like the density function h of $y_i^2 C_i^2$. Letting $t = y_i^2 C_i^2$, the inverse function is $C_i = t^{1/2}/y_i$. Taking the derivative, we have $\frac{dC_i}{dt} = \frac{1}{2y_i} t^{-1/2}$. Letting $f(t) = \frac{2}{\pi} \frac{1}{1+t^2}$ be the density function of the absolute value of a Cauchy random variable, we have by the change of variable technique,

$$h(t) \;=\; \frac{2}{\pi} \frac{1}{1 + t/y_i^2} \cdot \frac{1}{2y_i} t^{-1/2} = \frac{1}{\pi} \frac{1}{y_i t^{1/2} + t^{3/2}/y_i}.$$

We would also like the density function k of $y_i^2 E_i^2$, where $E_i \sim 1/U_i$. Letting $t = y_i^2 E_i^2$, the inverse function is $E_i = t^{1/2}/y_i$. Taking the derivative, $\frac{dE_i}{dt} = \frac{1}{2y_i} t^{-1/2}$. Letting $g(t) = t^{-2} e^{-1/t}$ be the density function of the reciprocal of an exponential random variable, we have by the change of variable technique,

$$k(t) \;=\; \frac{y_i^2}{t} e^{-y_i/t^{1/2}} \cdot \frac{1}{2y_i} t^{-1/2} = \frac{y_i}{2t^{3/2}} e^{-y_i/t^{1/2}}.$$

We claim that $k(t) \leq h(\gamma t)/\gamma$ for a sufficiently small constant $\gamma > 0$. This is equivalent to showing that

$$\frac{\pi}{2} \frac{y_i}{t^{3/2}} e^{-y_i/t^{1/2}} \gamma \leq \frac{1}{\gamma^{1/2} y_i t^{1/2} + \gamma^{3/2} t^{3/2}/y_i},$$

which for $\gamma < 1$, is implied by showing that

$$\frac{\pi}{2}\frac{y_i}{t^{3/2}}e^{-y_i/t^{1/2}}\gamma \leq \frac{1}{\gamma^{1/2}y_i t^{1/2} + \gamma^{1/2}t^{3/2}/y_i}.$$

We distinguish two cases: first suppose $t \geq y_i^2$. In this case, $e^{-y_i/t^{1/2}} \leq 1$. Note also that $y_i t^{1/2} \leq t^{3/2}/y_i$ in this case. Hence, $\gamma^{1/2}y_i t^{1/2} \leq \gamma^{1/2}t^{3/2}/y_i$. Therefore, the above is implied by showing

$$\frac{\pi}{2}\frac{y_i}{t^{3/2}}\gamma \leq \frac{y_i}{2\gamma^{1/2}t^{3/2}},$$

or

$$\gamma^{3/2} \leq \frac{1}{\pi},$$

which holds for a sufficiently small constant $\gamma \in (0,1)$.

Next suppose $t < y_i^2$. In this case $y_i t^{1/2} > t^{3/2}/y_i$, and it suffices to show

$$\frac{\pi}{2}\frac{y_i}{t^{3/2}}e^{-y_i/t^{1/2}}\gamma \leq \frac{1}{2\gamma^{1/2}y_i t^{1/2}},$$

or equivalently,

$$\pi y_i^2 \gamma^{3/2} \leq te^{y_i/t^{1/2}}.$$

Using that $e^x \geq x^2/2$ for $x \geq 0$, it suffices to show

$$\pi y_i^2 \gamma^{3/2} \leq y_i^2/2,$$

which holds for a small enough $\gamma \in (0,1)$.

We thus have,

$$\begin{aligned}
\Pr[X \geq t] &= \Pr[X^2 \geq t^2] \\
&= \Pr[\sum_{i=1}^{d} y_i^2/U_i^2 \geq t^2] \\
&= \int_{\sum_{i=1}^{d} t_i \geq t^2} k(t_1)\cdots k(t_d)dt_1\cdots dt_d \\
&\leq \int_{\sum_{i=1}^{d} t_i \geq t^2} \kappa^{-d}h(\kappa t_1)\cdots h(\kappa t_d)dt_1\cdots dt_d \\
&\leq \int_{\sum_{i=1}^{d} s_i \geq \kappa t^2} f(s_1)\cdots f(s_d)ds_1\cdots ds_d \\
&= \Pr[Y^2 \geq \kappa t^2] \\
&= \Pr[Y \geq \kappa^{1/2}t],
\end{aligned}$$

where we made the change of variables $s_i = \kappa t_i$. Setting $\gamma = \kappa^{1/2}$ completes the proof. ∎

We need a bound on $\Pr[Y \geq t]$, where $Y = (\sum_{i \in [d]} y_i^2 C_i^2)^{1/2}$ is as in Lemma 38.

Lemma 39 *There is a constant $c > 0$ so that for any $r > 0$,*

$$\Pr[Y \geq r\|y\|_1] \leq \frac{c}{r}.$$

Proof: For $i \in [d]$, let $\sigma_i \in \{-1, +1\}$ be i.i.d. random variables with $\Pr[\sigma_i = -1] = \Pr[\sigma_i = 1] = 1/2$. Let $Z = \sum_{i \in [d]} \sigma_i y_i C_i$. We will obtain tail bounds for Z in two different ways, and use this to establish the lemma.

On the one hand, by the 1-stability of the Cauchy distribution, we have that $Z \sim \|y\|_1 C$, where C is a standard Cauchy random variable. Note that this holds for any fixing of the σ_i. The cumulative distribution function of the Cauchy random variable is $F(z) = \frac{2}{\pi} \arctan(z)$. Hence for any $r > 0$,

$$\Pr[Z \geq r\|y\|_1] \quad = \quad \Pr[C \geq r] = 1 - \frac{2}{\pi} \arctan(r).$$

We can use the identity

$$\arctan(r) + \arctan\left(\frac{1}{r}\right) = \frac{\pi}{2},$$

and therefore using the Taylor series for arctan for $r > 1$,

$$\arctan(r) \geq \frac{\pi}{2} - \frac{1}{r}.$$

Hence,

$$\Pr[Z \geq r\|y\|_1] \leq \frac{2}{\pi r}. \tag{25}$$

On the other hand, for any fixing of $C_1, \ldots, C_d$, we have

$$\mathbf{E}[Z^2] = \sum_{i \in [d]} y_i^2 C_i^2,$$

and also

$$\mathbf{E}[Z^4] = 3 \sum_{i \neq j \in [d]} y_i^2 y_j^2 C_i^2 C_j^2 + \sum_{i \in [d]} y_i^4 C_i^4.$$

We recall the Paley-Zygmund inequality.

Fact 3 *If $R \geq 0$ is a random variable with finite variance, and $0 < \theta < 1$, then*

$$\Pr[R \geq \theta \mathbf{E}[R]] \geq (1-\theta)^2 \cdot \frac{\mathbf{E}[R]^2}{\mathbf{E}[R^2]}.$$

Applying this inequality with $R = Z^2$ and $\theta = 1/2$, we have

$$\begin{aligned} \Pr[Z^2 \geq \frac{1}{2} \cdot \sum_{i \in [d]} y_i^2 C_i^2] &\geq \frac{1}{4} \cdot \frac{\left(\sum_{i \in [d]} y_i^2 C_i^2\right)^2}{3 \sum_{i \neq j \in [d]} y_i^2 y_j^2 C_i^2 C_j^2 + \sum_{i \in [d]} y_i^4 C_i^4} \\ &\geq \frac{1}{12}, \end{aligned}$$

or equivalently

$$\Pr[Z \geq \frac{1}{\sqrt{2}} (\sum_{i \in [d]} y_i^2 C_i^2)^{1/2}] \geq \frac{1}{12}. \tag{26}$$

Suppose, towards a contradiction, that $\Pr[Y \geq r\|y\|_1] \geq c/r$ for a sufficiently large constant $c > 0$. By independence of the σ_i and the C_i, by (26) this implies

$$\Pr[Z \geq \frac{r\|y\|_1}{\sqrt{2}}] \geq \frac{c}{12r}.$$

By (25), this is a contradiction for $c > \frac{24}{\pi}$. It follows that $\Pr[Y \geq r\|y\|_1] < c/r$, as desired. ■

Corollary 40 *Let $y_1, \ldots, y_d \geq 0$ be scalars. Let $U_1, \ldots, U_d$ be d independent exponential random variables, and let $X = (\sum_{i \in [d]} y_i^2/U_i^2)^{1/2}$. There is a constant $c > 0$ for which for any $r > 0$,*

$$\Pr[X > r\|y\|_1] \leq c/r.$$

Proof: The corollary follows by combining Lemma 38 with Lemma 39, and rescaling the constant c from Lemma 39 by $1/\gamma$, where γ is the constant of Lemma 38. ■

Theorem 41 *([124]) Let $\mathbf{S}$ be an $r \times n$ CountSketch matrix with $r = d \cdot \operatorname{poly}\log d$, and $\mathbf{D}$ an $n \times n$ diagonal matrix with i.i.d. entries $\mathbf{D}_{i,i}$ distributed as a reciprocal of a standard exponential random variable. Then, with probability at least $9/10$ simultaneously for all x,*

$$\Omega\left(\frac{1}{d \log^{3/2} d}\right) \|\mathbf{Ax}\|_1 \leq \|\mathbf{SDAx}\|_1 \leq O(d \log d)\|\mathbf{Ax}\|_1.$$

Proof: By Theorem 10, with probability at least 99/100 over the choice of $\mathbf{S}$, $\mathbf{S}$ is an ℓ_2-subspace embedding for the matrix $\mathbf{D}\cdot\mathbf{A}$, that is, simultaneously for all $x \in \mathbb{R}^d$, $\|\mathbf{SDAx}\|_2 = (1\pm 1/2)\|\mathbf{DAx}\|_2$. We condition $\mathbf{S}$ on this event.

For the dilation, we need Khintchine's inequality.

Fact 4 *([58]). Let $Z = \sum_{i=1}^r \sigma_i z_i$ for i.i.d. random variables σ_i uniform in $\{-1, +1\}$, and $z_1, \ldots, z_r$ be scalars. There exists a constant $c > 0$ for which for all $t > 0$*

$$\Pr[|Z| > t\|\mathbf{y}\|_2] \leq \exp(-ct^2).$$

Let $\mathbf{y}^1, \ldots, \mathbf{y}^d$ be d vectors in an Auerbach basis for the column space of $\mathbf{A}$. Applying Fact 4 to a fixed entry j of $\mathbf{SDy}^i$ for a fixed i, and letting $\mathbf{z}^{i,j}$ denote the vector whose k-th coordinate is $\mathbf{y}^i_k$ if $\mathbf{S}_{j,k} \neq 0$, and otherwise $\mathbf{z}^{i,j}_k = 0$, we have for a constant $c' > 0$,

$$\Pr[|(\mathbf{SDy}^i)_j| > c'\sqrt{\log d}\|\mathbf{Dz}^{i,j}\|_2] \leq \frac{1}{d^3}.$$

By a union bound, with probability

$$1 - \frac{rd}{d^3} = 1 - \frac{d^2 \mathrm{poly}\log d}{d^3} = 1 - \frac{\mathrm{poly}\log d}{d},$$

for all i and j,

$$|(\mathbf{SDy}^i)_j| \leq c'\sqrt{\log d}\|\mathbf{Dz}^{i,j}\|_2,$$

which we denote by event $\mathcal{E}$ and condition on. Notice that the probability is taken only over the choice of the σ_i, and therefore conditions only the σ_i random variables.

In the following, $i \in [d]$ and $j \in [r]$. Let $\mathcal{F}_{i,j}$ be the event that

$$\|\mathbf{Dz}^{i,j}\|_2 \leq 100dr\|\mathbf{z}^{i,j}\|_1,$$

We also define

$$\mathcal{F}_j = \wedge_i \mathcal{F}_{i,j}, \quad \mathcal{F} = \wedge_j \mathcal{F}_j = \wedge_{i,j}\mathcal{F}_{i,j}.$$

By Corollary 40 and union bounds,

$$\Pr[\mathcal{F}_j] \geq 1 - \frac{1}{100r},$$

and union-bounding over $j \in [r]$,

$$\Pr[\mathcal{F}] \geq \frac{99}{100}.$$

We now bound $\mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}]$. By independence,

$$\mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}] = \mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_j].$$

Letting $p = \Pr[\mathcal{E} \wedge \mathcal{F}_{i,j}] \geq 99/100$, we have by Corollary 40,

$$\begin{aligned}
\mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_{i,j}] &= \int_{u=0}^{100dr} \Pr[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \geq u\|\mathbf{z}^{i,j}\|_1 \mid \mathcal{E}, \mathcal{F}_{i,j}] \cdot du \\
&\leq \frac{1}{p}(1 + \int_{u=1}^{100dr} \frac{c}{u}) \cdot du \\
&\leq \frac{c}{p}(1 + \ln(100dr)).
\end{aligned}$$

We can perform the following manipulation:

$$\begin{aligned}
\frac{c}{p}(1 + \ln(100dr)) &\geq \mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_{i,j}] \\
&\geq \mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_j] \cdot \Pr[\mathcal{F}_j \mid \mathcal{F}_{i,j}] \\
&= \mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_j] \cdot \Pr[\mathcal{F}_j] / \Pr[\mathcal{F}_{i,j}] \\
&\geq \frac{1}{2}\mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_j] \cdot \Pr[\mathcal{F}_j] \\
&= \frac{1}{2}\mathbf{E}[\|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}_j] \cdot \Pr[\mathcal{F}].
\end{aligned}$$

It follows by linearity of expectation that,

$$\mathbf{E}[\sum_{i \in [d], j \in [d\,\mathrm{poly}\log d]} \|\mathbf{D}\mathbf{z}^{i,j}\|_2 \mid \mathcal{E}, \mathcal{F}] \leq \frac{c}{p}(1 + \ln(100dr)) \sum_{i=1}^{d} \|\mathbf{y}^i\|_1.$$

Consequently, by a Markov bound, and using that $p \geq 1/2$, conditioned on $\mathcal{E} \wedge \mathcal{F}$, with probability at least $9/10$, we have the occurrence of the event $\mathcal{G}$ that

$$\sum_{i=1}^{d} \|\mathbf{D}\mathbf{y}^i\|_2 \leq 10\frac{c}{p}(1 + \ln(100dr)) \sum_{i=1}^{d} \|\mathbf{y}^i\|_1. \leq 40cd\ln(100dr) \tag{27}$$

To bound the dilation, consider a fixed vector $\mathbf{x} \in \mathbb{R}^d$. Then conditioned on $\mathcal{E} \wedge \mathcal{F} \wedge \mathcal{G}$, and for $\mathbf{A} = [\mathbf{y}^1, \ldots, \mathbf{y}^d]$ an Auerbach basis (without loss of

generality),

$$\begin{aligned}\|\mathbf{SDAx}\|_1 &\leq \|\mathbf{x}\|_\infty \sum_{i=1}^{d} \|\mathbf{SDy}^i\|_1 \\ &\leq \|\mathbf{Ax}\|_1 \sum_{i=1}^{d} \|\mathbf{SDy}^i\|_1 \\ &\leq \|\mathbf{Ax}\|_1 c' \sqrt{\log d} 40cd \ln(100dr) \\ &\leq c'' d(\log^{3/2} d)\|\mathbf{Ax}\|_1,\end{aligned}$$

where the first inequality follows from the triangle inequality, the second inequality uses that $\|\mathbf{x}\|_\infty \leq \|\mathbf{Ax}\|_1$ for a well-conditioned basis $\mathbf{A}$, the third inequality uses (27), and in the fourth inequality $c'' > 0$ is a sufficiently large constant. Thus for all $\mathbf{x} \in \mathbb{R}^d$,

$$\|\mathbf{SDAx}\|_1 \leq c'' d(\log^{3/2} d)\|\mathbf{Ax}\|_1. \tag{28}$$

For the contraction, we have

$$\begin{aligned}\|\mathbf{SDAx}\|_1 &\geq \|\mathbf{SDAx}\|_2 \\ &\geq \frac{1}{2}\|\mathbf{DAx}\|_2 \\ &\geq \frac{1}{2}\|\mathbf{DAx}\|_\infty \\ &= \frac{1}{2}\frac{\|\mathbf{Ax}\|_1}{U},\end{aligned}$$

where U is a standard exponential random variables, and where the first inequality uses our conditioning on $\mathbf{S}$, the second inequality uses a standard norm inequality, and the third inequality uses the max-stability of the exponential distribution. Thus, since the cumulative distribution of an exponential random variable $F(x) = 1 - e^{-x}$, we have that for any fixed x,

$$\Pr\left[\|\mathbf{SDAx}\|_1 \geq \frac{1}{4}\frac{\|\mathbf{Ax}\|_1}{d\log(2d^3)}\right] \geq 1 - (2d^2)^{2d}. \tag{29}$$

By Lemma 29, there exists a $\frac{1}{d^3}$-net $\mathcal{N}$ for which $|\mathcal{N}| \leq (2d^3)^d$, where $\mathcal{N}$ is a subset of $\{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = \mathbf{Ax} \text{ for some } \mathbf{x} \in \mathbb{R}^d \text{ and } \|\mathbf{y}\|_1 = 1\}$. Combining this with (29), by a union bound we have the event $\mathcal{E}$ that simultaneously for all $\mathbf{w} \in \mathcal{N}$,

$$\|\mathbf{SDw}\|_1 \geq \frac{1}{4}\frac{\|\mathbf{w}\|_1}{d\log(2d^3)}.$$

Now consider an arbitrary vector $\mathbf{y}$ of the form $\mathbf{SDAx}$ with $\|\mathbf{y}\|_1 = 1$. By definition of $\mathcal{N}$, one can write $\mathbf{y} = \mathbf{w} + (\mathbf{y} - \mathbf{w})$, where $\mathbf{w} \in \mathcal{N}$ and $\|\mathbf{y} - \mathbf{w}\|_1 \leq \frac{1}{d^3}$. We have,

$$\begin{aligned}
\|\mathbf{SDy}\|_1 &\geq \|\mathbf{SDw}\|_1 - \|\mathbf{SD}(\mathbf{y} - \mathbf{w})\|_1 \\
&\geq \frac{1}{4d\log(2d^3)} - \|\mathbf{SD}(\mathbf{y} - \mathbf{w})\|_1 \\
&\geq \frac{1}{4d\log(2d^3)} - \frac{O(d\log^{3/2} d)}{d^3} \\
&\geq \frac{1}{8d\log(2d^3)},
\end{aligned}$$

where the first inequality uses the triangle inequality, the second the occurrence of $\mathcal{E}$, and the third (28). This completes the proof. ■

Corollary 42 *There is an $O(\mathrm{nnz}(\mathbf{A})\log n) + \mathrm{poly}(d/\varepsilon)$ time algorithm for computing $\mathbf{\Pi A}$, where $\mathbf{\Pi}$ is a $\mathrm{poly}(d/\varepsilon)$ by n matrix satisfying, with probability at least $9/10$, $\|\mathbf{\Pi Ax}\|_1 = (1 \pm \varepsilon)\|\mathbf{Ax}\|_1$ for all x. Therefore, there is also an $O(\mathrm{nnz}(\mathbf{A})\log n) + \mathrm{poly}(d/\varepsilon)$ time algorithm for solving the ℓ_1-regression problem up to a factor of $(1+\varepsilon)$ with error probability $1/10$.*

Proof: The corollary follows by combining Theorem 30, Lemma 32 and its optimization in §3.3, and Theorem 41. ■

3.6 Application to hyperplane fitting

One application of ℓ_1-regression is to finding the best hyperplane to find a set of n points in $\mathbb{R}^d$, presented as an $n \times d$ matrix $\mathbf{A}$ [22, 23, 70, 71, 108, 26]. One seeks to find a hyperplane H so that the sum of ℓ_1-distances of the rows $\mathbf{A}_{i*}$ to H is as small as possible.

While in general, the points on H are those $\mathbf{x} \in \mathbb{R}^d$ for which $\langle \mathbf{x}, \mathbf{w} \rangle = \gamma$, where $\mathbf{w}$ is the normal vector of H and $\gamma \in \mathbb{R}$, we can in fact assume that $\gamma = 0$. Indeed, this follows by increasing the dimension d by one, placing the value 1 on all input points in the new coordinate, and placing the value γ on the new coordinate in $\mathbf{w}$. As this will negligibly affect our overall time complexity, we can therefore assume $\gamma = 0$ in what follows, that is, H contains the origin.

A nice feature of the ℓ_1-norm is that if one grows an ℓ_1-ball around a point $\mathbf{x} \in \mathbb{R}^d$, it first touches a hyperplane H at a vertex of the ℓ_1-ball. Hence, there is a coordinate direction $i \in [d]$ for which the point of closest

ℓ_1-distance to $\mathbf{x}$ on H is obtained by replacing the i-th coordinate of $\mathbf{x}$ by the unique real number v so that $(x_1, \ldots, x_{i-1}, v, x_{i+1}, \ldots, x_d)$ is on H.

An interesting observation is that this coordinate direction i only depends on H, that is, it is independent of $\mathbf{x}$, as shown in Corollary 2.3 of [88]. Let $\mathbf{A}^{-j}$ denote the matrix $\mathbf{A}$ with its j-th column removed. Consider a hyperplane H with normal vector $\mathbf{w}$. Let $\mathbf{w}^{-j}$ denote the vector obtained by removing its j-th coordinate. Then the sum of ℓ_1-distances of the rows $\mathbf{A}_{i*}$ of $\mathbf{A}$ to H is given by

$$\min_j \| - \mathbf{A}^{-j}\mathbf{w}^{-j} - \mathbf{A}_{*j}\|_1,$$

since $\mathbf{A}^{-j}\mathbf{w}^{-j}$ is the negation of the vector of j-th coordinates of the points projected (in the ℓ_1 sense) onto H, using that $\langle \mathbf{w}, \mathbf{x} \rangle = 0$ for $\mathbf{x}$ on the hyperplane. It follows that an optimal hyperplane H can be obtained by solving

$$\min_j \min_{\mathbf{w}} \| - \mathbf{A}^{-j}\mathbf{w}^{-j} - \mathbf{A}_{*j}\|_1,$$

which characterizes the normal vector $\mathbf{w}$ of H. Hence, by solving d ℓ_1-regression problems, each up to a $(1+\varepsilon)$-approximation factor and each on an $n \times (d-1)$ matrix, one can find a hyperplane whose cost is at most $(1+\varepsilon)$ times the cost of the optimal hyperplane.

One could solve each of the d ℓ_1-regression problems independently up to $(1+\varepsilon)$-approximation with error probability $1/d$, each taking $O(\mathrm{nnz}(\mathbf{A}) \log n) + \mathrm{poly}(d/\varepsilon)$ time. This would lead to an overall time of $O(\mathrm{nnz}(\mathbf{A}) d \log n) + \mathrm{poly}(d/\varepsilon)$, but we can do better by reusing computation.

That is, it suffices to compute a subspace embedding $\mathbf{\Pi A}$ once, using Corollary 42, which takes only $O(\mathrm{nnz}(\mathbf{A}) \log n) + \mathrm{poly}(\delta/\varepsilon)$ time.

For the subspace approximation problem, we can write

$$\begin{aligned} \min_j \min_{\mathbf{w}} \| - \mathbf{A}^{-j}\mathbf{w}^{-j} - \mathbf{A}_{*j}\|_1 &= \min_j \min_{\mathbf{w}|\mathbf{w}_j=0} \| - \mathbf{A}\mathbf{w} - \mathbf{A}_{*j}\|_1 \\ &= (1 \pm \varepsilon) \min_j \min_{\mathbf{w}|\mathbf{w}_j=0} \| - \mathbf{\Pi A}\mathbf{w} - \mathbf{\Pi A}_{*j}\|_1. \end{aligned}$$

Thus, having computed $\mathbf{\Pi A}$ once, one can solve the subspace approximation problem with an additional $\mathrm{poly}(d/\varepsilon)$ amount of time. We summarize our findings in the following theorem.

Theorem 43 *(ℓ_1-Hyperplane Approximation) There is an $O(\mathrm{nnz}(\mathbf{A}) \log n) + \mathrm{poly}(d/\varepsilon)$ time algorithm for solving the ℓ_1-Hyperplane approximation problem with constant probability.*

4 Low Rank Approximation

In this chapter we study the low rank approximation problem. We are given an $n \times d$ matrix $\mathbf{A}$, and would like to find a matrix $\tilde{\mathbf{A}}_k$ for which

$$\|\mathbf{A} - \tilde{\mathbf{A}}_k\| \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|,$$

where $\mathbf{A}_k$ is the best rank-k approximation to $\mathbf{A}$ with respect to some matrix norm, and $\tilde{\mathbf{A}}_k$ has rank k.

Low rank approximation can be used for a variety of problems, such as Non-Negative Matrix Factorization (NNMF) [75], Latent Dirichlet Allocation (LDA) [16], and face recognition. It has also recently been used for ℓ_2-error shape-fitting problems [50], such as k-means and projective clustering.

Here we demonstrate an application to latent semantic analysis (LSA). We define a *term-document* matrix $\mathbf{A}$ in which the rows correpond to terms (e.g., words) and columns correspond to documents. The entry $\mathbf{A}_{i,j}$ equals the number of occurrences of term i in document j. Two terms i and j can be regarded as correlated if the inner product $\langle \mathbf{A}_{i,*}, \mathbf{A}_{j,*} \rangle$ of their corresponding rows of $\mathbf{A}$ is large. The matrix $\mathbf{A}\mathbf{A}^T$ contains all such inner products. Similarly, one can look at document correlation by looking at $\mathbf{A}^T\mathbf{A}$. By writing $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ in its SVD, we have $\mathbf{A}\mathbf{A}^T = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^T$.

By taking the SVD of low rank approximation $\tilde{\mathbf{A}}_k$ to a matrix $\mathbf{A}$, one obtains $\tilde{\mathbf{A}}_k = \mathbf{L}\mathbf{U}\mathbf{R}^T$, where $\mathbf{L}$ and $\mathbf{R}$ have orthonormal columns, and $\mathbf{U}$ is a rank-k matrix. One can view the columns of $\mathbf{L}$ and $\mathbf{R}$ as approximations to the top k left and right singular vectors of $\mathbf{A}$. Note that, as we will see below, the algorithm for generating $\tilde{\mathbf{A}}_k$ usually generates its factorization into the product of $\mathbf{L}$, $\mathbf{U}$, and $\mathbf{R}^T$ so one does not need to perform an SVD on $\tilde{\mathbf{A}}_k$ (to achieve $O(\text{nnz}(\mathbf{A})) + (n+d)\text{poly}(k/\varepsilon)$ time algorithms for low rank approximation, one cannot actually afford to write down $\tilde{\mathbf{A}}_k$ other than in factored form, since $\tilde{\mathbf{A}}_k$ may be dense).

There are two well-studied norms in this context, the Frobenius and the spectral (operator) norm, both of which have the same minimizer $\mathbf{A}_k$ given by the singular value decomposition of $\mathbf{A}$. That is, if one writes $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ in its SVD, where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal and $\mathbf{\Sigma}$ is a non-negative diagonal matrix with $\mathbf{\Sigma}_{1,1} \geq \mathbf{\Sigma}_{2,2} \geq \cdots \mathbf{\Sigma}_{n,n} \geq 0$, then $\mathbf{A}_k = \mathbf{U}\mathbf{\Sigma}_k\mathbf{V}^T$, where $\mathbf{\Sigma}_k$ agrees with $\mathbf{\Sigma}$ on its top k diagonal entries, but is 0 otherwise. Clearly this is a rank-k matrix, and the Eckart-Young Theorem guarantees that it is the minimizer for any rotationally-invariant norm, which includes the Frobenius and spectral norms. The top k rows of $\mathbf{V}^T$ are known as the top k principal components of $\mathbf{A}$.

We will show how to use sketching to speed up algorithms for both problems, and further variants. Our exposition is based on combinations of several works in this area by Sárlos, Clarkson, and the author [105, 28, 27].

Chapter Overview: In §4.1 we give an algorithm for computing a low rank approximation achieving error proportional to the Frobenius norm. In §4.2 we give a different kind of low rank approximation, called a CUR decomposition, which computes a low rank approximation also achieving Frobenius norm error but in which the column space equals the span of a small subset of columns of the input matrix, while the row space equals the span of a small subset of rows of the input matrix. A priori, it is not even clear why such a low rank approximation should exist, but we show that it not only exists, but can be computed in nearly input sparsity time. We also show that it can be computed deterministically in polynomial time. This algorithm requires several detours into a particular kind of spectral sparsification given in §4.2.1, as well as an adaptive sampling technique given in §4.2.2. Finally in §4.2.3 we show how to put the pieces together to obtain the overall algorithm for CUR factorization. One tool we need is a way to compute the best rank-k approximation of the column space of a matrix when it is restricted to lie within a prescribed subspace; we defer the details of this to §4.4, where the tool is developed in the context of an application called Distributed Low Rank Approximation. In §4.3 we show how to perform low rank approximation with a stronger guarantee, namely, an error with respect to the spectral norm. While the solution quality is much better than in the case of the Frobenius norm, it is unknown how to compute this as quickly, though one can still compute it much more quickly than the SVD. In §4.4 we present the details of the Distributed Low Rank Approximation algorithm.

4.1 Frobenius norm error

We will say a k-dimensional subspace of $\mathbb{R}^d$ spans a $(1+\varepsilon)$ rank-k approximation to $\mathbf{A}$ if

$$\|\mathbf{A} - \mathbf{A}\mathbf{L}\mathbf{L}^T\|_{\mathrm{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}},$$

where $\mathbf{L}\mathbf{L}^T$ is the projection operator onto that subspace. We will sometimes abuse notation and refer to $\mathbf{L}$ as the subspace as well, meaning the k-dimensional subspace of $\mathbb{R}^d$ spanned by the rows of $\mathbf{L}^T$.

One way of interpreting the Frobenius low rank problem is to treat each of the n rows of $\mathbf{A}$ as a point in $\mathbb{R}^d$. A particularly nice property about the Frobenius norm is that if one is given a subspace L of $\mathbb{R}^d$ which is guaranteed to contain a rank-k subspace $L' \subseteq L$ spanning a $(1+\varepsilon)$ rank-k approximation

to $\mathbf{A}$, then it can be found by projecting each of the rows of $\mathbf{A}$ onto $\mathbf{V}$, and then finding the best rank-k approximation to the projected points inside of $\mathbf{V}$. This is a simple, but very useful corollary of the Pythagorean theorem.

Lemma 44 *The best rank-k approximation to $\mathbf{A}$ in Frobenius norm in the row space of a matrix $\mathbf{U}^T$ with orthonormal rows is given by $[\mathbf{AU}]_k\mathbf{U}^T$, where $[\mathbf{AU}]_k$ denotes the best rank-k approximation to $\mathbf{AU}$.*

Proof: Let $\mathbf{Z}$ be an arbitrary matrix of rank k of the same dimensions as $\mathbf{AU}$. Then,

$$\begin{aligned}\|\mathbf{AUU}^T - [\mathbf{AU}]_k\mathbf{U}^T\|_{\mathrm{F}}^2 &= \|\mathbf{AU} - [\mathbf{AU}]_k\|_{\mathrm{F}}^2 \\ &\leq \|\mathbf{AU} - \mathbf{Z}\|_{\mathrm{F}}^2 \\ &= \|\mathbf{AUU}^T - \mathbf{ZU}^T\|_{\mathrm{F}}^2,\end{aligned}$$

where the equalities use that the rows of $\mathbf{U}^T$ are orthonormal, while the inequality uses that $[\mathbf{AU}]_k$ is the best rank-k approximation to $\mathbf{AU}$.

Hence,

$$\begin{aligned}\|\mathbf{A} - [\mathbf{AU}]_k\mathbf{U}^T\|_{\mathrm{F}}^2 &= \|\mathbf{A} - \mathbf{AUU}^T\|_{\mathrm{F}}^2 + \|\mathbf{AUU}^T - [\mathbf{AU}]_k\mathbf{U}^T\|_{\mathrm{F}}^2 \\ &\leq \|\mathbf{A} - \mathbf{AUU}^T\|_{\mathrm{F}}^2 + \|\mathbf{AUU}^T - \mathbf{ZU}^T\|_{\mathrm{F}}^2 \\ &= \|\mathbf{A} - \mathbf{ZU}^T\|_{\mathrm{F}}^2,\end{aligned}$$

where the equalities use the Pythagorean theorem and the inequality uses the bound above. It follows that the best rank-k approximation to $\mathbf{A}$ in the rowspace of $\mathbf{U}^T$ is $[\mathbf{AU}]_k\mathbf{U}^T$. ■

The following lemma shows how to use sketching to find a good space L. For a matrix $\mathbf{A}$, if its SVD is $\mathbf{U\Sigma V}^T$, then the Moore-Penrose pseudoinverse $\mathbf{A}^\dagger$ of $\mathbf{A}$ is equal to $\mathbf{V\Sigma}^\dagger\mathbf{U}^T$, where $\mathbf{\Sigma}^\dagger$ for a diagonal matrix $\mathbf{\Sigma}$ satisfies $\mathbf{\Sigma}^\dagger_{i,i} = 1/\mathbf{\Sigma}_{i,i}$ if $\mathbf{\Sigma}_{i,i} > 0$, and is 0 otherwise.

Lemma 45 *Let $\mathbf{S}$ be an ℓ_2-subspace embedding for any fixed k-dimensional subspace M with probability at least $9/10$, so that $\|\mathbf{Sy}\|_2 = (1 \pm 1/3)\|\mathbf{y}\|_2$ for all $\mathbf{y} \in M$. Further, suppose $\mathbf{S}$ satisfies the $(\sqrt{\varepsilon/k}, 9/10, \ell)$-JL moment property for some $\ell \geq 2$ of Definition 12, so that the conclusion of Theorem 13 holds, namely, that for any fixed matrices $\mathbf{A}$ and $\mathbf{B}$ each with k rows,*

$$\Pr_{\mathbf{S}}[\|\mathbf{A}^T\mathbf{S}^T\mathbf{SB} - \mathbf{A}^T\mathbf{B}\|_{\mathrm{F}} > 3\sqrt{\varepsilon/k}\|\mathbf{A}\|_{\mathrm{F}}\|\mathbf{B}\|_{\mathrm{F}}] \leq \frac{1}{10}.$$

Then the rowspace of $\mathbf{SA}$ contains a $(1+\varepsilon)$ rank-k approximation to $\mathbf{A}$.

Proof: Let $\mathbf{U}_k$ denote the $n \times k$ matrix of top k left singular vectors of $\mathbf{A}$. Consider the quantity

$$\|\mathbf{U}_k(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k) - (\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2. \tag{30}$$

The goal is to show (30) is at most $(1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2$. Note that this implies the lemma, since $\mathbf{U}_k(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)$ is a rank-k matrix inside of the rowspace of $\mathbf{S}\mathbf{A}$.

Since the columns of $\mathbf{A} - \mathbf{A}_k$ are orthogonal to the columns of $\mathbf{U}_k$, by the matrix Pythagorean theorem (applied to columns),

$$\begin{aligned} & \|\mathbf{U}_k(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k) - (\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 \\ = \; & \|\mathbf{U}_k(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2 \\ = \; & \|(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2, \end{aligned}$$

where the second equality uses that the columns of $\mathbf{U}_k$ are orthonormal. It suffices to show $\|(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 = O(\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2$.

With probability at least $9/10$, $\mathbf{S}$ is an ℓ_2-subspace embedding for the column space of $\mathbf{U}_k$, that is $\|\mathbf{S}\mathbf{U}_k\mathbf{x}\|_2 = (1 \pm 1/3)\|\mathbf{U}_k\mathbf{x}\|_2$ for all $\mathbf{x}$. Since $\mathbf{U}_k$ has orthonormal columns, this implies that all of the singular values of $\mathbf{S}\mathbf{U}_k$ are in the range $[2/3, 4/3]$. Since $\mathbf{x}^t\mathbf{U}_k^T\mathbf{S}^T\mathbf{S}\mathbf{U}_k\mathbf{x} = \|\mathbf{S}\mathbf{U}_k\mathbf{x}\|_2^2$, this implies all the singular values of $(\mathbf{S}\mathbf{U}_k)^T\mathbf{S}\mathbf{U}_k = \mathbf{U}_k^T\mathbf{S}^T\mathbf{S}\mathbf{U}_k$ are in the range $[4/9, 16/9] = (1 \pm 7/9)$. It follows that

$$\begin{aligned} (1 \pm 7/9)\|(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 &= \|(\mathbf{S}\mathbf{U}_k)^T\mathbf{S}\mathbf{U}_k(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 \\ &= \|(\mathbf{S}\mathbf{U}_k)^T\mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 \\ &= \|\mathbf{U}_k^T\mathbf{S}^T\mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2. \end{aligned}$$

Since $\mathbf{S}$ satisfies the conclusion of Theorem 13, with probability at least $9/10$,

$$\|\mathbf{U}_k^T\mathbf{S}^T\mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 \leq 9 \cdot \frac{\varepsilon}{k}\|\mathbf{U}_k\|_{\mathrm{F}}^2\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2 \leq 9\varepsilon\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2,$$

which shows that $\|(\mathbf{S}\mathbf{U}_k)^\dagger \mathbf{S}(\mathbf{A} - \mathbf{A}_k)\|_{\mathrm{F}}^2 = O(\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2$. Rescaling ε by a constant factor completes the proof. ■

Lemma 44 and Lemma 45 give a natural way of using sketching to speed up low rank approximation. Namely, given $\mathbf{A}$, first compute $\mathbf{S} \cdot \mathbf{A}$, which is a small number of random linear combinations of the rows of $\mathbf{A}$. Using efficient ℓ_2-subspace embeddings, this can be done in $O(\mathrm{nnz}(\mathbf{A}))$ time, and

$\mathbf{S}$ need only have $\tilde{O}(k/\varepsilon)$ rows. Next, compute an orthogonal basis $\mathbf{U}^T$ for the rowspace of $\mathbf{S} \cdot \mathbf{A}$, which can be done in $\tilde{O}((k/\varepsilon)^2 d)$ time.

Next, compute $\mathbf{AU}$ in $\tilde{O}(\text{nnz}(\mathbf{A})k/\varepsilon)$ time. By invoking Lemma 44, we can now compute $[\mathbf{AU}]_k$, and our overall low rank approximation will be $[\mathbf{AU}]_k\mathbf{U}^T$, which is a $(1+\varepsilon)$-approximation. Note that we can compute the SVD of $\mathbf{AU}$ in $\tilde{O}((k/\varepsilon)^2 n)$ time, thereby giving us $[\mathbf{AU}]_k$. This allows us to obtain the SVD $\tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^T$ of $[\mathbf{AU}]_k$ in this amount of time as well. We don't require explicitly outputting the product of $[\mathbf{AU}]_k$ and $\mathbf{U}^T$, since this may be a dense matrix and so would require at least nd time to write down. In applications, it is usually better to have a factored form. Notice that $\tilde{\mathbf{V}}^T\mathbf{U}^T$ has orthonormal rows, since $\tilde{\mathbf{V}}^T\mathbf{U}^T\mathbf{U}\tilde{\mathbf{V}}$ is the identity. Therefore, we can output $\tilde{\mathbf{U}}, \tilde{\mathbf{\Sigma}}, \tilde{\mathbf{V}}^T\mathbf{U}^T$, which is the SVD of a rank-k matrix providing a $(1+\varepsilon)$-approximation.

The overall time of the above algorithm is $\tilde{O}(\text{nnz}(\mathbf{A}))k/\varepsilon + (n+d)(k/\varepsilon)^2)$. While this is a significant improvement over computing the SVD of $\mathbf{A}$, which would take $\min(nd^2, n^2 d)$ time, we could still hope to achieve a leading order running time of $O(\text{nnz}(\mathbf{A}))$ as opposed to $\tilde{O}(\text{nnz}(\mathbf{A})k/\varepsilon)$. The dominant cost is actually in computing $\mathbf{AU}$, the coordinate representation of the rows of $\mathbf{A}$ in the rowspace of $\mathbf{U}^T$. That is, it is inefficient to directly project the rows of $\mathbf{A}$ onto $\mathbf{U}$.

Fortunately, we can cast this projection problem as a regression problem, and solve it approximately.

Theorem 46 *Now let $\mathbf{R}$ be a $(1+O(\varepsilon))$-approximate ℓ_2-subspace embedding for the row space of $\mathbf{SA}$, where $\mathbf{S}$ is as in Lemma 45. Then*

$$\|\mathbf{AR}(\mathbf{SAR})^\dagger\mathbf{SA} - \mathbf{A}\|_{\text{F}}^2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}^2.$$

Furthermore,

$$\|[\mathbf{ARU}]_k\mathbf{U}^T(\mathbf{SAR})^\dagger\mathbf{SA} - \mathbf{A}\|_{\text{F}}^2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}^2,$$

where $\mathbf{UU}^T = (\mathbf{SAR})^\dagger\mathbf{SAR}$ is the projection matrix onto the rowspace of $\mathbf{SAR}$.

The time to compute these approximations is $O(\text{nnz}(\mathbf{A}))+(n+d)\text{poly}(k/\varepsilon)$.

Proof: Lemma 45 implies that

$$\min_{\mathbf{Y}} \|\mathbf{YSA} - \mathbf{A}\|_{\text{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\text{F}},$$

The minimizer of the regression problem $\min_{\mathbf{Y}} \|\mathbf{YSAR} - \mathbf{AR}\|_{\mathrm{F}}$ is equal to $\mathbf{Y} = \mathbf{AR}(\mathbf{SAR})^{\dagger}$, and since $\mathbf{R}$ is a subspace embedding we have

$$\|\mathbf{AR}(\mathbf{SAR})^{\dagger}\mathbf{SA} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon) \min_{\mathbf{Y}} \|\mathbf{YSA} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon)^2 \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}},$$

implying the first part of the theorem after rescaling ε by a constant factor.

For the second part of the theorem, note that Lemma 45 gives the stronger guarantee that

$$\min_{\text{rank } k\mathbf{Y}} \|\mathbf{YSA} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}.$$

By the properties of an ℓ_2-subspace embedding, we thus have if $\mathbf{Z}$ is the solution to the regression problem

$$\min_{\text{rank k}\mathbf{Z}} \|\mathbf{ZSAR} - \mathbf{AR}\|_{\mathrm{F}},$$

then

$$\|\mathbf{ZSA} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon) \min_{\text{rank } k\mathbf{Y}} \|\mathbf{YSA} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon)^2 \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}.$$

Therefore, it suffices to find $\mathbf{Z}$. Note that $\mathbf{ZSAR}$ is the best rank-k approximation to $\mathbf{AR}$ in the rowspace of $\mathbf{SAR}$. Therefore, by Lemma 44, $\mathbf{ZSAR} = [(\mathbf{AR})\mathbf{U}]_k\mathbf{U}^T$, where $\mathbf{UU}^T = (\mathbf{SAR})^{\dagger}\mathbf{SAR}$ is the projector onto the row space of $\mathbf{SAR}$. Note that $\mathbf{SAR}(\mathbf{SAR})^{\dagger} = \mathbf{I}$ since $\mathbf{S}$ has fewer rows than columns, and therefore

$$\mathbf{ZSA} = [(\mathbf{AR})\mathbf{U}]_k\mathbf{U}^T(\mathbf{SAR})^{\dagger}\mathbf{SA}.$$

For the time complexity, the dominant cost is in computing $\mathbf{AR}$ and $\mathbf{SA}$, both of which can be done in $O(\text{nnz}(\mathbf{A}))$ time. The remaining operations are on matrices for which at least one dimension is $\text{poly}(k/\varepsilon)$, and therefore can be computed in $(n+d)\text{poly}(k/\varepsilon)$ time. ■

While the representation $\mathbf{AR}(\mathbf{SAR})^{\dagger}\mathbf{SA}$ in Theorem 46 might be useful in its own right as a low rank approximation to $\mathbf{A}$, given it is technically a bicriteria solution since its rank may be $O(k/\varepsilon + k\text{poly}\log k)$, whereas the original problem formulation wants our representation to have rank at most k. The second part of Theorem 46 gives a rank-k approximation.

4.2 CUR decomposition

We now give an alternative to low rank approximation which involves finding a decomposition of an $n \times n$ matrix $\mathbf{A}$ into $\mathbf{C} \cdot \mathbf{U} \cdot \mathbf{R}$, where $\mathbf{C}$ is a subset of columns of $\mathbf{A}$, $\mathbf{R}$ is a subset of rows of $\mathbf{A}$, and $\mathbf{U}$ is a low rank matrix. Ideally, we would like the following properties:

1. $\|\mathbf{CUR} - \mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}$.
2. $\mathbf{C}$ is $n \times c$ for a small value of c. Similarly, $\mathbf{R}$ is $r \times n$ for a small value of r.
3. $\mathbf{U}$ has rank k.
4. The matrices $\mathbf{C}$, $\mathbf{U}$, and $\mathbf{R}$ can be found quickly, ideally in $\mathrm{nnz}(\mathbf{A}) + \mathrm{poly}(k/\varepsilon)n$ time.

A CUR decomposition of a matrix is thus a rank-k approximation whose column space and row space are spanned by a small subset of actual rows and columns of $\mathbf{A}$. This often makes it more interpretable than a generic low rank approximation, or even the SVD, whose column and row spaces are spanned by arbitrary linear combinations of all of the columns and rows of $\mathbf{A}$, respectively, see, e.g., [86] for a discussion of this.

Before discussing the details of some of the available CUR algorithms in [37, 38, 41, 87, 52, 122, 21], we briefly mention a similar problem which constructs factorizations of the form $\mathbf{A} = \mathbf{CX}+\mathbf{E}$, where $\mathbf{C}$ contains columns of $\mathbf{A}$ and $\mathbf{X}$ has rank at most k. There are also optimal algorithms for this problem [19, 57], in both the spectral and the Frobenius norm. Indeed, to obtain a relative-error optimal CUR, one uses a sampling method from [19], which allows to select $O(k)$ columns and rows. For a more detailed discussion of this CX problem, which is also known as CSSP (Column Subset Selection Problem) see [20, 19, 57].

Drineas and Kannan brought CUR factorizations to the theoretical computer science community in [37]; we refer the reader to the jounral version of their work together with Mahoney [38]. Their main algorithm (see Theorem 5 in [38]) is randomized and samples columns and rows from $\mathbf{A}$ with probabilities proportional to their Euclidean length. The running time of this algorithm is linear in m and n and proportional to a small-degree polynomial in k and $1/\varepsilon$, for some $\varepsilon > 0$, but the approximation bound is additive rather than relative (see Theorem 3.1 in [37]): with $c = O(k/\varepsilon^4)$ columns and $r = O(k/\varepsilon^2)$ rows the bound is $\|\mathbf{A} - \mathbf{CUR}\|_{\mathrm{F}}^2 \leq \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2 + \varepsilon\|\mathbf{A}\|_{\mathrm{F}}^2$.

The first relative-error CUR algorithm appeared in [41] (see Theorem 2 of [41]). The algorithm of [41] is based on subspace sampling and requires $c = O(k \log(k/\varepsilon^2) \log \delta^{-1})$ columns and $r = O(c \log(c/\varepsilon^2) \log \delta^{-1})$ rows to construct a relative-error CUR with failure probability δ. The running time of the method in [41] is $O(mn \min\{m, n\})$, since subspace sampling is based on sampling with probabilities proportional to the so-called leverage scores, i.e., the row norms of the matrix $\mathbf{V}_k$ from the SVD of $\mathbf{A}$.

Mahoney and Drineas [87], using again subspace sampling, improved slightly upon the number of columns and rows, compared to [41], but achieved only a constant factor error (see Eqn.(5) in [87]). Gittens and Mahoney [52] discuss CUR decompositions on symmetric positive semidefinite (SPSD) matrices and present approximation bounds for Frobenius, trace, and spectral norms (see Lemma 2 in [52]). Using the near-optimal column subset selection methods in [19] along with a novel adaptive sampling technique, Wang and Zhang [122] present a CUR algorithm selecting $c = (2k/\varepsilon)(1 + o(1))$ columns and $r = (2k/\varepsilon^2)(1 + \varepsilon)(1 + o(1))$ rows from $\mathbf{A}$ (see Theorem 8 in [122]). The running time of this algorithm is

$$O(mnk\varepsilon^{-1} + mk^3\varepsilon^{-\frac{2}{3}} + nk^3\varepsilon^{-\frac{2}{3}} + mk^2\varepsilon^{-2} + nk^2\varepsilon^{-4}).$$

Boutsidis and the author [21] improve this to achieve a simlutaneously optimal $c = r = O(k/\varepsilon)$, and $\text{rank}(\mathbf{U}) = k$. This in fact is optimal up to constant factors, as shown by [21] by presenting a matching lower bound. Boutsidis and the author also show how to do this in $O(\text{nnz}(\mathbf{A}) \log n) + n \cdot \text{poly}(k/\varepsilon)$ time. There is also some desire to make the CUR decomposition deterministic. We will see that this is possible as well, as shown in [21].

Finally, there are several interesting results on CUR developed within the numerical linear algebra community [118, 119, 55, 54, 61, 101, 95, 56, 15, 112]. For example, [118, 119, 55, 54] discuss the so-called skeleton approximation, which focuses on the spectral norm version of the CUR problem via selecting exactly k columns and k rows. The algorithms there are deterministic, run in time proportional to the time to compute the rank k SVD of $\mathbf{A}$, and achieve bounds of the order,

$$\|\mathbf{A} - \mathbf{CUR}\|_2 \leq O(\sqrt{k(n-k)} + \sqrt{k(m-k)})\|\mathbf{A} - \mathbf{A}_k\|_2.$$

We now outline the approach of Boutsidis and the author [21]. A key lemma we need is the following, which is due to Boutsidis, Drineas, and Magdon-Ismail [19].

Lemma 47 *Let $\mathbf{A} = \mathbf{A}\mathbf{Z}\mathbf{Z}^T + \mathbf{E} \in \mathbb{R}^{m\times n}$ be a low-rank matrix factorization of $\mathbf{A}$, with $\mathbf{Z} \in \mathbb{R}^{n\times k}$, and $\mathbf{Z}^T\mathbf{Z} = \mathbf{I}_k$. Let $\mathbf{S} \in \mathbb{R}^{n\times c}$ ($c \geq k$) be any matrix such that $rank(\mathbf{Z}^T\mathbf{S}) = rank(\mathbf{Z}) = k$. Let $\mathbf{C} = \mathbf{A}\mathbf{S} \in \mathbb{R}^{m\times c}$. Then,*

$$\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\|_{\mathrm{F}}^2 \leq \|\mathbf{A} - \Pi_{C,k}(\mathbf{A})\|_{\mathrm{F}}^2 \leq \|\mathbf{E}\|_{\mathrm{F}}^2 + \|\mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\|_{\mathrm{F}}^2.$$

Here, $\Pi_{C,k}(\mathbf{A}) = \mathbf{C}\mathbf{X}_{opt} \in \mathbb{R}^{m\times n}$, where $\mathbf{X}_{opt} \in \mathbb{R}^{c\times n}$ has rank at most k, $\mathbf{C}\mathbf{X}_{opt}$ is the best rank k approximation to $\mathbf{A}$ in the column space of $\mathbf{C}$, and $(\mathbf{Z}^T\mathbf{S})^\dagger$ denotes the Moore-Penrose pseudoinverse of $\mathbf{Z}^T\mathbf{S}$.

Proof: First note that $\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\|_{\mathrm{F}}^2 \leq \|\mathbf{A} - \Pi_{C,k}(\mathbf{A})\|_{\mathrm{F}}^2$ since $\mathbf{C}\mathbf{C}^\dagger\mathbf{A}$ is the projection of the columns of $\mathbf{A}$ onto the column space of $\mathbf{C}$, whereas $\Pi_{C,k}(\mathbf{A})$ is the best rank-k approximation to $\mathbf{A}$ in the column space of $\mathbf{C}$.

For the second inequality in the lemma statement, the main idea in the proof is to consider the matrix $\mathbf{X} = \mathbf{C}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T$. Since this matrix is in the column space of $\mathbf{C}$, we have

$$\|\mathbf{A} - \Pi_{C,k}(\mathbf{A})\|_{\mathrm{F}}^2 \leq \|\mathbf{A} - \mathbf{X}\|_{\mathrm{F}}^2,$$

since $\Pi_{C,k}(\mathbf{A})$ is the best rank-k approximation to $\mathbf{A}$ inside the column space of $\mathbf{C}$.

Manipulating $\mathbf{A} - \mathbf{X}$, we have that $\|\mathbf{A} - \mathbf{C}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2$ is equal to

$$\begin{aligned}
&= \|\mathbf{A}\mathbf{Z}\mathbf{Z}^T + \mathbf{E} - (\mathbf{A}\mathbf{Z}\mathbf{Z}^T + \mathbf{E})\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2 \\
&= \|\mathbf{A}\mathbf{Z}\mathbf{Z}^T - \mathbf{A}\mathbf{Z}\mathbf{Z}^T\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T + \mathbf{E} - \mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2 \\
&= \|\mathbf{E} - \mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2 \\
&= \|\mathbf{E}\|_{\mathrm{F}}^2 + \|\mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2,
\end{aligned}$$

where the first equality uses that $\mathbf{A} = \mathbf{A}\mathbf{Z}\mathbf{Z}^T + \mathbf{E}$ and that $\mathbf{C} = \mathbf{A}\mathbf{S}$, the second equality is a rearrangement of terms, the third equality uses that $\mathrm{rank}(\mathbf{Z}^T\mathbf{S}) = k$ and so $(\mathbf{Z}^T\mathbf{S})(\mathbf{Z}^T\mathbf{S})^\dagger = \mathbf{I}_k$, and the last equality follows from the Pythagorean theorem since $\mathbf{E} = \mathbf{A}(\mathbf{I}_k - \mathbf{Z}\mathbf{Z}^T)$ has rows orthogonal to the row space of $\mathbf{Z}^T$, while $\mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T$ has rows in the row space of $\mathbf{Z}^T$. Finally, noting that

$$\|\mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\mathbf{Z}^T\|_{\mathrm{F}}^2 \leq \|\mathbf{E}\mathbf{S}(\mathbf{Z}^T\mathbf{S})^\dagger\|_{\mathrm{F}}^2\|\mathbf{Z}^T\|_{\mathrm{F}}^2,$$

by submultiplicativity, and that $\|\mathbf{Z}^T\|_2 = 1$, completes the proof. ■

We will apply Lemma 47 twice, and adaptively. First, we compute an $n \times k$ matrix $\mathbf{Z}$ with orthonormal columns for which $\|\mathbf{A} - \mathbf{A}\mathbf{Z}\mathbf{Z}^T\|_{\mathrm{F}}^2 \leq (1 +$

$\frac{1}{9})\|\mathbf{A}-\mathbf{A}_k\|_{\mathrm{F}}^2$. This can be done in $O(\mathrm{nnz}(\mathbf{A}))+n\cdot\mathrm{poly}(k)$ time as shown in the second part of Theorem 46 of the previous section. Specifically, from the statement of that theorem, for an $n\times d$ matrix $\mathbf{A}$, the column space of $[\mathbf{ARU}]_k$ spans a $(1+\varepsilon)$ rank-k approximation to $\mathbf{A}$, where $\mathbf{U}$ satisfies $\mathbf{UU}^T=(\mathbf{SAR})^\dagger\mathbf{SAR}$. We can apply that theorem to $\mathbf{A}^T$ to obtain a $k\times d$ matrix $\mathbf{Z}^T$ which spans a $(1+\varepsilon)$ rank-k approximation to $\mathbf{A}$.

Given $\mathbf{Z}$, we will sample $O(k\log k)$ columns of $\mathbf{Z}$ proportional to the squared row norms, or leverage scores of $\mathbf{Z}$. Let $\ell_i^2=\|e_i^T\mathbf{Z}\|_2^2$ be the i-th leverage score. Since $\sum_{i=1}^n\ell_i^2=k$, the $p_i=\ell_i^2/k$ values define a probability distribution.

We now invoke the RandSampling$(\mathbf{Z},s,p)$ algorithm of Definition 16 with $s=\Theta(k(\log k))$. By the guarantee of Theorem 17, we obtain matrices $\mathbf{\Omega}$ and $\mathbf{D}$ for which with probability $1-1/\mathrm{poly}(k)$, for all $i\in[k]$,

$$\frac{1}{2}\leq\sigma_i^2(\mathbf{Z}^T\mathbf{\Omega D})\leq\frac{3}{2}. \tag{31}$$

Here $\mathbf{\Omega D}$ implements sampling s columns of $\mathbf{Z}^T$ and re-scaling them by the coefficients in the diagonal matrix $\mathbf{D}$. We also record the following simple fact about the RandSampling algorithm.

Lemma 48 *With probability at least .9 over the randomness in the* R*andSampling*$(\mathbf{Z},s,p)$ *algorithm,*

$$\|\mathbf{Z}^T\mathbf{\Omega D}\|_{\mathrm{F}}^2\leq 10\|\mathbf{Z}^T\|_{\mathrm{F}}^2.$$

Proof: We show $\mathbf{E}[\|\mathbf{Z}^T\mathbf{\Omega D}\|_{\mathrm{F}}^2]=\|\mathbf{Z}^T\|_{\mathrm{F}}^2$. By linearity of expectation, it suffices to show for a fixed column $j\in[s]$, $\mathbf{E}[\|(\mathbf{Z}^T\mathbf{\Omega D})_{*j}\|_{\mathrm{F}}^2]=\|\mathbf{Z}^T\|_{\mathrm{F}}^2/s$. We have,

$$\mathbf{E}[\|(\mathbf{Z}^T\mathbf{\Omega D})_{*j}\|_{\mathrm{F}}^2] \;=\; \sum_{i=1}^n p_i\cdot\frac{1}{p_is}\|\mathbf{Z}_{*i}^T\|_2^2=\frac{1}{s}\|\mathbf{Z}^T\|_{\mathrm{F}}^2,$$

as needed. The lemma now follows by Markov's bound. ■

Our algorithm thus far, is given $\mathbf{A}$, to compute $\mathbf{Z}$, and then to compute $\mathbf{\Omega}$ and $\mathbf{D}$ via RandSampling$(\mathbf{Z},s,p)$, where $s=O(k\log k)$. At this point, we could look at $\mathbf{A\Omega D}$, which samples s columns of $\mathbf{A}$. While this set of columns can be shown to have good properties, namely, its column space contains a k-dimensional subspace spanning an $O(1)$ rank-k approximation to A, which can then be used as a means for obtaining a $(1+\varepsilon)$-approximation by adaptive sampling, as will be seen in §4.2.2. However, the number of

columns is $O(k \log k)$, which would result in an overall CUR decomposition with at least $O(k \log k/\varepsilon)$ columns and rows using the techniques below, which is larger by a $\log k$ factor than what we would like (namely, $O(k/\varepsilon)$ columns and rows).

We therefore wish to first downsample the s columns of $\mathbf{A}$ we have now to $O(k)$ columns by right-multiplying by a matrix $\mathbf{S} \in \mathbb{R}^{s \times k}$, so that $\mathbf{Z}^T \mathbf{\Omega} \mathbf{D} \mathbf{S}$ has rank k and has a reasonably large k-th singular value.

To proceed, we need an algorithm in the next subsection, which uses a method of Batson, Spielman, and Srivastava [14] refined for this application by Boutsidis, Drineas, and Magdon-Ismail [19].

4.2.1 Batson-Spielman-Srivastava sparsification

Define the parameters

$$\delta_{LOW} = 1, \quad \delta_{UP} = \frac{\|\mathbf{A}\|_{\mathrm{F}}^2}{1 - \sqrt{\frac{k}{r}}}.$$

Define the function

$$\phi(L, \mathbf{M}) = \mathrm{Tr}\left((\mathbf{M} - L\mathbf{I})^{-1}\right) = \sum_{i=1}^{k} \frac{1}{\lambda_i(\mathbf{M}) - L}.$$

Note that $\phi(L, \mathbf{M})$ measures how far the eigenvalues of $\mathbf{M}$ are from L, since the closer they are to L, the more ϕ "blows up".

Also, define the functions

$$UP(a, \delta_{UP}) = \delta_{UP}^{-1} a^T a,$$

and

$$\begin{aligned} LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}, L) \quad = \quad & \frac{\mathbf{v}^T (\mathbf{M} - (L + \delta_{LOW})\mathbf{I}_k)^{-2} \mathbf{v}}{\phi(L + \delta_{LOW}, \mathbf{M}) - \phi(L, \mathbf{M})} \\ & - \mathbf{v}^T (\mathbf{M} - (L + \delta_{LOW})\mathbf{I}_k)^{-1} \mathbf{v}. \end{aligned}$$

These two functions will be used in each iteration of Algorithm 2 below to make progress in each iteration. What we will be able to show is that in each iteration of the algorithm, the current value of our potential function UP will be less than the current value of our potential function LOW, and this will enable us to choose a new vector $\mathbf{v}_i$ and add $\mathbf{v}_i \mathbf{v}_i^T$ in our decomposition of the identity. This corresponds to a rank-one update of

Algorithm 2 Deterministic Dual Set Spectral Sparsification

Input:

- $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ with $\sum_{i=1}^n \mathbf{v}_i \mathbf{v}_i^T = \mathbf{I}_k$
- $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$.

Output: A set of n non-negative weights s_i, at most r of which are non-zero.

1. Initialize $s_0 = \mathbf{0}_{n\times 1}$, $\mathbf{M}_0 = \mathbf{0}_{k\times k}$
2. For $\tau = 0, \ldots, r-1$
   - Set $L_\tau = \tau - \sqrt{rk}$ and $U_\tau = \tau\delta_{UP}$.
   - Find an index $j \in \{1, 2, \ldots, n\}$ such that $UP(\mathbf{a}_j, \delta_{UP}) \leq LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}_\tau, L_\tau)$.
   - Let $t^{-1} = \frac{1}{2}\left(UP(\mathbf{a}_j, \delta_{UP}) + LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}_\tau, L_\tau)\right)$.
   - Update the j-th component of $\mathbf{s}$ and $\mathbf{M}_\tau$:
   $$s_{\tau+1}[j] = s_\tau[j] + t,\ \mathbf{M}_{\tau+1} = \mathbf{M}_\tau + t\mathbf{v}_j\mathbf{v}_j^T.$$
3. Return $\mathbf{s} = r^{-1}(1 - \sqrt{k/r}) \cdot \mathbf{s}_r$.

our current decomposition $\mathbf{M}$, and we use the Sherman-Morrison-Woodbury formula to analyze how the eigenvalues of $\mathbf{M}$ change, as well as how the values of UP and LOW change, given this rank-one update.

The following theorem shows correctness of the Deterministic Dual Set Spectral Sparsification algorithm described in Algorithm 2.

Theorem 49 *(Dual Set Spectral-Frobenius Sparsification) Let $\mathbf{v}_i \in \mathbb{R}^k$ for $i = 1, \ldots, n$ with $k < n$, and $\sum_{i=1}^n \mathbf{v}_i\mathbf{v}_i^T = \mathbf{I}_k$. Let $A = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ be an arbitrary set of vectors, where $\mathbf{a}_i \in \mathbb{R}^\ell$ for all i. Then, given an integer r such that $k < r \leq n$, there exists a set of weights $s_i \geq 0$ $(i = 1, \ldots, n)$, at most r of which are non-zero, such that*

$$\begin{aligned}
\lambda_k\left(\sum_{i=1}^n s_i\mathbf{v}_i\mathbf{v}_i^T\right) &\geq \left(1 - \sqrt{\frac{k}{r}}\right)^2, \\
\mathrm{Tr}\left(\sum_{i=1}^n s_i\mathbf{a}_i\mathbf{a}_i^T\right) &\leq \mathrm{Tr}\left(\sum_{i=1}^n \mathbf{a}_i\mathbf{a}_i^T\right) = \sum_{i=1}^n \|\mathbf{a}_i\|_2^2.
\end{aligned}$$

Equivalently, if $\mathbf{V} \in \mathbb{R}^{n\times k}$ *is a matrix whose rows are the vectors* $\mathbf{v}_i^T$, $\mathbf{A} \in \mathbb{R}^{n\times \ell}$ *is a matrix whose rows are the vectors* $\mathbf{a}_i^T$, *and* $\mathbf{S} \in \mathbb{R}^{n\times r}$ *is the sampling matrix containing the weights* $s_i > 0$, *then:*

$$\sigma_k\left(\mathbf{V}^T\mathbf{S}\right) \geq (1-\sqrt{k/r})^2, \qquad \|\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \|\mathbf{A}^T\|_{\mathrm{F}}^2.$$

The weights s_i *can be computed in* $O\left(rnk^2 + n\ell\right)$ *time. We denote this procedure as*

$$\mathbf{S} = BssSampling(\mathbf{V}, \mathbf{A}, r).$$

Proof: First, regarding the time complexity, in each of r iterations we need to compute L on each of the n vectors $\mathbf{v}_j$. The costly matrix inversion in the definition of L can be performed once in $O(k^3)$ time, which also upper bounds the time to compute $\phi(L+\delta_{LOW}, \mathbf{M})$ and $\phi(L, \mathbf{M})$. Given these quantities, computing L for a single vector $\mathbf{v}_j$ takes $O(k^2)$ time and so for all n vectors $\mathbf{v}_j$ $O(nk^2)$ time, and across all r iterations, $O(rnk^2)$ time. Computing $UP(\mathbf{a}_j, \delta_{UP})$ just corresponds to computing the Euclidean norm of $\mathbf{a}_j$, and these can be computed once at the beginning of the algorithm in $O(n\ell)$ time. This implies the overall time complexity of the lemma.

We now turn to correctness. The crux of the analysis turns out to be to show there always exists an index j in each iteration for which

$$UP(\mathbf{a}_j, \delta_{UP}) \leq LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}_\tau, L_\tau).$$

For a real symmetric matrix $\mathbf{M}$ we let $\lambda_i(\mathbf{M})$ denote its i-th largest eigenvalue of matrix $\mathbf{M}$. It will be useful to observe that for L_τ and U_τ as defined by the algorithm, we have chosen the definitions so that $L_\tau + \delta_{LOW} = L_{\tau+1}$ and $U_\tau + \delta_{UP} = U_{\tau+1}$.

We start with a lemma which uses the Sherman-Morrison-Woodbury identity to analyze a rank-1 perturbation.

Lemma 50 *Fix* $\delta_{LOW} > 0$, $\mathbf{M} \in \mathbb{R}^{k\times k}$, $\mathbf{v} \in \mathbb{R}^k$, *and* $L < \lambda_k(\mathbf{M})$. *If* $t > 0$ *satisfies*

$$t^{-1} \leq LOW(\mathbf{v}, \delta_{LOW}, \mathbf{M}, L),$$

then

1. $\lambda_k(\mathbf{M} + t\mathbf{v}\mathbf{v}^T) \geq L + \delta_{LOW}$, *and*
2. $\phi(L+\delta_{LOW}, \mathbf{M} + t\mathbf{v}\mathbf{v}^T) \leq \phi(L, \mathbf{M})$.

Proof: Note that by definition of $\phi(L, \mathbf{M})$, given that $\lambda_k(\mathbf{M}) > L$, and $\phi(L, \mathbf{M}) \leq \frac{1}{\delta_{LOW}}$, this implies that $\lambda_k(\mathbf{M}) > L + \delta_{LOW}$, and so for any $t > 0$, $\lambda_k(\mathbf{M} + t\mathbf{v}\mathbf{v}^T) > L + \delta_{LOW}$. This proves the first part of the lemma.

For the second part, we use the following well-known formula.

Fact 5 *(Sherman-Morrison-Woodbury Formula) If* $\mathbf{M}$ *is an invertible* $n \times n$ *matrix and* $\mathbf{v}$ *is an* n*-dimensional vector, then*

$$(\mathbf{M} + \mathbf{v}\mathbf{v}^T)^{-1} = \mathbf{M}^{-1} - \frac{\mathbf{M}^{-1}\mathbf{v}\mathbf{v}^T\mathbf{M}^{-1}}{1 + \mathbf{v}^T\mathbf{M}^{-1}\mathbf{v}}.$$

Letting $L' = L + \delta_{LOW}$, we have

$$\begin{aligned}
\phi(L + \delta_{LOW}, \mathbf{M} + t\mathbf{v}\mathbf{v}^T) &= \mathrm{Tr}\left((\mathbf{M} + t\mathbf{v}\mathbf{v}^T - L'\mathbf{I})^{-1}\right) \\
&= \mathrm{Tr}\left((\mathbf{M} - L'\mathbf{I})^{-1}\right) \\
&\quad -\mathrm{Tr}\left(\frac{t(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}}{1 + t\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}}\right) \\
&= \mathrm{Tr}\left((\mathbf{M} - L'\mathbf{I})^{-1}\right) \\
&\quad -\frac{t\mathrm{Tr}\left(\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}\right)}{1 + t\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}} \\
&= \phi(L', \mathbf{M}) - \frac{t\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-2}\mathbf{v}}{1 + t\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}} \\
&= \phi(L, \mathbf{M}) + (\phi(L', \mathbf{M}) - \phi(L, \mathbf{M})) \\
&\quad -\frac{\mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-2}\mathbf{v}}{1/t + \mathbf{v}^T(\mathbf{M} - L'\mathbf{I})^{-1}\mathbf{v}} \\
&\leq \phi(L, \mathbf{M}),
\end{aligned}$$

where the first equality uses the definition of ϕ, the second equality uses the Sherman-Morrison Formula, the third equality uses that the trace is a linear operator and satisfies $\mathrm{Tr}(\mathbf{X}\mathbf{Y}) = \mathrm{Tr}(\mathbf{Y}\mathbf{X})$, the fourth equality uses the definition of ϕ and that the trace of a number is the number itself, the fifth equality follows by rearranging terms, and the final inequality follows by assumption that $t^{-1} \leq LOW(\mathbf{v}, \delta_{LOW}, \mathbf{M}, L)$. ■

We also need the following lemma concerning properties of the UP function.

Lemma 51 *Let* $\mathbf{W} \in \mathbb{R}^{\ell \times \ell}$ *be a symmetric positive semi-definite matrix, let* $\mathbf{a} \in \mathbb{R}^{\ell}$ *be a vector, and let* $U \in \mathbb{R}$ *satisfy* $U > Tr(\mathbf{W})$*. If* $t > 0$ *satisfies*

$$UP(\mathbf{a}, \delta_{UP}) \leq t^{-1},$$

then

$$Tr(\mathbf{W} + t\mathbf{v}\mathbf{v}^T) \leq U + \delta_{UP}.$$

Proof: By the assumption of the lemma,

$$UP(\mathbf{a}, \delta_{UP}) = \delta_{UP}^{-1} \mathbf{a}^T \mathbf{a} \leq t^{-1},$$

or equivalently, $t\mathbf{a}^T\mathbf{a} \leq \delta_{UP}$. Hence,

$$\begin{aligned} \mathrm{Tr}\left(\mathbf{W} + t\mathbf{a}\mathbf{a}^T\right) - U - \delta_{UP} &= \mathrm{Tr}\left(\mathbf{W}\right) - U + (t\mathbf{a}^T\mathbf{a} - \delta_{UP}) \\ &\leq \mathrm{Tr}\left(\mathbf{W}\right) - U < 0. \end{aligned}$$

■

Equipped with Lemma 50 and Lemma 51, we now prove the main lemma we need.

Lemma 52 *At every iteration $\tau = 0, \ldots, r-1$, there exists an index $j \in \{1, 2, \ldots, n\}$ for which*

$$UP(\mathbf{a}_j, \delta_{UP}) \leq t^{-1} \leq LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}_\tau, L_\tau).$$

Proof: It suffices to show that

$$\sum_{i=1}^{n} UP(\mathbf{a}_j, \delta_{UP}) = 1 - \sqrt{\frac{k}{r}} \leq \sum_{i=1}^{n} LOW(\mathbf{v}_i, \delta_{LOW}, \mathbf{M}_\tau, L_\tau). \tag{32}$$

Indeed, if we show (32), then by averaging there must exist an index j for which

$$UP(\mathbf{a}_j, \delta_{UP}) \leq t^{-1} \leq LOW(\mathbf{v}_j, \delta_{LOW}, \mathbf{M}_\tau, L_\tau).$$

We first prove the equality in (32) using the definition of δ_{UP}. Observe that it holds that

$$\begin{aligned} \sum_{i=1}^{n} UP(\mathbf{a}_i, \delta_{UP}) &= \delta_{UP}^{-1} \sum_{i=1}^{n} \mathbf{a}_i^T \mathbf{a}_i \\ &= \delta_{UP}^{-1} \sum_{i=1}^{n} \|\mathbf{a}_i\|_2^2 \\ &= 1 - \sqrt{\frac{k}{r}}. \end{aligned}$$

We now prove the inequality in (32). Let λ_i denote the i-th largest eigenvalue of $\mathbf{M}_\tau$. Using that $\mathrm{Tr}\left(\mathbf{v}^T\mathbf{Y}\mathbf{v}\right) = \mathrm{Tr}\left(\mathbf{Y}\mathbf{v}\mathbf{v}^T\right)$ and $\sum_i \mathbf{v}_i\mathbf{v}_i^T = \mathbf{I}_k$, we have

$$\begin{aligned}\sum_{i=1}^{n} LOW(\mathbf{v}_i, \delta_{LOW}, \mathbf{M}_\tau, L_\tau) &= \frac{\mathrm{Tr}\left((\mathbf{M}_\tau - L_{\tau+1}\mathbf{I}_k)^{-2}\right)}{\phi(L_{\tau+1}, \mathbf{M}_\tau) - \phi(L_\tau, \mathbf{M}_\tau)} \\ &\quad -\phi(L_{\tau+1}, \mathbf{M}_\tau) \\ &= \frac{\sum_{i=1}^k \frac{1}{(\lambda_i - L_{\tau+1})^2}}{\delta_{LOW} \sum_{i=1}^k \frac{1}{(\lambda_i - L_{\tau+1})(\lambda_i - L_\tau)}} \\ &\quad -\sum_{i=1}^{k} \frac{1}{(\lambda_i - L_{\tau+1})} \\ &= \frac{1}{\delta_{LOW}} - \phi(L_\tau, M_\tau) + \mathcal{E},\end{aligned}$$

where

$$\begin{aligned}\mathcal{E} &= \frac{1}{\delta_{LOW}} \left(\frac{\sum_{i=1}^k \frac{1}{(\lambda_i - L_{\tau+1})^2}}{\sum_{i=1}^k \frac{1}{(\lambda_i - L_\tau + 1)(\lambda_i - L_\tau)}} - 1 \right) \\ &\quad -\delta_{LOW} \sum_{i=1}^{k} \frac{1}{(\lambda_i - L_\tau)(\lambda_i - L_{\tau+1})}.\end{aligned}$$

We will show $\mathcal{E} \geq 0$ below. Given this, we have

$$\phi(L_\tau, \mathbf{M}_\tau) \leq \phi(L_0, \mathbf{M}_0) = \phi(-\sqrt{rk}, \mathbf{0}_{k\times k}) = \frac{-k}{-\sqrt{rk}} = \sqrt{\frac{k}{r}},$$

where the inequality uses Lemma 50. Since $\delta_{LOW} = 1$, we have

$$\sum_{i=1}^{n} LOW(\mathbf{v}_i, \delta_{LOW}, \mathbf{M}_\tau, L_\tau) \geq 1 - \sqrt{\frac{k}{r}},$$

which will complete the proof.

We now turn to the task of showing $\mathcal{E} \geq 0$. The Cauchy-Schwarz inequality implies that for $a_i, b_i \geq 0$, one has $(\sum_i a_i b_i)^2 \leq (\sum_i a_i^2 b_i)(\sum_i b_i)$,

and therefore

$$\begin{aligned}
\mathcal{E}\sum_{i=1}^{k}\frac{1}{(\lambda_i - L_{\tau+1})(\lambda_i - L_\tau)} &= \frac{1}{\delta_{LOW}}\sum_{i=1}^{k}\frac{1}{(\lambda_i - L_{\tau+1})^2(\lambda_i - L_\tau)} \\
&\quad -\delta_{LOW}\left(\sum_{i=1}^{k}\frac{1}{(\lambda_i - L_\tau)(\lambda_i - L_{\tau+1})}\right)^2 \\
&\geq \frac{1}{\delta_{LOW}}\sum_{i=1}^{k}\frac{1}{(\lambda_i - L_{\tau+1})^2(\lambda_i - L_\tau)} \\
&\quad -\sum_{i=1}^{k}\frac{\delta_{LOW}}{(\lambda_i - L_{\tau+1})^2(\lambda_i - L_\tau)}\sum_{i=1}^{k}\frac{1}{\lambda_i - L_\tau} \\
&= \sum_{i=1}^{k}\frac{\left(\frac{1}{\delta_{LOW}} - \delta_{LOW}\cdot\phi(L_\tau, \mathbf{M}_\tau)\right)}{(\lambda_i - L_{\tau+1})^2(\lambda_i - L_\tau)}. \qquad (33)
\end{aligned}$$

Since $\delta_{LOW} = 1$ and we have computed above that $\phi(L, \mathbf{M}) \leq \sqrt{k/r}$, we have $\frac{1}{\delta_{LOW}} - \delta_{LOW}\cdot\phi(L_\tau, \mathbf{M}_\tau) \geq 1 - \sqrt{k/r} > 0$ since $r > k$.

Also,

$$\begin{aligned}
\lambda_i &\geq \lambda_k(\mathbf{M}_\tau) \\
&\geq L_\tau + \frac{1}{\phi(L_\tau, \mathbf{M}_\tau)} \\
&\geq L_\tau + \frac{1}{\phi(L_0, \mathbf{M}_0)} \\
&\geq L_\tau + \sqrt{\frac{r}{k}} \\
&> L_\tau + 1 \\
&= L_{\tau+1}.
\end{aligned}$$

Plugging into (33), we conclude that $\mathcal{E} \geq 0$, as desired. ■

By Lemma 52, the algorithm is well-defined, finding a $t \geq 0$ at each iteration (note that $t \geq 0$ since $t^{-1} \geq UP(a_j, \delta_{UP}) \geq 0$).

It follows by Lemma 50 and induction that for every τ, we have $\lambda_k(\mathbf{M}_\tau) \geq L_\tau$. Similarly, by Lemma 51 and induction, for every τ it holds that $\mathrm{Tr}\,(\mathbf{W}_\tau) \leq U_\tau$.

In particular, for $\tau = r$ we have

$$\lambda_k(\mathbf{M}_r) \geq L_r = r(1 - \sqrt{k/r}), \qquad (34)$$

and

$$\mathrm{Tr}\,(\mathbf{W}_r) \leq U_r = r(1-\sqrt{k/r})^{-1}\|\mathbf{A}\|_{\mathrm{F}}^2. \tag{35}$$

Rescaling by $r^{-1}(1-\sqrt{k/r})$ in Step 3 of the algorithm therefore results in the guarantees on $\lambda_k(\mathbf{M}_r)$ and $\mathrm{Tr}\,(\mathbf{W}_r)$ claimed in the theorem statement.

Finally, note that Algorithm 2 runs in r steps. The vector s of weights is initialized to the all-zero vector, and one of its entries is updated in each iteration. Thus, s will contain at most r non-zero weights upon termination. As shown above, the value t chosen in each iteration is non-negative, so the weights in s are non-negative.

This completes the proof. ■

We will also need the following corollary, which shows how to perform the dual set sparsification much more efficiently if we allow it to be randomized.

Corollary 53 *Let $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ be a decomposition of the identity, where $\mathbf{v}_i \in \mathbb{R}^k$ ($k < n$) and $\sum_{i=1}^n \mathbf{v}_i\mathbf{v}_i^T = \mathbf{I}_k$; let $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ be an arbitrary set of vectors, where $\mathbf{a}_i \in \mathbb{R}^\ell$. Let $\mathbf{W} \in \mathbb{R}^{\xi\times\ell}$ be a randomly chosen sparse subspace embedding with $\xi = O(n^2/\varepsilon^2) < \ell$, for some $0 < \varepsilon < 1$. Consider a new set of vectors $\mathcal{B} = \{\mathbf{W}\mathbf{a}_1, \ldots, \mathbf{W}\mathbf{a}_n\}$, with $\mathbf{W}\mathbf{a}_i \in \mathbb{R}^\xi$. Run Algorithm 2 with $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, $\mathcal{B} = \{\mathbf{W}\mathbf{a}_1, \ldots, \mathbf{W}\mathbf{a}_n\}$, and some integer r such that $k < r \leq n$. Let the output of this be a set of weights $s_i \geq 0$ ($i = 1 \ldots n$), at most r of which are non-zero. Then, with probability at least* 0.99,

$$\begin{aligned}
\lambda_k\left(\sum_{i=1}^n s_i\mathbf{v}_i\mathbf{v}_i^T\right) &\geq \left(1-\sqrt{\frac{k}{r}}\right)^2, \\
\mathrm{Tr}\left(\sum_{i=1}^n s_i\mathbf{a}_i\mathbf{a}_i^T\right) &\leq \frac{1+\varepsilon}{1-\varepsilon}\cdot\mathrm{Tr}\left(\sum_{i=1}^n \mathbf{a}_i\mathbf{a}_i^T\right) \\
&= \frac{1+\varepsilon}{1-\varepsilon}\cdot\sum_{i=1}^n \|\mathbf{a}_i\|_2^2.
\end{aligned}$$

Equivalently, if $\mathbf{V} \in \mathbb{R}^{n\times k}$ is a matrix whose rows are the vectors $\mathbf{v}_i^T$, $\mathbf{A} \in \mathbb{R}^{n\times\ell}$ is a matrix whose rows are the vectors $\mathbf{a}_i^T$, $\mathbf{B} = \mathbf{A}\mathbf{W}^T \in \mathbb{R}^{n\times\xi}$ is a matrix whose rows are the vectors $\mathbf{a}_i^T\mathbf{W}^T$, and $\mathbf{S} \in \mathbb{R}^{n\times r}$ is the sampling matrix containing the weights $s_i > 0$, then with probability at least 0.99,

$$\sigma_k\left(\mathbf{V}^T\mathbf{S}\right) \geq 1-\sqrt{k/r}, \qquad \|\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \frac{1+\varepsilon}{1-\varepsilon}\cdot\|\mathbf{A}\|_{\mathrm{F}}^2.$$

The weights s_i can be computed in $O\left(\mathrm{nnz}(\mathbf{A}) + rnk^2 + n\xi\right)$ time. We denote this procedure as

$$\mathbf{S} = BssSamplingSparse(\mathbf{V}, \mathbf{A}, r, \varepsilon).$$

Proof: The algorithm constructs $\mathbf{S}$ as follows,

$$\mathbf{S} = BssSampling(\mathbf{V}, \mathbf{B}, r).$$

The lower bound for the smallest singular value of $\mathbf{V}$ is immediate from Theorem 49. That theorem also ensures,

$$\|\mathbf{B}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \|\mathbf{B}^T\|_{\mathrm{F}}^2,$$

i.e.,

$$\|\mathbf{W}\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \|\mathbf{W}\mathbf{A}^T\|_{\mathrm{F}}^2.$$

Since $\mathbf{W}$ is an ℓ_2-subspace embedding, we have that with probability at least 0.99 and for all vectors $\mathbf{y} \in \mathbb{R}^n$ simultaneously,

$$(1-\varepsilon)\,\|\mathbf{A}^T\mathbf{y}\|_2^2 \leq \|\mathbf{W}\mathbf{A}^T\mathbf{y}\|_2^2.$$

Apply this r times for $\mathbf{y} \in \mathbb{R}^n$ being columns from $\mathbf{S} \in \mathbb{R}^{n\times r}$ and take a sum on the resulting inequalities,

$$(1-\varepsilon)\,\|\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \|\mathbf{W}\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2.$$

Now, since $\mathbf{W}$ is an ℓ_2-subspace embedding,

$$\|\mathbf{W}\mathbf{A}^T\|_{\mathrm{F}}^2 \leq (1+\varepsilon)\,\|\mathbf{A}^T\|_{\mathrm{F}}^2,$$

which can be seen by applying $\mathbf{W}$ to each of the vectors $\mathbf{W}\mathbf{A}^T\mathbf{e}_i$. Combining all these inequalities together, we conclude that with probability at least 0.99,

$$\|\mathbf{A}^T\mathbf{S}\|_{\mathrm{F}}^2 \leq \frac{1+\varepsilon}{1-\varepsilon}\cdot\|\mathbf{A}^T\|_{\mathrm{F}}^2.$$

■

Implications for CUR. Returning to our CUR decomposition algorithm, letting $\mathbf{M} = \mathbf{Z}_1^T\mathbf{\Omega}_1\mathbf{D}_1$ where $\mathbf{\Omega}_1$ and $\mathbf{D}_1$ are found using RandSampling$(\mathbf{Z}, s, p)$, we apply Corollary 53 to compute $\mathbf{S}_1$ =BssSamplingSparse$(\mathbf{V}_\mathbf{M}, (\mathbf{A}-\mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^T)^T\mathbf{\Omega}_1\mathbf{D}_1, 4k, .5)$, where $\mathbf{V}_\mathbf{M}$ is determined by writing $\mathbf{M}$ in its SVD as $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}_\mathbf{M}^T$.

At this point we set $\mathbf{C}_1 = \mathbf{A}\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1 \in \mathbb{R}^{m\times 4k}$ which contains $c_1 = 4k$ rescaled columns of $\mathbf{A}$.

Lemma 54 *With probability at least .8,*

$$\|\mathbf{A} - \mathbf{C}_1\mathbf{C}_1{}^\dagger\mathbf{A}\|_{\mathrm{F}}^2 \leq 90 \cdot \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2.$$

Proof: We apply Lemma 47 with $\mathbf{Z} = \mathbf{Z}_1 \in \mathbb{R}^{n\times k}$ and $\mathbf{S} = \mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1 \in \mathbb{R}^{n\times c_1}$. First, we show that with probability .9, the rank assumption of Lemma 47 is satisfied for our choice of $\mathbf{S}$, namely, that $\mathrm{rank}(\mathbf{Z}^T\mathbf{S}) = k$. We have

$$\mathrm{rank}(\mathbf{Z}^T\mathbf{S}) = \mathrm{rank}(\mathbf{Z}_1^T\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1) = \mathrm{rank}(\mathbf{M}\mathbf{S}_1) = \mathrm{rank}(\mathbf{V}_{\mathbf{M}}^T\mathbf{S}_1) = k,$$

where the first two equalities follow from the definitions, the third equality follows assuming the $1 - \frac{1}{\mathrm{poly}(k)}$ event of (31) that $\mathrm{rank}(\mathbf{M}) = k$, and the last equality follows from the fact that Corollary 53 guarantees that with probability at least .98, $\sigma_k\left(\mathbf{V}_M^T\mathbf{S}\right) \geq \frac{1}{2}$.

Now applying Lemma 47 with the $\mathbf{C}$ there equal to $\mathbf{C}_1$ and the $\mathbf{E}$ there equal to $\mathbf{E}_1 = \mathbf{A} - \mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^T$, we have

$$\begin{aligned}
\|\mathbf{A} - \mathbf{C}_1\mathbf{C}_1{}^\dagger\mathbf{A}\|_{\mathrm{F}}^2 &\leq \|\mathbf{A} - \Pi_{\mathbf{C}_1,k}(\mathbf{A})\|_{\mathrm{F}}^2 \\
&\leq \|\mathbf{A} - \mathbf{C}_1(\mathbf{Z}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1)^\dagger\mathbf{Z}_1^T\|_{\mathrm{F}}^2 \\
&\leq \|\mathbf{E}_1\|_{\mathrm{F}}^2 + \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1(\mathbf{Z}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1)^\dagger\|_{\mathrm{F}}^2.
\end{aligned}$$

We have that $\|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1(\mathbf{Z}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1)^\dagger\|_{\mathrm{F}}^2$ is at most

$$\begin{aligned}
&\overset{(a)}{\leq} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot \|(\mathbf{Z}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1)^\dagger\|_2^2 \\
&\overset{(b)}{=} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot \|(\mathbf{U}_{\mathbf{M}}\mathbf{\Sigma}_{\mathbf{M}}\mathbf{V}_{\mathbf{M}}^{\mathrm{T}}\mathbf{S}_1)^\dagger\|_2^2 \\
&\overset{(c)}{=} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot \|\left(\mathbf{V}_{\mathbf{M}}^{\mathrm{T}}\mathbf{S}_1\right)^\dagger(\mathbf{U}_{\mathbf{M}}\mathbf{\Sigma}_{\mathbf{M}})^\dagger\|_2^2 \\
&\overset{(d)}{\leq} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot \|\left(\mathbf{V}_{\mathbf{M}}^{\mathrm{T}}\mathbf{S}_1\right)^\dagger\|_2^2 \cdot \|(\mathbf{U}_{\mathbf{M}}\mathbf{\Sigma}_{\mathbf{M}})^\dagger\|_2^2 \\
&\overset{(e)}{=} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot \frac{1}{\sigma_k^2\left(\mathbf{V}_{\mathbf{M}}^{\mathrm{T}}\mathbf{S}_1\right)} \cdot \frac{1}{\sigma_k^2\left(\mathbf{U}_{\mathbf{M}}\mathbf{\Sigma}_{\mathbf{M}}\right)} \\
&\overset{(f)}{\leq} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\mathbf{S}_1\|_{\mathrm{F}}^2 \cdot 8 \\
&\overset{(g)}{\leq} \|\mathbf{E}_1\mathbf{\Omega}_1\mathbf{D}_1\|_{\mathrm{F}}^2 \cdot 8 \\
&\overset{(h)}{\leq} 80\|\mathbf{E}_1\|_{\mathrm{F}}^2
\end{aligned}$$

where (a) follows by the sub-multiplicativity property of matrix norms, (b) follows by replacing $\mathbf{Z}_1\mathbf{\Omega}_1\mathbf{D}_1 = \mathbf{M} = \mathbf{U}_{\mathbf{M}}\mathbf{\Sigma}_{\mathbf{M}}\mathbf{V}_{\mathbf{M}}^{\mathrm{T}}$, (c) follows by the fact

that $\mathbf{U_M \Sigma_M}$ is a full rank $k \times k$ matrix assuming the $1 - \frac{1}{\text{poly}(k)}$ probability event of (31) (d) follows by the sub-multiplicativity property of matrix norms, (e) follows by the connection of the spectral norm of the pseudo-inverse with the singular values of the matrix to be pseudo-inverted, (f) follows if the $1 - \frac{1}{\text{poly}(k)}$ event of (31) occurs and the probability .98 event of Corollary 53 occurs, (g) follows by Corollary 53, and (h) follows by Lemma 48 and by adding a 0.1 to the overall failure probability. So, overall with probability at least 0.8,

$$\|\mathbf{E}_1 \mathbf{\Omega D S}_1 (\mathbf{Z}_1 \mathbf{\Omega D S}_1)^\dagger\|_{\text{F}}^2 \leq 80 \|\mathbf{E}_1\|_{\text{F}}^2,$$

Hence, with the same probability,

$$\|\mathbf{A} - \mathbf{C}_1 \mathbf{C}_1^\dagger \mathbf{A}\|_{\text{F}}^2 \leq \|\mathbf{E}_1\|_{\text{F}}^2 + 80 \|\mathbf{E}_1\|_{\text{F}}^2.$$

By our choice of $\mathbf{Z}$, $\|\mathbf{E}_1\|_{\text{F}}^2 \leq \frac{10}{9} \|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}^2$. Hence, with probability at least 0.8,

$$\|\mathbf{A} - \mathbf{C}_1 \mathbf{C}_1^\dagger \mathbf{A}\|_{\text{F}}^2 \leq 90 \|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}^2.$$

■

Lemma 54 gives us a way to find $4k$ columns providing an $O(1)$-approximation. We would like to refine this approximation to a $(1+\varepsilon)$-approximation using only an additional $O(k/\varepsilon)$ number of columns. To do so, we perform a type of residual sampling from this $O(1)$-approximation, as described in the next section.

4.2.2 Adaptive sampling

Given $O(k)$ columns providing a constant factor approximation, we can sample $O(k/\varepsilon)$ additional columns from their "residual" to obtain a $(1+\varepsilon)$-approximation. This was shown in the following lemma of Deshpande, Rademacher, Vempala, and Wang. It is actually more general in the sense that the matrix $\mathbf{V}$ in the statement of the theorem need not be a subset of columns of $\mathbf{A}$.

Theorem 55 (Theorem 2.1 of [35]) *Given $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{V} \in \mathbb{R}^{m \times c_1}$ (with $c_1 \leq n, m$), define the residual*

$$\mathbf{B} = \mathbf{A} - \mathbf{V}\mathbf{V}^\dagger \mathbf{A} \in \mathbb{R}^{m \times n}.$$

For $i = 1, \ldots, n$, and some fixed constant $\alpha > 0$, let p_i be a probability distribution such that for each i :

$$p_i \geq \alpha \|\mathbf{b}_i\|_2^2 / \|\mathbf{B}\|_{\mathrm{F}}^2,$$

where $\mathbf{b}_i$ is the i-th column of the matrix $\mathbf{B}$. Sample c_2 columns from $\mathbf{A}$ in c_2 i.i.d. trials, where in each trial the i-th column is chosen with probability p_i. Let $\mathbf{C}_2 \in \mathbb{R}^{m \times c_2}$ contain the c_2 sampled columns and let $\mathbf{C} = [\mathbf{V} \;\; \mathbf{C}_2] \in \mathbb{R}^{m \times (c_1 + c_2)}$ contain the columns of $\mathbf{V}$ and $\mathbf{C}_2$. Then, for any integer $k > 0$,

$$\mathbb{E}\left[\|\mathbf{A} - \Pi_{\mathbf{C},k}^{\mathrm{F}}(\mathbf{A})\|_{\mathrm{F}}^2\right] \leq \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2 + \frac{k}{\alpha \cdot c_2} \|\mathbf{A} - \mathbf{V}\mathbf{V}^{\dagger}\mathbf{A}\|_{\mathrm{F}}^2.$$

We denote this procedure as

$$\mathbf{C}_2 = AdaptiveCols(\mathbf{A}, \mathbf{V}, \alpha, c_2).$$

Given $\mathbf{A}$ and $\mathbf{V}$, the above algorithm requires $O(c_1 mn + c_2 \log c_2)$ arithmetic operations to find $\mathbf{C}_2$.

Rather than prove Theorem 55 directly, we will prove the following theorem of Wang and Zhang which generalizes the above theorem. One can think of the following theorem as analyzing the deviations of $\mathbf{A}$ from an arbitrary space - the row space of $\mathbf{R}$, that occur via sampling additional columns according to the residual from a given set of columns. These columns may have nothing to do with $\mathbf{R}$. This therefore generalizes the result of Deshpande and Vempala which considered $\mathbf{R}$ to be the top k right singular vectors of $\mathbf{A}$ (we will make this generalization precise below).

Theorem 56 (Theorem 4 in [122]) *Given $\mathbf{A} \in \mathbb{R}^{m \times n}$ and a matrix $\mathbf{R} \in \mathbb{R}^{r \times n}$ such that*

$$\mathrm{rank}(\mathbf{R}) = \mathrm{rank}(\mathbf{A}\mathbf{R}^{\dagger}\mathbf{R}) = \rho,$$

with $\rho \leq r \leq n$, we let $\mathbf{C}_1 \in \mathbb{R}^{m \times c_1}$ consist of c_1 columns of $\mathbf{A}$ and define the residual

$$\mathbf{B} = \mathbf{A} - \mathbf{C}_1 {\mathbf{C}_1}^{\dagger} \mathbf{A} \in \mathbb{R}^{m \times n}.$$

For $i = 1, \ldots, n$ let p_i be a probability distribution such that for each i :

$$p_i \geq \alpha \|\mathbf{b}_i\|_2^2 / \|\mathbf{B}\|_{\mathrm{F}}^2,$$

where $\mathbf{b}_i$ is the i-th column of $\mathbf{B}$. Sample c_2 columns from $\mathbf{A}$ in c_2 i.i.d. trials, where in each trial the i-th column is chosen with probability p_i. Let

$\mathbf{C}_2 \in \mathbb{R}^{m \times c_2}$ *contain the* c_2 *sampled columns and let* $\mathbf{C} = [\mathbf{C}_1, \mathbf{C}_2] \in \mathbb{R}^{m \times c_2}$. *Then,*

$$\mathbb{E}\left[\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2\right] \leq \|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 + \frac{\rho}{\alpha c_2}\|\mathbf{A} - \mathbf{C}_1{\mathbf{C}_1}^\dagger\mathbf{A}\|_{\mathrm{F}}^2.$$

We denote this procedure as

$$\mathbf{R}_2 = AdaptiveCols(\mathbf{A}, \mathbf{V}, \mathbf{R}_1, \alpha, c_2).$$

Given $\mathbf{A}$, $\mathbf{R}$, $\mathbf{C}_1$, *the above algorithm requires* $O(c_1 mn + c_2 \log c_2)$ *arithmetic operations to find* $\mathbf{C}_2$.

Proof: Write $\mathbf{A}\mathbf{R}^\dagger\mathbf{R}$ in its SVD as $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. The key to the proof is to define the following matrix

$$\mathbf{F} = \left(\sum_{q=1}^{\rho} \sigma_q^{-1}\mathbf{w}_q\mathbf{u}_q^T\right)\mathbf{A}\mathbf{R}^\dagger\mathbf{R},$$

where σ_q is the q-th singular value of $\mathbf{A}\mathbf{R}^\dagger\mathbf{R}$ with corresponding left singular vector $\mathbf{u}_q$. The $\mathbf{w}_q \in \mathbb{R}^m$ are random column vectors which will depend on the sampling and have certain desirable properties described below.

To analyze the expected error of the algorithm with respect to the choices made in the sampling procedure, we have

$$\begin{aligned}\mathbf{E}\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 &= \mathbf{E}\|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R} + \mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 \\ &= \mathbf{E}\|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 \\ &\quad + \mathbf{E}\|\mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2. \qquad (36)\end{aligned}$$

where the second equality uses the Pythagorean theorem.

One property of the $\mathbf{w}_q$ we will ensure below is that the $\mathbf{w}_q$ each lie in the span of the columns of $\mathbf{C}$. Given this, we have

$$\begin{aligned}\|\mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 &\leq \|\mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{W}\mathbf{W}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 \\ &\leq \|\mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{F}\|_{\mathrm{F}}^2. \qquad (37)\end{aligned}$$

Plugging (37) into (36), we have

$$\mathbf{E}[\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2] \leq \|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 + \mathbf{E}\left[\|\mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{F}\|_{\mathrm{F}}^2\right], \qquad (38)$$

where note that $\mathbf{R}$ is deterministic so we can remove the expectation.

Let $\mathbf{v}_1, \ldots, \mathbf{v}_\rho$ be the right singular vectors of $\mathbf{AR}^\dagger\mathbf{R}$. As both the rows of $\mathbf{AR}^\dagger\mathbf{R}$ and of $\mathbf{F}$ lie in the span of $\mathbf{v}_1, \ldots, \mathbf{v}_\rho$, we can decompose (38) as follows:

$$\begin{aligned}
\mathbf{E}[\|\mathbf{A} - \mathbf{V}\mathbf{V}^\dagger\mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2] &\leq \|\mathbf{A} - \mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 && (39)\\
&+ \mathbf{E}\left[\|\mathbf{AR}^\dagger\mathbf{R} - \mathbf{F}\|_{\mathrm{F}}^2\right]\\
&\leq \|\mathbf{A} - \mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 && (40)\\
&+ \sum_{j=1}^{\rho} \mathbf{E}\|(\mathbf{AR}^\dagger\mathbf{R} - \mathbf{F})\mathbf{v}_j\|_2^2\\
&\leq \|\mathbf{A} - \mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2\\
&+ \sum_{j=1}^{\rho} \mathbf{E}\|\mathbf{AR}^\dagger\mathbf{R}\mathbf{v}_j - \sum_{q=1}^{\rho} \sigma_q^{-1}\mathbf{w}_q\mathbf{u}_q^T\sigma_j\mathbf{u}_j\|_2^2\\
&= \|\mathbf{A} - \mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 && (41)\\
&+ \sum_{j=1}^{\rho} \mathbf{E}\|\mathbf{AR}^\dagger\mathbf{R}\mathbf{v}_j - \mathbf{w}_j\|_2^2\\
&= \|\mathbf{A} - \mathbf{AR}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 && (42)\\
&+ \sum_{j=1}^{\rho} \mathbf{E}\|\mathbf{A}\mathbf{v}_j - \mathbf{w}_j\|_2^2, && (43)
\end{aligned}$$

where the final equality follows from the fact that $\mathbf{v}_j$ is, by definition, in the row space of $\mathbf{R}$, and so

$$\mathbf{A}\mathbf{v}_j - \mathbf{AR}^\dagger\mathbf{R}\mathbf{v}_j = \mathbf{A}(\mathbf{I} - \mathbf{R}^\dagger\mathbf{R})\mathbf{v}_j = \mathbf{0}.$$

Looking at (39), it becomes clear what the properties of the $\mathbf{w}_j$ are that we want. Namely, we want them to be in the column space of $\mathbf{V}$ and to have the property that $\mathbf{E}\|\mathbf{A}\mathbf{v}_j - \mathbf{w}_j\|_2^2$ is as small as possible.

To define the $\mathbf{w}_j$ vectors, we begin by defining auxiliary random variables $\mathbf{x}_{j,(\ell)} \in \mathbb{R}^m$, for $j = 1, \ldots, \rho$ and $\ell = 1, \ldots, c_2$:

$$\mathbf{x}_{j,(\ell)} = \frac{\mathbf{v}_{i,j}}{p_i}\mathbf{b}_i = \frac{\mathbf{v}_{i,j}}{p_i}\left(\mathbf{a}_i - \mathbf{C}_1\mathbf{C}_1^\dagger\mathbf{a}_i\right),$$

with probability p_i, for $i = 1, \ldots, n$. We have that $\mathbf{x}_{j,(\ell)}$ is a deterministic linear combination of a random column sampled from the distribution defined in the theorem statement. Moreover,

$$\mathbf{E}[\mathbf{x}_{j,(\ell)}] = \sum_{i=1}^{n} p_i \frac{\mathbf{v}_{i,j}}{p_i}\mathbf{b}_i = \mathbf{B}\mathbf{v}_j,$$

and

$$\mathbf{E}\|\mathbf{x}_{j,(\ell)}\|_2^2 \;=\; \sum_{i=1}^{n} p_i \frac{\mathbf{v}_{i,j}^2}{p_i^2}\|\mathbf{b}_i\|_2^2 \leq \sum_{i=1}^{n} \frac{\mathbf{v}_{i,j}^2}{\alpha\|\mathbf{b}_i\|_2^2/\|\mathbf{B}\|_{\mathrm{F}}^2}\|\mathbf{b}_i\|_2^2 = \frac{\|\mathbf{B}\|_{\mathrm{F}}^2}{\alpha}.$$

We now define the average vector

$$\mathbf{x}_j = \frac{1}{c_2}\sum_{\ell=1}^{c_2}\mathbf{x}_{j,(\ell)},$$

and we have

$$\mathbf{E}[\mathbf{x}_j] \;=\; \mathbf{E}[\mathbf{x}_{j,(\ell)}] = \mathbf{B}\mathbf{v}_j,$$

and

$$\begin{aligned}\mathbf{E}\|\mathbf{x}_j - \mathbf{B}\mathbf{v}_j\|_2^2 &= \mathbf{E}\|\mathbf{x}_j - \mathbf{E}[\mathbf{x}_j]\|_2^2 \\ &= \frac{1}{c_2}\mathbf{E}\|\mathbf{x}_{j,(\ell)} - \mathbf{E}[\mathbf{x}_{j,(\ell)}]\|_2^2 \qquad (44)\\ &= \frac{1}{c_2}\mathbf{E}\|\mathbf{x}_{j,(\ell)} - \mathbf{B}\mathbf{v}_j\|_2^2,\end{aligned}$$

where (44) follows from the fact that the samples are independent. In fact, pairwise independence suffices for this statement, which we shall use in our later derandomization of this theorem.

Notice that $\mathbf{x}_j$ is in the span of the columns of $\mathbf{C}$, for every $j = 1, \ldots, n$ (note that while for fixed j, the $\mathbf{x}_{j,(\ell)}$ are pairwise independent, for two different j, j' we have that $\mathbf{x}_j$ and $\mathbf{x}_{j'}$ are dependent).

For $j = 1, \ldots, \rho$, we now define

$$\mathbf{w}_j = \mathbf{C}_1\mathbf{C}_1^{\dagger}\mathbf{A}\mathbf{v}_j + \mathbf{x}_j, \qquad (45)$$

and we also have that $\mathbf{w}_1, \ldots, \mathbf{w}_\rho$ are in the column space of $\mathbf{C}$, as required above. It remains to bound $\mathbf{E}\|\mathbf{w}_j - \mathbf{A}\mathbf{v}_j\|_2^2$ as needed for (39).

We have

$$\begin{aligned}\mathbf{E}[\mathbf{w}_j] &= \mathbf{C}_1\mathbf{C}_1^{\dagger}\mathbf{A}\mathbf{v}_j + \mathbf{E}[\mathbf{x}_j] \\ &= \mathbf{C}_1\mathbf{C}_1^{\dagger}\mathbf{A}\mathbf{v}_j + \mathbf{B}\mathbf{v}_j \\ &= \mathbf{A}\mathbf{v}_j, \qquad (46)\end{aligned}$$

where (46) together with (45) imply that

$$\mathbf{w}_j - \mathbf{A}\mathbf{v}_j = \mathbf{x}_j - \mathbf{B}\mathbf{v}_j.$$

At long last we have

$$\begin{aligned}
\mathbf{E}\|\mathbf{w}_j - \mathbf{A}\mathbf{v}_j\|_2^2 &= \mathbf{E}\|\mathbf{x}_j - \mathbf{B}\mathbf{v}_j\|_2^2 \\
&= \frac{1}{c_2}\mathbf{E}\|\mathbf{x}_{j,(\ell)} - \mathbf{B}\mathbf{v}_j\|_2^2 \\
&= \frac{1}{c_2}\mathbf{E}\|\mathbf{x}_{j,(\ell)}\|_2^2 - \frac{2}{c_2}(\mathbf{B}\mathbf{v}_j)^T\mathbf{E}[\mathbf{x}_{j,(\ell)}] + \frac{1}{c_2}\|\mathbf{B}\mathbf{v}_j\|_2^2 \\
&= \frac{1}{c_2}\mathbf{E}\|\mathbf{x}_{j,(\ell)}\|_2^2 - \frac{1}{c_2}\|\mathbf{B}\mathbf{v}_j\|_2^2 \\
&\leq \frac{1}{\alpha c_2}\|\mathbf{B}\|_{\mathrm{F}}^2 - \frac{1}{c_2}\|\mathbf{B}\mathbf{v}_j\|_2^2 \\
&\leq \frac{1}{\alpha c_2}\|\mathbf{B}\|_{\mathrm{F}}^2. \qquad (47)
\end{aligned}$$

Plugging (47) into (39), we obtain

$$\mathbf{E}[\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2] \leq \|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 + \frac{\rho}{\alpha c_2}\|\mathbf{A} - \mathbf{C}_1\mathbf{C}_1^\dagger\mathbf{A}\|_{\mathrm{F}}^2,$$

which completes the proof. ■

4.2.3 CUR wrapup

Obtaining a Good Set of Columns. We will apply Theorem 55 with the $\mathbf{V}$ of that theorem set to $\mathbf{C}_1$. For the distribution p, we need to quickly approximate the column norms of $\mathbf{B} = \mathbf{A} - \mathbf{C}_1{\mathbf{C}_1}^\dagger\mathbf{A}$. To do so, by Lemma 18 it suffices to compute $\mathbf{B}\cdot\mathbf{G}$, where $\mathbf{G}$ is an $n \times t$ matrix of i.i.d. $N(0, 1/t)$ random variables, for $t = O(\log n)$. By Lemma 18, with probability at least $1 - 1/n$, simultaneously for all $i \in [n]$,

$$\frac{\|\mathbf{b}_i\|_2^2}{2} \leq \|(\mathbf{B}\mathbf{G})_{*i}\|_2^2 \leq \frac{3}{2}\|\mathbf{b}_i\|_2^2,$$

where $\mathbf{b}_i = \mathbf{B}_{*i}$ is the i-th column of $\mathbf{B}$. It follows that we can set $\alpha = \frac{1}{3}$ in Theorem 55 using the distribution p on $[n]$ given by

$$\forall i \in [n],\ p_i = \frac{\|(\mathbf{B}\mathbf{G})_{*i}\|_2^2}{\|\mathbf{B}\mathbf{G}\|_{\mathrm{F}}^2}.$$

Hence, for a parameter $c_2 > 0$, if we set

$$\mathbf{C}_2 = \mathit{AdaptiveCols}(\mathbf{A}, \mathbf{C}_1, \frac{1}{3}, c_2),$$

if $\mathbf{C} = [\mathbf{C}_1, \mathbf{C}_2]$, where $\mathbf{C}_2$ are the columns sampled by $AdaptiveCols(\mathbf{A}, \mathbf{C}_1, \frac{1}{3}, c_2)$, then by the conclusion of Theorem 55,

$$\mathbb{E}\left[\|\mathbf{A} - \Pi^{\mathrm{F}}_{\mathbf{C},k}(\mathbf{A})\|_{\mathrm{F}}^2\right] \leq \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2 + \frac{3k}{c_2}\|\mathbf{A} - \mathbf{C}_1\mathbf{C}_1^{\dagger}\mathbf{A}\|_{\mathrm{F}}^2.$$

By Lemma 54, with probability at least .8, $\|\mathbf{A} - \mathbf{C}_1\mathbf{C}_1^{\dagger}\mathbf{A}\|_{\mathrm{F}}^2 \leq 90\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2$, which we condition on. It follows by setting $c_2 = 270k/\varepsilon$, then taking expectations with respect to the randomness in $AdaptiveCols(\mathbf{A}, \mathbf{C}_1, \frac{1}{3}, c_2)$, we have

$$\mathbb{E}\left[\|\mathbf{A} - \Pi^{\mathrm{F}}_{\mathbf{C},k}(\mathbf{A})\|_{\mathrm{F}}^2\right] \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2.$$

Running Time. A few comments on the running time are in order. We can compute $\mathbf{Z}$ in $O(\mathrm{nnz}(A)) + (m+n)\cdot\mathrm{poly}(k)$ time via Theorem 46. Given $\mathbf{Z}$, we can run RandSampling$(\mathbf{Z}, s, p)$, where $s = O(k\log k)$ and p is the leverage score distribution defined by $\mathbf{Z}$. This can be done in $n\cdot\mathrm{poly}(k/\varepsilon)$ time.

We then run BssSamplingSparse$(\mathbf{V}_{\mathbf{M}}, (\mathbf{A} - \mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^T)^T\mathbf{\Omega}_1\mathbf{D}_1, 4k, .5)$. To do this efficiently, we can't afford to explicitly compute the matrix $(\mathbf{A} - \mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^T)^T\mathbf{\Omega}_1\mathbf{D}_1$. We only form $\mathbf{A}\mathbf{\Omega}\mathbf{D}$ and $\mathbf{Z}_1^{\mathrm{T}}\mathbf{\Omega}\mathbf{D}$ in $O(\mathrm{nnz}(A))+n\cdot\mathrm{poly}(k)$ time. Then, BssSamplingSparse multiplies $\left(\mathbf{A} - \mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^{\mathrm{T}}\right)\mathbf{\Omega}\mathbf{D}$ from the left with a sparse subspace embedding matrix $\mathbf{W} \in \mathbb{R}^{\xi\times m}$ with $\xi = O(k^2\log^2 k)$. Computing $\mathbf{W}\mathbf{A}$ takes $O\left(\mathrm{nnz}(\mathbf{A})\right)$ time. Then, computing $(\mathbf{W}\mathbf{A})\mathbf{Z}_1$ and $(\mathbf{W}\mathbf{A}\mathbf{Z}_1)\mathbf{Z}_1^{\mathrm{T}}$ takes another $O(\xi mk) + O(\xi nk)$ time, respectively. Finally, the sampling algorithm on $\mathbf{W}(\mathbf{A} - \mathbf{A}\mathbf{Z}_1\mathbf{Z}_1^{\mathrm{T}})\mathbf{\Omega}\mathbf{D}$ is $O(k^4\log k + mk\log k)$ time.

Given $\mathbf{A}, \mathbf{Z}_1, \mathbf{\Omega}, \mathbf{D}$ and $\mathbf{S}_1$ we then know the matrix $\mathbf{C}_1$ needed to run $AdaptiveCols(\mathbf{A}, \mathbf{C}_1, \frac{1}{3}, c_2)$. The latter algorithm samples columns of $\mathbf{A}$, which can be done in $O(\mathrm{nnz}(A))+nk/\varepsilon$ time given the distribution p to sample from. Here to find p we need to compute $\mathbf{B}\cdot\mathbf{G}$, where $\mathbf{B} = \mathbf{A} - \mathbf{C}_1{\mathbf{C}_1}^{\dagger}\mathbf{A}$ and $\mathbf{G}$ is an $n \times O(\log n)$ matrix. We can compute this matrix product in time $O(\mathrm{nnz}(A)\log n) + (m+n)\mathrm{poly}(k/\varepsilon)$.

It follows that the entire procedure to find $\mathbf{C}$ is $O(\mathrm{nnz}(A)\log n) + (m+n)\mathrm{poly}(k/\varepsilon)$ time.

Simultaneously Obtaining a Good Set of Rows. At this point we have a set $\mathbf{C}$ of $O(k/\varepsilon)$ columns of $\mathbf{A}$ for which

$$\mathbb{E}\left[\|\mathbf{A} - \Pi^{\mathrm{F}}_{\mathbf{C},k}(\mathbf{A})\|_{\mathrm{F}}^2\right] \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2.$$

If we did not care about running time, we could now find the best k-dimensional subspace of the columns of $\mathbf{C}$ for approximating the column

space of $\mathbf{A}$, that is, if $\mathbf{U}$ has orthonormal columns with the same column space as $\mathbf{C}$, then by Lemma 44,

$$\mathbb{E}\left[\|\mathbf{A}-\mathbf{U}[\mathbf{U}^T\mathbf{A}]_k\|_{\mathrm{F}}^2\right] \leq (1+\varepsilon)\|\mathbf{A}-\mathbf{A}_k\|_{\mathrm{F}}^2,$$

where $[\mathbf{U}^T\mathbf{A}]_k$ denotes the best rank-k approximation to $\mathbf{U}^T\mathbf{A}$ in Frobenius norm. So if $\mathbf{L}$ is an $m \times k$ matrix with orthonormal columns with the same column space as $\mathbf{U}[\mathbf{U}^T\mathbf{A}]_k$, we could then attempt to execute the analogous algorithm to the one that we just ran. That algorithm was for finding a good set of columns $\mathbf{C}$ starting with $\mathbf{Z}$, and now we would like to find a good set $\mathbf{R}$ of rows starting with $\mathbf{L}$. This is the proof strategy used by Boutsidis and the author in [21].

Indeed, the algorithm of [21] works by first sampling $O(k \log k)$ rows of $\mathbf{A}$ according to the leverage scores of $\mathbf{L}$. It then downsamples this to $O(k)$ rows using BssSamplingSparse. Now, instead of using Theorem 55, the algorithm invokes Theorem 56, applied to $\mathbf{A}^T$, to find $O(k/\varepsilon)$ rows.

Applying Theorem 56 to $\mathbf{A}^T$, the error has the form:

$$\mathbf{E}\|\mathbf{A}-\mathbf{V}\mathbf{V}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_{\mathrm{F}}^2 \leq \|\mathbf{A}-\mathbf{V}\mathbf{V}^\dagger\mathbf{A}\|_{\mathrm{F}}^2 + \frac{\rho}{r_2}\|\mathbf{A}-\mathbf{A}\mathbf{R}_1^\dagger\mathbf{R}_1\|_{\mathrm{F}}^2 \qquad (48)$$

where ρ is the rank of $\mathbf{V}$. Note that had we used $\mathbf{U}$ here in place of $\mathbf{L}$, ρ could be $\Theta(k/\varepsilon)$, and then the number r_2 of samples we would need in (48) would be $\Theta(k/\varepsilon^2)$, which is more than the $O(k/\varepsilon)$ columns and $O(k/\varepsilon)$ rows we could simultaneously hope for. It turns out that these procedures can also be implemented in $O(\mathrm{nnz}(\mathbf{A}))\log n + (m+n)\mathrm{poly}(k/\varepsilon)$ time.

We glossed over the issue of how to find the best k-dimensional subspace $\mathbf{L}$ of the columns of $\mathbf{C}$ for approximating the column space of $\mathbf{A}$, as described above. Naïvely doing this would involve projecting the columns of $\mathbf{A}$ onto the column space of $\mathbf{C}$, which is too costly. Fortunately, by Theorem 60 in §4.4, in $O(\mathrm{nnz}(\mathbf{A})) + (m+n)\mathrm{poly}(k/\varepsilon)$ time it is possible to find an $m \times k$ matrix $\mathbf{L}'$ with orthonormal columns so that

$$\|\mathbf{A}-\mathbf{L}'(\mathbf{L}')^T\mathbf{A}\|_{\mathrm{F}} \leq (1+\varepsilon)\|\mathbf{A}-\mathbf{L}\mathbf{L}^T\mathbf{A}\|_{\mathrm{F}}.$$

Indeed, Theorem 60 implies that if $\mathbf{W}$ is an ℓ_2-subspace embedding, then we can take $\mathbf{L}'$ to be the top k left singular vectors of $\mathbf{U}\mathbf{U}^T\mathbf{A}\mathbf{W}$, and since $\mathbf{U}\mathbf{U}^T$ has rank $O(k/\varepsilon)$, this matrix product can be computed in $O(\mathrm{nnz}(\mathbf{A}))+(m+n)\cdot\mathrm{poly}(k/\varepsilon)$ time using sparse subspace embeddings. We can thus use $\mathbf{L}'$ in place of $\mathbf{L}$ in the algorithm for selecting a subset of $O(k/\varepsilon)$ rows of $\mathbf{A}$.

Finding a U With Rank k. The above outline shows how to simultaneously obtain a matrix $\mathbf{C}$ and a matrix $\mathbf{R}$ with $O(k/\varepsilon)$ columns and rows, respectively. Given such a $\mathbf{C}$ and a $\mathbf{R}$, we need to find a rank-k matrix $\mathbf{U}$ which is the minimizer to the problem

$$\min_{\text{rank}-k\mathbf{U}} \|\mathbf{A} - \mathbf{CUR}\|_{\text{F}}.$$

We are guaranteed that there is such a rank-k matrix $\mathbf{U}$ since crucially, when we apply Theorem 56, we apply it with $\mathbf{V} = \mathbf{L}$, which has rank k. Therefore, the resulting approximation $\mathbf{VV}^{\dagger}\mathbf{AR}^{\dagger}\mathbf{R}$ is a rank-k matrix, and since $\mathbf{L}$ is in the span of $\mathbf{C}$, can be expressed as $\mathbf{CUR}$. It turns out one can quickly find $\mathbf{U}$, as shown in [21]. We omit the details.

Deterministic CUR Decomposition. The main idea in [21] to achieve a CUR Decomposition with the same $O(k/\varepsilon)$ columns and rows and a rank-k matrix $\mathbf{U}$ deterministically is to derandomize Theorem 55 and Theorem 56. The point is that the proofs involve the second moment method, and therefore by a certain discretization of the sampling probabilities, one can derandomize the algorithm using pairwise-independent samples (of either columns or rows, depending on whether one is derandomizing Theorem 55 or Theorem 56). This increases the running time when applied to an $n \times n$ matrix $\mathbf{A}$ to $n^4 \cdot \text{poly}(k/\varepsilon)$, versus, say, n^3 using other deterministic algorithms such as the SVD, but gives an actual subset of rows and columns.

4.3 Spectral norm error

Here we show how to quickly obtain a $(1+\varepsilon)$ rank-k approximation with respect to the spectral norm $\|\mathbf{A}\|_2 = \sup_{\mathbf{x}} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2}$. That is, given an $m \times n$ matrix $\mathbf{A}$, compute a rank-k matrix $\tilde{\mathbf{A}}_k$, where $\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2$.

It is well-known that $\|\mathbf{A}-\mathbf{A}_k\|_2 = \sigma_{k+1}(\mathbf{A})$, where $\sigma_{k+1}(\mathbf{A})$ is the $(k+1)$-st singular value of $\mathbf{A}$, and that $\mathbf{A}_k$ is the matrix $\mathbf{U}\mathbf{\Sigma}_k\mathbf{V}^T$, where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is the SVD of $\mathbf{A}$ and $\mathbf{\Sigma}_k$ is a diagonal matrix with first k diagonal entries equal to those of $\mathbf{\Sigma}$, and 0 otherwise.

Below we present an algorithm, proposed by Halko, Martinsson and Tropp [59], that was shown by the authors to be a bicriteria rank-k approximation. That is, they efficiently find an $n \times 2k$ matrix $\mathbf{Z}$ with orthonormal columns for which $\|\mathbf{A}-\mathbf{ZZ}^T\mathbf{A}\|_2 \leq (1+\varepsilon)\|\mathbf{A}-\mathbf{A}_k\|_2$. By slightly modifying their analysis, this matrix $\mathbf{Z}$ can be shown to have dimensions $n \times (k+4)$ with the same error guarantee. The analysis of this algorithm was somewhat

simplified by Boutsidis, Drineas, and Magdon-Ismail [19], and by slightly modifying their analysis, this results in an $n \times (k+2)$ matrix $\mathbf{Z}$ with orthonormal columns for which $\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2$. We follow the analysis of [19], but simplify and improve it slightly in order to output a true rank-k approximation, that is, an $n \times k$ matrix $\mathbf{Z}$ with orthonormal columns for which $\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2$. This gives us a new result which has not appeared in the literature to the best of our knowledge.

Before presenting the algorithm, we need the following lemma. Suppose we have an $n \times k$ matriz $\mathbf{Z}$ with orthonormal columns for which there exists an $\mathbf{X}$ for which $\|\mathbf{A} - \mathbf{Z}\mathbf{X}\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2$. How do we find such an $\mathbf{X}$? It turns out the optimal such $\mathbf{X}$ is equal to $\mathbf{Z}^T\mathbf{A}$.

Lemma 57 *If we let $\mathbf{X}^* = argmin_{\mathbf{X}}\|\mathbf{A} - \mathbf{Z}\mathbf{X}\|_2$, then $\mathbf{X}^*$ satisfies $\mathbf{Z}\mathbf{X}^* = \mathbf{Z}\mathbf{Z}^T\mathbf{A}$.*

Proof: On the one hand, $\|\mathbf{A} - \mathbf{Z}\mathbf{X}^*\|_2 \leq \|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2$, since $\mathbf{X}^*$ is the minimizer. On the other hand, for any vector $\mathbf{v}$, by the Pythagorean theorem,

$$\begin{aligned} \|(\mathbf{A} - \mathbf{Z}\mathbf{X}^*)\mathbf{v}\|_2^2 &= \|(\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A})\mathbf{v}\|_2^2 + \|(\mathbf{Z}\mathbf{Z}^T\mathbf{A} - \mathbf{Z}\mathbf{X}^*)\mathbf{v}\|_2^2 \\ &\geq \|(\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A})\mathbf{v}\|_2^2, \end{aligned}$$

and so $\|\mathbf{A} - \mathbf{Z}\mathbf{X}^*\|_2 \geq \|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2$. ■

We also collect a few facts about the singular values of a Gaussian matrix.

Fact 6 *(see, e.g., [104]) Let $\mathbf{G}$ be an $r \times s$ matrix of i.i.d. normal random variables with mean 0 and variance 1. There exist constants $C, C' > 0$ for which*

(1) The maximum singular value $\sigma_1(\mathbf{G})$ satisfies $\sigma_1(\mathbf{G}) \leq C\sqrt{\max(r,s)}$ with probability at least 9/10.

(2) If $r = s$, then the minimum singular value $\sigma_r(\mathbf{G})$ satisfies $\sigma_r(\mathbf{G}) \geq C'\sqrt{r}$ with probability at least 9/10

The algorithm, which we call SubspacePowerMethod is as follows. The intuition is, like the standard power method, if we compute $(\mathbf{A}\mathbf{A}^T)^q\mathbf{A}\mathbf{g}$ for a random vector $\mathbf{g}$, then for large enough q this very quickly converges to the top left singular vector of $\mathbf{A}$. If we instead compute $(\mathbf{A}\mathbf{A}^T)^q\mathbf{A}\mathbf{G}$ for a random $n \times k$ matrix $\mathbf{G}$, for large enough q this also very quickly converges to an $n \times k$ matrix which is close, in a certain sense, to the top k left singular vectors of $\mathbf{A}$.

1. Compute $\mathbf{B} = (\mathbf{A}\mathbf{A}^T)^q\mathbf{A}$ and $\mathbf{Y} = \mathbf{B}\mathbf{G}$, where $\mathbf{G}$ is an $n \times k$ matrix of i.i.d. $N(0,1)$ random variables.

2. Let $\mathbf{Z}$ be an $n \times k$ matrix with orthonormal columns whose column space is equal to that of $\mathbf{Y}$.

3. Output $\mathbf{Z}\mathbf{Z}^T\mathbf{A}$.

In order to analyze SubspacePowerMethod, we need a key lemma shown in [59] concerning powering of a matrix.

Lemma 58 *Let $\mathbf{P}$ be a projection matrix, i.e., $\mathbf{P} = \mathbf{Z}\mathbf{Z}^T$ for a matrix $\mathbf{Z}$ with orthonormal columns. For any matrix $\mathbf{X}$ of the appropriate dimensions and integer $q \geq 0$,*

$$\|\mathbf{P}\mathbf{X}\|_2 \leq (\|\mathbf{P}(\mathbf{X}\mathbf{X}^T)^q\mathbf{X}\|_2)^{1/(2q+1)}.$$

Proof: Following [59], we first show that if $\mathbf{R}$ is a projection matrix and $\mathbf{D}$ a non-negative diagonal matrix, then $\|\mathbf{R}\mathbf{D}\mathbf{R}\|_2^t \leq \|\mathbf{R}\mathbf{D}^t\mathbf{R}\|_2$. To see this, suppose $\mathbf{x}$ is a unit vector for which $\mathbf{x}^T\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{x} = \|\mathbf{R}\mathbf{D}\mathbf{R}\|_2$. We can assume that $\|\mathbf{R}\mathbf{x}\|_2 = 1$, as otherwise since $\mathbf{R}$ is a projection matrix, $\|\mathbf{R}\mathbf{x}\|_2 < 1$, and taking the unit vector $\mathbf{z} = \mathbf{R}\mathbf{x}/\|\mathbf{R}\mathbf{x}\|_2$, we have

$$\mathbf{z}^T\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{z} = \frac{\mathbf{x}\mathbf{R}^2\mathbf{D}\mathbf{R}^2\mathbf{x}}{\|\mathbf{R}\mathbf{x}\|_2^2} = \frac{\mathbf{x}^T\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{x}}{\|\mathbf{R}\mathbf{x}\|_2^2} > \mathbf{x}^T\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{x},$$

contradicting that $\mathbf{x}^t\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{x} = \|\mathbf{R}\mathbf{D}\mathbf{R}\|_2$. We thus have,

$$\begin{aligned}
\|\mathbf{R}\mathbf{D}\mathbf{R}\|^t &= (\mathbf{x}^T\mathbf{R}\mathbf{D}\mathbf{R}\mathbf{x})^t = (\mathbf{x}^T\mathbf{D}\mathbf{x})^t = (\sum_j \mathbf{D}_{j,j}\mathbf{x}_j^2)^t \\
&\leq \sum_j \mathbf{D}_{j,j}^t\mathbf{x}_j^2 = \mathbf{x}^T\mathbf{D}^t\mathbf{x} = (\mathbf{R}\mathbf{x})^T\mathbf{D}^t\mathbf{R}\mathbf{x} \\
&\leq \|\mathbf{R}\mathbf{D}^t\mathbf{R}\|_2,
\end{aligned}$$

where we have used Jensen's inequality to show that $(\sum_j \mathbf{D}_{j,j}\mathbf{x}_j^2)^t \leq \sum_j \mathbf{D}_{j,j}^t\mathbf{x}_j^2$, noting that $\sum_j \mathbf{x}_j^2 = 1$ and the function $z \to |z|^t$ is convex.

Given this claim, let $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be a decomposition of $\mathbf{X}$ in which $\mathbf{U}$ and $\mathbf{V}^T$ are square matrices with orthonormal columns and rows, and Σ has non-negative entries on the diagonal (such a decomposition can be obtained

from the SVD). Then,

$$\begin{aligned}
\|\mathbf{PX}\|_2^{2(2q+1)} &= \|\mathbf{PXX}^T\mathbf{P}\|_2^{2q+1} \\
&= \|(\mathbf{U}^T\mathbf{PU})\mathbf{\Sigma}^2(\mathbf{U}^T\mathbf{PU})\|_2^{2q+1} \\
&\leq \|(\mathbf{U}^T\mathbf{PU})\mathbf{\Sigma}^{2(2q+1)}(\mathbf{U}^T\mathbf{PU})\|_2 \\
&= \|\mathbf{P}(\mathbf{XX}^T)^{(2q+1)}\mathbf{P}\|_2 \\
&= \|\mathbf{P}(\mathbf{XX}^T)^q\mathbf{XX}^T(\mathbf{XX}^T)^q\mathbf{P}\|_2 \\
&= \|\mathbf{P}(\mathbf{XX}^T)^q\mathbf{X}\|_2^2,
\end{aligned}$$

where the first equality follows since $\|\mathbf{PX}\|_2^2 = \|\mathbf{PXX}^T\mathbf{P}\|_2$, the second equality uses that $\mathbf{XX}^T = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^T$ and rotational invariance given that $\mathbf{U}$ has orthonormal rows and columns, the first inequality uses the claim above with $\mathbf{R} = \mathbf{U}^T\mathbf{PU}$, the next equality uses that $\mathbf{XX}^T = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^T$, the next equality regroups terms, and the final equality writes the operator norm as the equivalent squared operator norm.

If we raise both sides to the $1/(2(2q+1))$-th power, then this completes the proof. ■

We can now prove the main theorem about SubspacePowerMethod

Theorem 59 *For appropriate $q = O(\log(mn)/\varepsilon)$, with probability at least 4/5,* SubspacePowerMethod *outputs a rank-k matrix $\mathbf{ZZ}^T\mathbf{A}$ for which $\|\mathbf{A} - \mathbf{ZZ}^T\mathbf{A}\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2$. Note that* SubspacePowerMethod *can be implemented in $O(\text{nnz}(\mathbf{A})k\log(mn)/\varepsilon)$ time.*

Proof: By Lemma 57, $\mathbf{ZZ}^T\mathbf{A}$ is the best rank-k approximation of $\mathbf{A}$ in the column space of $\mathbf{Z}$ with respect to the spectral norm. Hence,

$$\begin{aligned}
\|\mathbf{A} - \mathbf{ZZ}^T\mathbf{A}\|_2 &\leq \|\mathbf{A} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger\mathbf{A}\|_2 \\
&= \|(\mathbf{I} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger)\mathbf{A}\|_2,
\end{aligned}$$

where the inequality follows since $\mathbf{ZZ}^T\mathbf{B}$ is of rank k and in the column space of $\mathbf{Z}$. Since $\mathbf{I} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger$ is a projection matrix, we can apply Lemma 58 to infer that $\|(\mathbf{I} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger)\mathbf{A}\|_2$ is at most $\|(\mathbf{I} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger)(\mathbf{AA}^T)^q\mathbf{A}\|_2^{1/(2q+1)}$, which is equal to

$$\begin{aligned}
&= \|\mathbf{B} - (\mathbf{ZZ}^T\mathbf{B})(\mathbf{ZZ}^T\mathbf{B})^\dagger\mathbf{B}\|_2^{1/(2q+1)} \\
&= \|\mathbf{B} - \mathbf{ZZ}^T\mathbf{B}\|_2^{1/(2q+1)},
\end{aligned}$$

where we use that $(\mathbf{Z}\mathbf{Z}^T\mathbf{B})^\dagger = (\mathbf{Z}^T\mathbf{B})^\dagger\mathbf{Z}^T$ since $\mathbf{Z}$ has orthonormal columns, and thus

$$(\mathbf{Z}\mathbf{Z}^T\mathbf{B})(\mathbf{Z}\mathbf{Z}^T\mathbf{B})^\dagger\mathbf{B} = (\mathbf{Z}\mathbf{Z}^T\mathbf{B})(\mathbf{Z}^T\mathbf{B})^\dagger(\mathbf{Z}^T\mathbf{B}) = \mathbf{Z}\mathbf{Z}^T\mathbf{B}.$$

Hence,

$$\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2 \leq \|\mathbf{B} - \mathbf{Z}\mathbf{Z}^T\mathbf{B}\|_2^{1/(2q+1)}. \tag{49}$$

Let $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be the SVD of $\mathbf{B}$. Let $\mathbf{\Omega}_U = \mathbf{V}_k^T\mathbf{G}$ and $\mathbf{\Omega}_L = \mathbf{V}_{n-k}^T\mathbf{G}$, where $\mathbf{V}_k^T$ denotes the top k rows of $\mathbf{V}^T$, and $\mathbf{V}_{n-k}^T$ the remaining $n-k$ rows. Since the rows of $\mathbf{V}^T$ are orthonormal, by rotational invariance of the Gaussian distribution, $\mathbf{\Omega}_U$ and $\mathbf{\Omega}_L$ are independent matrices of i.i.d. $N(0,1)$ entries.

We now apply Lemma 47 with the $\mathbf{C}$ of that lemma equal to $\mathbf{Z}$ above, the $\mathbf{Z}$ of that lemma equal to $\mathbf{V}_k$, and the $\mathbf{A}$ of that lemma equal to $\mathbf{B}$ above. This implies the $\mathbf{E}$ of that lemma is equal to $\mathbf{B} - \mathbf{B}_k$. Note that to apply the lemma we need $\mathbf{V}_k^T\mathbf{G}$ to have full rank, which holds with probability 1 since it is a $k \times k$ matrix of i.i.d. $N(0,1)$ random variables. We thus have,

$$\begin{aligned}
\|\mathbf{B} - \mathbf{Z}\mathbf{Z}^T\mathbf{B}\|_2^2 &\leq \|\mathbf{B} - \mathbf{B}_k\|_2^2 + \|(\mathbf{B} - \mathbf{B}_k)\mathbf{G}(\mathbf{V}_k^T\mathbf{G})^\dagger\|_2^2 \\
&= \|\mathbf{B} - \mathbf{B}_k\|_2^2 + \|\mathbf{U}_{n-k}\mathbf{\Sigma}_{n-k}\mathbf{V}_{n-k}^T\mathbf{G}(\mathbf{V}_k^T\mathbf{G})^\dagger\|_2^2 \\
&= \|\mathbf{B} - \mathbf{B}_k\|_2^2 + \|\mathbf{\Sigma}_{n-k}\mathbf{V}_{n-k}^T\mathbf{G}(\mathbf{V}_k^T\mathbf{G})^\dagger\|_2^2 \\
&\leq \|\mathbf{B} - \mathbf{B}_k\|_2^2 \left(1 + \|\mathbf{\Omega}_2\|_2^2\|\mathbf{\Omega}_1^\dagger\|_2^2\right),
\end{aligned}$$

where $\mathbf{\Sigma}_{n-k}$ denotes the $n-k \times n-k$ diagonal matrix whose entries are the bottom $n-k$ diagonal entries of $\mathbf{\Sigma}$, and $\mathbf{U}_{n-k}$ denotes the rightmost $n-k$ columns of $\mathbf{U}$. Here in the second equality we use unitary invariance of $\mathbf{U}_{n-k}$, while in the inequality we use sub-multiplicativity of the spectral norm. By Fact 6 and independence of $\mathbf{\Omega}_2$ and $\mathbf{\Omega}_1$, we have that $\|\mathbf{\Omega}_2\|_2^2 \leq C(n-k)$ and $\|\mathbf{\Omega}_1^\dagger\|_2^2 \leq \frac{1}{(C')^2 k}$ with probability at least $(9/10)^2 > 4/5$. Consequently for a constant $c > 0$,

$$\|\mathbf{B} - \mathbf{Z}\mathbf{Z}^T\mathbf{B}\|_2^2 \leq \|\mathbf{B} - \mathbf{B}_k\|_2^2 \cdot c(n-k)/k. \tag{50}$$

Combining (50) with (49), we have

$$\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\|_2 \leq \|\mathbf{B} - \mathbf{B}_k\|_2^{1/(2q+1)} \cdot (c(n-k)/k)^{1/(4q+2)}.$$

Noting that $\|\mathbf{B} - \mathbf{B}_k\|_2 = \|\mathbf{A} - \mathbf{A}_k\|_2^{2q+1}$, and setting $q = O((\log n)/\varepsilon)$ so that

$$(c(n-k)/k)^{1/(4q+2)} = (1+\varepsilon)^{\log_{1+\varepsilon} c(n-k)/(k(4q+2))} \leq 1 + \varepsilon,$$

completes the proof ■

4.4 Distributed low rank approximation

In this section we study an algorithm for distributed low rank approximation. The model is called the *arbitrary partition model*. In this model there are s players (also called servers), each locally holding an $n \times d$ matrix $\mathbf{A}^t$, and we let $\mathbf{A} = \sum_{t \in [s]} \mathbf{A}^t$. We would like for each player to obtain a rank-k projection matrix $\mathbf{W}\mathbf{W}^T \in \mathbb{R}^{d \times d}$, for which

$$\|\mathbf{A} - \mathbf{A}\mathbf{W}\mathbf{W}^T\|_{\mathrm{F}}^2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2.$$

The motivation is that each player can then locally project his/her matrix $\mathbf{A}^t$ by computing $\mathbf{A}^t\mathbf{W}\mathbf{W}^T$. It is often useful to have such a partition of the original input matrix $\mathbf{A}$. For instance, consider the case when a customer coresponds to a row of $\mathbf{A}$, and a column to his/her purchase of a specific item. These purchases could be distributed across servers corresponding to different vendors. The communication is point-to-point, that is, all pairs of players can talk to each other through a private channel for which the other $s-2$ players do not have access to. The assumption is that $n \gg d$, though d is still large, so having communication independent of n and as small in d as possible is ideal. In [68] an $\Omega(sdk)$ bit communication lower bound was shown. Below we show an algorithm of Kannan, Vempala, and the author [68] using $O(sdk/\varepsilon)$ words of communication, assuming a word is $O(\log n)$ bits and the entries of each $\mathbf{A}^t$ are $O(\log n)$-bit integers.

We first show the following property about the top k right singular vectors of $\mathbf{SA}$ for a subspace embedding $\mathbf{S}$, as shown in [68]. The property shows that the top k right singular vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$ of $\mathbf{SA}$ provide a $(1+\varepsilon)$-approximation to the best rank-k approximation to $\mathbf{A}$. This fact quickly follows from the fact that $\|\mathbf{S}\mathbf{A}\mathbf{v}_i\|_2 = (1 \pm \varepsilon)\|\mathbf{A}\mathbf{v}_i\|_2$ for the bottom $d-k$ right singular vectors $\mathbf{v}_{k+1}, \ldots, \mathbf{v}_d$ of $\mathbf{SA}$. It is crucial that $\mathbf{S}$ is an ℓ_2-subspace embedding for $\mathbf{A}$, as otherwise there is a dependency issue since the vectors $\mathbf{A}\mathbf{v}_{k+1}, \ldots, \mathbf{A}\mathbf{v}_d$ depend on $\mathbf{S}$.

Theorem 60 *Suppose $\mathbf{A}$ is an $n \times d$ matrix. Let $\mathbf{S}$ be an $m \times d$ matrix for which $(1-\varepsilon)\|\mathbf{A}\mathbf{x}\|_2 \leq \|\mathbf{S}\mathbf{A}\mathbf{x}\|_2 \leq (1+\varepsilon)\|\mathbf{A}\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{R}^d$, that is, $\mathbf{S}$ is a subspace embedding for the column space of $\mathbf{A}$. Suppose $\mathbf{V}\mathbf{V}^T$ is a $d \times d$ matrix which projects vectors in $\mathbb{R}^d$ onto the space of the top k singular vectors of $\mathbf{SA}$. Then $\|\mathbf{A} - \mathbf{A}\mathbf{V}\mathbf{V}^T\|_{\mathrm{F}} \leq (1+O(\varepsilon)) \cdot \|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}$.*

Proof: Form an orthonormal basis of $\mathbf{R}^d$ using the right singular vectors

of $\mathbf{SA}$. Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d$ be the basis.

$$\begin{aligned}\|\mathbf{A} - \mathbf{A}\sum_{i=1}^{k} \mathbf{v}_i\mathbf{v}_i^T\|_{\mathrm{F}}^2 &= \sum_{i=k+1}^{d} \|\mathbf{A}\mathbf{v}_i\|_2^2 \leq (1+\varepsilon)^2 \sum_{i=k+1}^{d} \|\mathbf{S}\mathbf{A}\mathbf{v}_i\|_2^2 \\ &= (1+\varepsilon)^2\|\mathbf{S}\mathbf{A} - [\mathbf{S}\mathbf{A}]_k\|_{\mathrm{F}}^2,\end{aligned}$$

where the first equality follows since $\mathbf{v}_1, \ldots, \mathbf{v}_d$ is an orthonormal basis of $\mathbb{R}^d$, the inequality follows using the fact that $(1-\varepsilon)\|\mathbf{A}\mathbf{x}\|_2 \leq \|\mathbf{S}\mathbf{A}\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{R}^d$, and the final equality follows using that the $\mathbf{v}_1, \ldots, \mathbf{v}_d$ are the right singular vectors of $\mathbf{SA}$.

Suppose now $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_d$ is an orthonormal basis consisting of the singular vectors of $\mathbf{A}$. Then, we have

$$\begin{aligned}\|\mathbf{S}\mathbf{A} - [\mathbf{S}\mathbf{A}]_k\|_{\mathrm{F}}^2 &\leq \|\mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{A}\sum_{i=1}^{k} \mathbf{u}_i\mathbf{u}_i^T\|_{\mathrm{F}}^2 \\ &= \sum_{i=k+1}^{d} \|\mathbf{S}\mathbf{A}\mathbf{u}_i\|_2^2 \\ &\leq (1+\varepsilon)^2 \sum_{i=k+1}^{d} \|\mathbf{A}\mathbf{u}_i\|_2^2 \\ &= (1+\varepsilon)^2\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2,\end{aligned}$$

where the first inequality uses that the rank-k matrix $\sum_i \mathbf{u}_i\mathbf{u}_i^T$ is no better at approximating $\mathbf{SA}$ than $[\mathbf{SA}]_k$, the first equality uses that $\mathbf{u}_1, \ldots, \mathbf{u}_d$ is an orthonormal basis of $\mathbb{R}^d$, the second inequality uses that $\|\mathbf{S}\mathbf{A}\mathbf{x}\|_2 \leq (1+\varepsilon)\|\mathbf{A}\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{R}^d$, and the final equality uses that $\mathbf{u}_1, \ldots, \mathbf{u}_d$ are the right singular vectors of $\mathbf{A}$.

Thus,

$$\|\mathbf{A} - \mathbf{A}\sum_{i=1}^{k} \mathbf{v}_i\mathbf{v}_i^T\|_{\mathrm{F}}^2 \leq (1+\varepsilon)^4\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}^2,$$

and the theorem follows. ■

We will also need a variant of Lemma 45 from Chapter 4 which intuitively states that for a class of random matrices $\mathbf{S}$, if we project the rows of $\mathbf{A}$ onto the row space of $\mathbf{SA}$, we obtain a good low rank approximation. Here we use an $m \times n$ matrix $\mathbf{S}$ in which each of the entries is $+1/\sqrt{m}$ or $-1/\sqrt{m}$ with probability $1/2$, and the entries of $\mathbf{S}$ are $O(k)$-wise independent. We cite a theorem of Clarkson and Woodruff [28] which shows what we need. It can be shown by showing the following properties:

1. $\mathbf{S}$ is an ℓ_2-subspace embedding for any fixed k-dimensional subspace with probability at least 9/10, and

2. $\mathbf{S}$ has the $(\varepsilon, \delta, \ell)$-JL moment property for some $\ell \geq 2$ (see Definition 12).

Theorem 61 *(combining Theorem 4.2 and the second part of Lemma 4.3 of [28]) Let $\mathbf{S} \in \mathbb{R}^{m \times n}$ be a random sign matrix with $m = O(k \log(1/\delta)/\varepsilon)$ in which the entries are $O(k + \log(1/\delta))$-wise independent. Then with probability at least $1 - \delta$, if $\mathbf{U}\mathbf{U}^T$ is the $d \times d$ projection matrix onto the row space of $\mathbf{SA}$, then if $[\mathbf{AU}]_k$ is the best rank-k approximation to matrix $\mathbf{AU}$, we have*

$$\|[\mathbf{AU}]_k \mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}} \leq (1 + O(\varepsilon))\|\mathbf{A} - \mathbf{A}_k\|_{\mathrm{F}}.$$

The main algorithm AdaptiveCompress of [68] is given in Algorithm AdaptiveCompress below.

Here we state the key idea behind Theorem 62 below. The idea is that if each of the servers projects their matrix $\mathbf{A}^t$ to $\mathbf{PA}^t$ using an ℓ_2 subspace embedding $\mathbf{P}$, then $\mathbf{PA} = \sum_t \mathbf{PA}^t$ and by Theorem 60, if we compute the top k right singular vectors of $\mathbf{PA}$, we can send these to each server to locally project their data on. Since $\mathbf{PA}^t$ is more efficient to communicate than $\mathbf{A}^t$, this provides a considerable improvement in communication. However, the communication is proportional to d^2 and we can make it proportional to only d by additionally using Theorem 61 to first "replace" the $\mathbf{A}^t$ matrices with $\mathbf{A}^t\mathbf{U}$ matrices, where the columns of $\mathbf{U}$ are an orthonormal basis containing a $(1 + \varepsilon)$ rank-k approximation.

Theorem 62 *Consider the arbitrary partition model where an $n \times d$ matrix $\mathbf{A}^t$ resides in server t and the data matrix $\mathbf{A} = \mathbf{A}^1 + \mathbf{A}^2 + \cdots + \mathbf{A}^s$. For any $1 \geq \varepsilon > 0$, there is a protocol* AdaptiveCompress *that, on termination, leaves an $n \times d$ matrix $\mathbf{C}^t$ in server t such that the matrix $\mathbf{C} = \mathbf{C}^1 + \mathbf{C}^2 + \cdots + \mathbf{C}^s$ with arbitrarily large constant probability achieves $\|\mathbf{A} - \mathbf{C}\|_{\mathrm{F}} \leq (1+\varepsilon) \min_{\mathbf{X}:rank(\mathbf{X}) \leq k} \|\mathbf{A} - \mathbf{X}\|_{\mathrm{F}}$, using linear space, polynomial time and with total communication complexity $O(sdk/\varepsilon + sk^2/\varepsilon^4)$ real numbers. Moreover, if the entries of each $\mathbf{A}^t$ are b bits each, then the total communication is $O(sdk/\varepsilon + sk^2/\varepsilon^4)$ words each consisting of $O(b + \log(nd))$ bits.*

Proof: By definition of the AdaptiveCompress protocol, we have $\|\mathbf{A} - \mathbf{C}\|_{\mathrm{F}} = \|\mathbf{A} - \mathbf{AUVV}^T\mathbf{U}^T\|_{\mathrm{F}}$.

Algorithm 3 The AdaptiveCompress(k,ε,δ) protocol

1. Server 1 chooses a random seed for an $m \times n$ sketching matrix $\mathbf{S}$ as in Theorem 61, given parameters k, ε, and δ, where δ is a small positive constant. It communicates the seed to the other servers.

2. Server t uses the random seed to compute $\mathbf{S}$, and then $\mathbf{SA}^t$, and sends it to Server 1.

3. Server 1 computes $\sum_{t=1}^{s} \mathbf{SA}^t = \mathbf{SA}$. It computes an $m \times d$ orthonormal basis $\mathbf{U}^T$ for the row space of $\mathbf{SA}$, and sends $\mathbf{U}$ to all the servers.

4. Each server t computes $\mathbf{A}^t\mathbf{U}$.

5. Server 1 chooses another random seed for a $O(k/\varepsilon^3) \times n$ matrix $\mathbf{P}$ which is to be $O(k)$-wise independent and communicates this seed to all servers.

6. The servers then agree on a subspace embedding matrix $\mathbf{P}$ of Theorem 60 for $\mathbf{AU}$, where $\mathbf{P}$ is an $O(k/\varepsilon^3) \times n$ matrix which can be described with $O(k \log n)$ bits.

7. Server t computes $\mathbf{PA}^t\mathbf{U}$ and send it to Server 1.

8. Server 1 computes $\sum_{t=1}^{s} \mathbf{PA}^t\mathbf{U} = \mathbf{PAU}$. It computes $\mathbf{VV}^T$, which is an $O(k/\varepsilon) \times O(k/\varepsilon)$ projection matrix onto the top k singular vectors of $\mathbf{PAU}$, and sends $\mathbf{V}$ to all the servers.

9. Server t outputs $\mathbf{C}^t = \mathbf{A}^t\mathbf{UVV}^T\mathbf{U}^T$. Let $\mathbf{C} = \sum_{t=1}^{s} \mathbf{C}^t$. $\mathbf{C}$ is not computed explicitly.

Notice that $\mathbf{U}\mathbf{U}^T$ and $\mathbf{I}_d - \mathbf{U}\mathbf{U}^T$ are projections onto orthogonal subspaces. It follows by the Pythagorean theorem applied to each row that

$$\begin{aligned} & \|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}}^2 \\ = \; & \|(\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A})(\mathbf{U}\mathbf{U}^T)\|_{\mathrm{F}}^2 \qquad (51) \\ + \; & \|(\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A})(\mathbf{I}_d - \mathbf{U}\mathbf{U}^T)\|_{\mathrm{F}}^2 \\ = \; & \|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T\|^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2, \qquad (52) \end{aligned}$$

where the second equality uses that $\mathbf{U}^T\mathbf{U} = \mathbf{I}_c$, where c is the number of columns of $\mathbf{U}$.

Observe that the row spaces of $\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T$ and $\mathbf{A}\mathbf{U}\mathbf{U}^T$ are both in the row space of $\mathbf{U}^T$, and therefore in the column space of $\mathbf{U}$. It follows that since $\mathbf{U}$ has orthonormal columns, $\|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}} = \|(\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T)\mathbf{U}\|_{\mathrm{F}}$, and therefore

$$\begin{aligned} & \|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 \\ = \; & \|(\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T)\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 \\ = \; & \|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2, \qquad (53) \end{aligned}$$

where the second equality uses that $\mathbf{U}^T\mathbf{U} = \mathbf{I}_c$. Let $(\mathbf{A}\mathbf{U})_k$ be the best rank-k approximation to the matrix $\mathbf{A}\mathbf{U}$. By Theorem 60, with probability $1 - o(1)$, $\|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 \leq (1 + O(\varepsilon))\|(\mathbf{A}\mathbf{U})_k - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2$, and so

$$\begin{aligned} & \|\mathbf{A}\mathbf{U}\mathbf{V}\mathbf{V}^T - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 \\ \leq \; & (1 + O(\varepsilon))\|(\mathbf{A}\mathbf{U})_k - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 \\ \leq \; & (1 + O(\varepsilon))(\|(\mathbf{A}\mathbf{U})_k - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2). \qquad (54) \end{aligned}$$

Notice that the row space of $(\mathbf{A}\mathbf{U})_k$ is spanned by the top k right singular vectors of $\mathbf{A}\mathbf{U}$, which are in the row space of $\mathbf{U}$. Let us write $(\mathbf{A}\mathbf{U})_k = \mathbf{B}\cdot\mathbf{U}$, where $\mathbf{B}$ is a rank-k matrix.

For any vector $\mathbf{v} \in \mathbb{R}^d$, $\mathbf{v}\mathbf{U}\mathbf{U}^T$ is in the row space of $\mathbf{U}^T$, and since the columns of $\mathbf{U}$ are orthonormal, $\|\mathbf{v}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 = \|\mathbf{v}\mathbf{U}\mathbf{U}^T\mathbf{U}\|_{\mathrm{F}}^2 = \|\mathbf{v}\mathbf{U}\|_{\mathrm{F}}^2$, and so

$$\begin{aligned} & \|(\mathbf{A}\mathbf{U})_k - \mathbf{A}\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 \\ = \; & \|(\mathbf{B} - \mathbf{A})\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{A}(\mathbf{I} - \mathbf{U}\mathbf{U}^T)\|_{\mathrm{F}}^2 \\ = \; & \|\mathbf{B}\mathbf{U}\mathbf{U}^T - \mathbf{A}\mathbf{U}\mathbf{U}^T\|_{\mathrm{F}}^2 + \; \|\mathbf{A}\mathbf{U}\mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}}^2). \qquad (55) \end{aligned}$$

We apply the Pythagorean theorem to each row in the expression in (55), noting that the vectors $(\mathbf{B}_i - \mathbf{A}_i)\mathbf{U}\mathbf{U}^T$ and $\mathbf{A}_i\mathbf{U}\mathbf{U}^T - \mathbf{A}_i$ are orthogonal,

where $\mathbf{B}_i$ and $\mathbf{A}_i$ are the i-th rows of $\mathbf{B}$ and $\mathbf{A}$, respectively. Hence,

$$\begin{aligned} & \|\mathbf{BUU}^T - \mathbf{AUU}^T\|_{\mathrm{F}}^2 + \|\mathbf{AUU}^T - \mathbf{A}\|_{\mathrm{F}}^2 & (56)\\ = \; & \|\mathbf{BUU}^T - \mathbf{A}\|_{\mathrm{F}}^2 & (57)\\ = \; & \|(\mathbf{AU})_k\mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}}^2, & (58)\end{aligned}$$

where the first equality uses that

$$\begin{aligned} & \|\mathbf{BUU}^T - \mathbf{A}\|_{\mathrm{F}}^2 \\ = \; & \|(\mathbf{BUU}^T - \mathbf{A})\mathbf{UU}^T\|_{\mathrm{F}}^2 + \|(\mathbf{BUU}^T - \mathbf{A})(\mathbf{I} - \mathbf{UU}^T)\|_{\mathrm{F}}^2 \\ = \; & \|\mathbf{BUU}^T - \mathbf{AUU}^T\|_{\mathrm{F}}^2 + \|\mathbf{AUU}^T - \mathbf{A}\|_{\mathrm{F}}^2,\end{aligned}$$

and the last equality uses the definition of $\mathbf{B}$. By Theorem 61, with constant probability arbitrarily close to 1, we have

$$\|[\mathbf{AU}]_k\mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}}^2 \;\; \leq \;\; (1 + O(\varepsilon))\|\mathbf{A}_k - \mathbf{A}\|_{\mathrm{F}}^2. \qquad (59)$$

It follows by combining (51), (53), (54), (55), (56), (59), that $\|\mathbf{AUVV}^T\mathbf{U}^T - \mathbf{A}\|_{\mathrm{F}}^2 \leq (1 + O(\varepsilon))\|\mathbf{A}_k - \mathbf{A}\|_{\mathrm{F}}^2$, which shows the correctness property of AdaptiveCompress.

We now bound the communication. In the first step, by Theorem 61, m can be set to $O(k/\varepsilon)$ and the matrix $\mathbf{S}$ can be described using a random seed that is $O(k)$-wise independent. The communication of steps 1-3 is thus $O(sdk/\varepsilon)$ words. By Theorem 60, the remaining steps take $O(s(k/\varepsilon)^2/\varepsilon^2) = O(sk^2/\varepsilon^4)$ words of communication.

To obtain communication with $O(b + \log(nd))$-bit words if the entries of the matrices $\mathbf{A}^t$ are specified by b bits, Server 1 can instead send $\mathbf{SA}$ to each of the servers. The t-th server then computes $\mathbf{PA}^t(\mathbf{SA})^T$ and sends this to Server 1. Let $\mathbf{SA} = \mathbf{RU}^T$, where $\mathbf{U}^T$ is an orthonormal basis for the row space of $\mathbf{SA}$, and $\mathbf{R}$ is an $O(k/\varepsilon) \times O(k/\varepsilon)$ change of basis matrix. Server 1 computes $\sum_t \mathbf{PA}^t(\mathbf{SA})^T = P\mathbf{A}(\mathbf{SA})^T$ and sends this to each of the servers. Then, since each of the servers knows $\mathbf{R}$, it can compute $\mathbf{PA}(\mathbf{SA})^T(\mathbf{R}^T)^{-1} = \mathbf{PAU}$. It can then compute the SVD of this matrix, from which it obtains $\mathbf{VV}^T$, the projection onto its top k right singular vectors. Then, since Server t knows $\mathbf{A}^t$ and $\mathbf{U}$, it can compute $\mathbf{A}^t\mathbf{U}(\mathbf{VV}^T)\mathbf{U}^T$, as desired. Notice that in this variant of the algorithm what is sent is $\mathbf{SA}^t$ and $\mathbf{PA}^t(\mathbf{SA})^T$, which each can be specified with $O(b + \log(nd))$-bit words if the entries of the $\mathbf{A}^t$ are specified by b bits. ■

5 Graph Sparsification

Chapter Overview: This chapter is devoted to showing how sketching can be used to perform spectral sparsification of graphs. While ℓ_2-subspace embeddings compress tall and skinny matrices to small matrices, they are not particularly useful at compressing roughly square matrices, as in the case of a graph Laplacian. This chapter shows how related sketching techniques can still be used to sparsify such square matrices, resulting in a useful compression.

While ℓ_2-subspace embeddings are a powerful tool, such embeddings compress an $n \times d$ matrix to a $\text{poly}(d/\varepsilon) \times d$ matrix. This is not particularly useful if n is not too much larger than d. For instance, one natural problem is to compress a graph $G = (V, E)$ on n vertices using linear sketches so as to preserve all spectral information. In this case one is interested in a subspace embedding of the Laplacian of G, which is an $n \times n$ matrix, for which an ℓ_2-subspace embedding does not provide compression. In this chapter we explore how to use linear sketches for graphs.

We formally define the problem as follows, following the notation and outlines of [69]. Consider an ordering on the n vertices, denoted $1, \ldots, n$. We will only consider undirected graphs, though we will often talk about edges $e = \{u, v\}$ as $e = (u, v)$, where here u is less than v in the ordering we have placed on the edges. This will be for notational convenience only; the underlying graphs are undirected.

Let $\mathbf{B}_n \in \mathbb{R}^{\binom{n}{2} \times n}$ be the vertex edge incidence of the undirected, unweighted complete graph on n vertices, where the e-th row $\mathbf{b}_e$ for edge $e = (u, v)$ has a 1 in column u, a (-1) in column v, and zeroes elsewhere.

One can then write the vertex edge incidence matrix of an arbitrary undirected graph G as $\mathbf{B} = \mathbf{T} \cdot \mathbf{B}_n$, where $\mathbf{T} \in \mathbb{R}^{\binom{n}{2} \times \binom{n}{2}}$ is a diagonal matrix with a $\sqrt{w_e}$ in the e-th diagonal entry if and only if e is an edge of G and its weight is w_e. The remaining diagonal entries of $\mathbf{T}$ are equal to 0. The Laplacian is $\mathbf{K} = \mathbf{B}^T\mathbf{B}$.

The spectral sparsification problem can then be defined as follows: find a weighted subgraph H of G so that if $\tilde{\mathbf{K}}$ is the Laplacian of H, then

$$\forall \mathbf{x} \in \mathbb{R}^n,\ (1-\varepsilon)\mathbf{x}^T\mathbf{K}\mathbf{x} \leq \mathbf{x}^T\tilde{\mathbf{K}}\mathbf{x} \leq (1+\varepsilon)\mathbf{x}^T\mathbf{K}\mathbf{x}. \tag{60}$$

We call H a *spectral sparsifier* of G The usual notation for (60) is

$$(1-\varepsilon)\mathbf{K} \preceq \tilde{\mathbf{K}} \preceq (1+\varepsilon)\mathbf{K},$$

where $\mathbf{C} \preceq \mathbf{D}$ means that $\mathbf{D} - \mathbf{C}$ is positive semidefinite. We also sometimes

use the notation

$$(1-\varepsilon)\mathbf{K} \preceq_R \tilde{\mathbf{K}} \preceq_R (1+\varepsilon)\mathbf{K},$$

to mean that $(1-\varepsilon)\mathbf{x}^T\mathbf{K}\mathbf{x} \leq \mathbf{x}^T\tilde{\mathbf{K}}\mathbf{x} \leq (1+\varepsilon)\mathbf{x}^T\mathbf{K}\mathbf{x}$ for all vectors $\mathbf{x}$ in the row space of $\mathbf{K}$, which is a weaker notion since there is no guarantee for vectors $\mathbf{x}$ outside of the row space of $\mathbf{K}$.

One way to solve the spectral sparsification problem is via leverage score sampling. Suppose we write the above matrix $\mathbf{B}$ in its SVD as $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. Let us look at the leverage scores of $\mathbf{U}$, where recall the i-th leverage score $\ell_i = \|\mathbf{U}_{i*}\|_2^2$. Recall the definition of Leverage Score Sampling given in Definition 16. By Theorem 17, if we take $O(n\varepsilon^{-2}\log n)$ samples of rows of $\mathbf{U}$, constructing the sampling and rescaling matrices of Definition 16, then with probability $1-1/n$, simultaneously for all $i \in [n]$,

$$(1-\varepsilon/3) \leq \sigma_i^2(\mathbf{D}^T\mathbf{\Omega}^T\mathbf{U}) \leq (1+\varepsilon/3). \tag{61}$$

Suppose we set

$$\tilde{\mathbf{K}} = (\mathbf{D}^T\mathbf{\Omega}^T\mathbf{B})^T(\mathbf{D}^T\mathbf{\Omega}^T\mathbf{B}). \tag{62}$$

Theorem 63 *For $\tilde{\mathbf{K}}$ defined as in (62), with probability $1-1/n$,*

$$(1-\varepsilon)\mathbf{K} \preceq \tilde{\mathbf{K}} \preceq (1+\varepsilon)\mathbf{K}.$$

Proof: Using that $\mathbf{K} = \mathbf{B}^T\mathbf{B}$ and the definition of $\tilde{\mathbf{K}}$, it suffices to show for all $\mathbf{x}$,

$$\|\mathbf{B}\mathbf{x}\|_2^2 = (1 \pm \varepsilon/3)\|\mathbf{D}^T\mathbf{\Omega}^T\mathbf{B}\mathbf{x}\|_2^2.$$

By (61), and using that $\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$,

$$\|\mathbf{D}^T\mathbf{\Omega}^T\mathbf{B}\mathbf{x}\|_2^2 = (1 \pm \varepsilon/3)\|\mathbf{\Sigma}\mathbf{V}^T\mathbf{x}\|_2^2,$$

and since $\mathbf{U}$ has orthonormal columns,

$$\|\mathbf{\Sigma}\mathbf{V}^T\mathbf{x}\|_2^2 = \|\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{x}\|_2^2 = \|\mathbf{B}\mathbf{x}\|_2^2,$$

which completes the proof. ■

Notice that Theorem 63 shows that if one knows the leverage scores, then by sampling $O(n\varepsilon^{-2}\log n)$ edges of G and reweighting them, one obtains a spectral sparsifier of G. One can use algorithms for approximating the leverage scores of general matrices [44], though more efficient algorithms,

whose overall running time is near-linear in the number of edges of G, are known [111, 110].

A beautiful theorem of Kapralov, Lee, Musco, Musco, and Sidford is the following [69].

Theorem 64 *There exists a distribution* Π *on* $\varepsilon^{-2}\text{polylog}(n) \times \binom{n}{2}$ *matrices* $\mathbf{S}$ *for which with probability* $1 - 1/n$*, from* $\mathbf{S} \cdot \mathbf{B}$*, it is possible to recover a weighted subgraph* H *with* $O(\varepsilon^{-2} n \log n)$ *edges such that* H *is a spectral sparsifier of* G*. The algorithm runs in* $O(\varepsilon^{-2} n^2 \text{polylog}(n))$ *time.*

We note that Theorem 64 is not optimal in its time complexity or the number of edges in H. Indeed, Spielman and Srivastava [111] show that in $\tilde{O}(m(\log n)\varepsilon^{-2})$ time it is possible to find an H with the same number $O(\varepsilon^{-2} n \log n)$ of edges as in Theorem 64, where m is the number of edges of H. For sparse graphs, this results in significantly less time for finding H. Also, Batson, Spielman, and Srivastava [14] show that it is possible to deterministically find an H with $O(\varepsilon^{-2} n)$ edges, improving the $O(\varepsilon^{-2} n \log n)$ number of edges in Theorem 64. This latter algorithm is a bit slow, requiring $O(n^3 m \varepsilon^{-2})$ time, with some improvements for dense graphs given by Zouzias [126], though these are much slower than Theorem 64.

Despite these other works, the key feature of Theorem 64 is that it is a *linear sketch*, namely, it is formed by choosing a random oblivious (i.e., independent of $\mathbf{B}$) linear map $\mathbf{S}$ and storing $\mathbf{S} \cdot \mathbf{B}$. Then, the sparsifier H can be found using only $\mathbf{S} \cdot \mathbf{B}$, i.e., without requiring access to $\mathbf{B}$. This gives it a number of advantages, such that it implies the first algorithm for maintaining a spectral sparsifier in a data stream in the presence of insertions and deletions to the graph's edges. That is, for the other works, it was unknown how to rebuild the sparsifier if an edge is deleted; in the case when linear sketches are used to summarize the graph, it is trivial to update the sketch in the presence of an edge deletion.

In the remainder of the chapter, we give an outline of the proof of Theorem 64, following the exposition given in [69]. We restrict to unweighted graphs for the sake of presentation; the arguments generalize in a natural way to weighted graphs.

The main idea, in the author's opinion, is the use of an elegant technique due to Li, Miller and Peng [78] called "Introduction and Removal of Artificial Bases". We suspect this technique should have a number of other applications; Li, Miller and Peng use it for obtaining approximation algorithms for ℓ_2 and ℓ_1 regression. Intuitively, the theorem states that if you take any PSD matrix $\mathbf{K}$, you can form a sequence of matrices $\mathbf{K}(0), \mathbf{K}(1), \ldots, \mathbf{K}(d)$, where $\mathbf{K}(0)$ has a spectrum which is within a factor of 2 of the identity,

$\mathbf{K}(d)$ has a spectrum within a factor of 2 of $\mathbf{K}$, and for each ℓ, $\mathbf{K}(\ell-1)$ has a spectrum within a factor of 2 of $\mathbf{K}(\ell)$. Furthermore if $\mathbf{K}$ is the Laplacian of an unweighted graph, $d = O(\log n)$.

The proof of the following theorem is elementary. We believe the power in the theorem is its novel use in algorithm design.

Theorem 65 *(Recursive Sparsification of [78], as stated in [69]) Let $\mathbf{K}$ be a PSD matrix with maximum eigenvalue bounded above by λ_u and minimum eigenvalue bounded from below by λ_ℓ. Let $d = \lceil \log_2(\lambda_u/\lambda_\ell) \rceil$. For $\ell \in \{0, 1, 2, \ldots, d\}$, set*

$$\gamma(\ell) = \frac{\lambda_u}{2^\ell}.$$

Note that $\gamma(d) \leq \lambda_\ell$ and $\gamma(0) = \lambda_u$. Consider the sequence of PSD matrices $\mathbf{K}(0), \mathbf{K}(1), \ldots, \mathbf{K}(d)$, where

$$\mathbf{K}(\ell) = \mathbf{K} + \gamma(\ell)\mathbf{I}_n.$$

Then the following conditions hold.

1. $\mathbf{K} \preceq_R \mathbf{K}(d) \preceq_R 2\mathbf{K}$.
2. $\mathbf{K}(\ell) \preceq \mathbf{K}(\ell-1) \preceq 2\mathbf{K}(\ell)$ *for* $\ell = 1, 2, \ldots, d$.
3. $\mathbf{K}(0) \preceq 2\gamma(0)\mathbf{I} \preceq 2\mathbf{K}(0)$.

If $\mathbf{K}$ is the Laplacian of an unweighted graph, then its maximum eigenvalue is at most $2n$ and its minimum eigenvalue is at least $8/n^2$. We can thus set $d = \lceil \log_2 \lambda_u/\lambda_\ell \rceil = O(\log n)$ in the above.

Proof: For the first condition, for all $\mathbf{x}$ in the row space of $\mathbf{K}$,

$$\mathbf{x}^T\mathbf{K}\mathbf{x} \leq \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T(\gamma(d)\mathbf{I})\mathbf{x} \leq \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T\lambda_\ell\mathbf{x} \leq 2\mathbf{x}^T\mathbf{K}\mathbf{x}.$$

For the second condition, for all $\mathbf{x}$,

$$\mathbf{x}^T\mathbf{K}(\ell)\mathbf{x} = \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T\gamma(\ell)\mathbf{I}\mathbf{x} \leq \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T\gamma(\ell-1)\mathbf{x} = \mathbf{x}^T\mathbf{K}(\ell-1)\mathbf{x},$$

and

$$\mathbf{x}^T\mathbf{K}(\ell-1)\mathbf{x} = \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T\gamma(\ell-1)\mathbf{I}\mathbf{x} = \mathbf{x}^T\mathbf{K}\mathbf{x} + 2\mathbf{x}^T\gamma(\ell)\mathbf{I}\mathbf{x} \leq 2\mathbf{x}^T\mathbf{K}(\ell)\mathbf{x}.$$

Finally, for the third condition, for all $\mathbf{x}$,

$$\begin{aligned}\mathbf{x}^T\mathbf{K}(0)\mathbf{x} &= \mathbf{x}^T\mathbf{K}\mathbf{x} + \mathbf{x}^T\lambda_u\mathbf{I}\mathbf{x} \\ &\leq \mathbf{x}^T(2\lambda_u\mathbf{I})\mathbf{x} \\ &\leq 2\mathbf{x}^T\mathbf{K}\mathbf{x} + 2\mathbf{x}^T\lambda_u\mathbf{I}\mathbf{x} \\ &\leq 2\mathbf{x}^T\mathbf{K}(0)\mathbf{x}.\end{aligned}$$

The bounds on the eigenvalues of a Laplacian are given in [109] (the bound on the maximum eigenvalue follows from the fact that n is the maximum eigenvalue of the Laplacian of the complete graph on n vertices. The bound on the minimum eigenvalue follows from Lemma 6.1 of [109]). ■

The main idea of the algorithm is as follows. We say a PSD matrix $\tilde{\mathbf{K}}$ is a C-approximate row space sparsifier of a PSD matrix $\mathbf{K}$ if $\mathbf{K} \preceq_R \tilde{\mathbf{K}} \preceq_R C \cdot \mathbf{K}$. If we also have the stronger condition that $\mathbf{K} \preceq \tilde{\mathbf{K}} \preceq C \cdot \mathbf{K}$ we say that $\tilde{\mathbf{K}}$ is a C-approximate sparsifier of $\mathbf{K}$.

By the first condition of Theorem 65, if we had a matrix $\tilde{\mathbf{K}}$ which is a C-approximate row space sparsifier of $\mathbf{K}(d)$, then $\tilde{\mathbf{K}}$ is also a $2C$-approximate row space sparsifier to $\mathbf{K}$.

If we were not concerned with compressing the input graph G with a linear sketch, at this point we could perform Leverage Score Sampling to obtain a $(1+\varepsilon)$-approximate sparsifier to $\mathbf{K}(d)$. Indeed, by Theorem 17, it is enough to construct a distribution q for which $q_i \geq p_i/\beta$ for all i, where $\beta > 0$ is a constant.

To do this, first observe that the leverage score for a potential edge $i = (u, v)$ is given by

$$\begin{aligned}\|\mathbf{U}_{i*}\|_2^2 &= \mathbf{U}_{i*}\mathbf{\Sigma}\mathbf{V}^T(\mathbf{V}\mathbf{\Sigma}^{-2}\mathbf{V}^T)\mathbf{V}\mathbf{\Sigma}\mathbf{U}_{i*}^T &(63)\\ &= \mathbf{b}_i^T\mathbf{K}^\dagger\mathbf{b}_i. &(64)\end{aligned}$$

As $\mathbf{b}_i$ is in the row space of $\mathbf{B}$, it is also in the row space of $\mathbf{K} = \mathbf{B}^T\mathbf{B}$, since $\mathbf{B}$ and $\mathbf{K}$ have the same row space (to see this, write $\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ in its SVD and then $\mathbf{K} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$). Since $\tilde{\mathbf{K}}$ is a $2C$-approximate row space sparsifier of $\mathbf{K}$, for all $\mathbf{u}$ in the row space of $\mathbf{K}$,

$$\mathbf{u}^T\mathbf{K}\mathbf{u} \leq \mathbf{u}^T\tilde{\mathbf{K}}\mathbf{u} \leq 2C\mathbf{u}^T\mathbf{K}\mathbf{u},$$

which implies since $\mathbf{K}^+$ has the same row space as $\mathbf{K}$ (to see this, again look at the SVD),

$$\frac{1}{2C}\mathbf{u}^T\mathbf{K}^+\mathbf{u} \leq \mathbf{u}^T\tilde{K}^+\mathbf{u} \leq \mathbf{u}^T\mathbf{K}^+\mathbf{u}.$$

Since this holds for $\mathbf{u} = \mathbf{b}_i$ for all i, it follows that $\mathbf{b}_i^T \tilde{\mathbf{K}}^+ \mathbf{b}_i$ is within a factor of $2C$ of $\mathbf{b}_i^T \mathbf{K}^+ \mathbf{b}_i$ for all i. It follows by setting $q_i = \mathbf{b}_i^T \tilde{\mathbf{K}}^\dagger \mathbf{b}_i / n$, we have that $q_i \geq p_i / 2C$, where p_i are the leverage scores of $\mathbf{B}$. Hence, by Theorem 17, it suffices to take $O(n\varepsilon^{-2} \log n)$ samples of the edges of G according to q_i, reweight them, and one obtains a spectral sparsifier to the graph G.

Hence, if we were not concerned with compressing G with a linear sketch, i.e., of computing the sparsifier H from $\mathbf{SB}$ for a random oblivious mapping $\mathbf{S}$, one approach using Theorem 65 would be the following. By the third condition of Theorem 65, we can start with a sparsifier $\tilde{\mathbf{K}} = 2\gamma(0)\mathbf{I}$ which provides a 2-approximation to $\mathbf{K}(0)$, in the sense that $\mathbf{K}(0) \preceq \tilde{\mathbf{K}} \preceq 2\mathbf{K}(0)$. Then, we can apply Leverage Score Sampling and by Theorem 17, obtain a sparsifier $\tilde{\mathbf{K}}$ for which

$$\mathbf{K}(0) \preceq \tilde{\mathbf{K}} \preceq (1+\varepsilon)\mathbf{K}(0).$$

Then, by the second property of Theorem 65,

$$\mathbf{K}(1) \preceq \mathbf{K}(0) \preceq \tilde{\mathbf{K}} \preceq 2(1+\varepsilon)\mathbf{K}(1).$$

Hence, $\tilde{\mathbf{K}}$ is a $2(1+\varepsilon)$-approximation to $\mathbf{K}(1)$. We can now apply Leverage Score Sampling again, and in this way obtain $\tilde{\mathbf{K}}$ which is a $2(1+\varepsilon)$-approximation to $\mathbf{K}(2)$, etc. Note that the errors do not compound, and the number of samples in $\tilde{\mathbf{K}}$ is always $O(n\varepsilon^{-2} \log n)$. By the argument above, when $\tilde{\mathbf{K}}$ becomes a $2(1+\varepsilon)$-approximation to $\mathbf{K}(d)$, it is a $4(1+\varepsilon)$ approximation to $\mathbf{K}$, and we obtain a spectral sparsifier of G by sampling $O(n\varepsilon^{-2} \log n)$ edges according to the leverage scores of $\tilde{\mathbf{K}}$.

Thus, the only task left is to implement this hierarchy of leverage score sampling using linear sketches.

For this, we need the following standard theorem from the sparse recovery literature.

Theorem 66 *(see, e.g., [24, 51]) For any $\eta > 0$, there is an algorithm D and a distribution on matrices $\mathbf{\Phi}$ in $\mathbb{R}^{O(\eta^{-2}\mathrm{polylog}(n)) \times n}$ such that for any $\mathbf{x} \in \mathbb{R}^n$, with probability $1 - n^{-100}$ over the choice of $\mathbf{\Phi}$, the output of D on input $\mathbf{\Phi x}$ is a vector $\mathbf{w}$ with $\eta^{-2}\mathrm{polylog}(n)$ non-zero entries which satisfies the guarantee that*

$$\|\mathbf{x} - \mathbf{w}\|_\infty \leq \eta \|\mathbf{x}\|_2.$$

Several standard consequences of this theorem, as observed in [69], can be derived by setting $\eta = \frac{\varepsilon}{C \log n}$ for a constant $C > 0$, which is the setting of η we use throughout. Of particular interest is that for $0 < \varepsilon < 1/2$, from

w_i one can determine if $\mathbf{x}_i \geq \frac{1}{C \log n}\|\mathbf{x}\|_2$ or $\mathbf{x}_i < \frac{1}{2C \log n}\|\mathbf{x}\|_2$ given that it satisfies one of these two conditions. We omit the proof of this fact which can be readily verified from the statement of Theorem 66, as shown in [69].

The basic idea behind the sketching algorithm is the following intuition. Let $\mathbf{x}_e = \mathbf{T}\mathbf{B}_n\mathbf{K}^\dagger\mathbf{b}_e$ for an edge e which may or may not occur in G, which as we will see below is a vector with the important property that its e-th coordinate is either ℓ_e or 0. Then,

$$\ell_e = \|\mathbf{U}_e\|_2^2 = \mathbf{b}_e^T\mathbf{K}^\dagger\mathbf{K}\mathbf{K}^\dagger\mathbf{b}_e = \|\mathbf{B}\mathbf{K}^\dagger\mathbf{b}_e\|_2^2 = \|\mathbf{T}\mathbf{B}_n\mathbf{K}^\dagger\mathbf{b}_e\|_2^2 = \|\mathbf{x}_e\|_2^2,$$

where the first equality follows by definition of the leverage scores, the second equality follows by (63) and using that $\mathbf{K}^+ = \mathbf{K}^+\mathbf{K}\mathbf{K}^+$, the third equality follows by definition of $\mathbf{K} = \mathbf{B}^T\mathbf{B}$, the fourth equality follows from $\mathbf{T}\mathbf{B}_n = \mathbf{B}$, and the final equality follows by definition of $\mathbf{x}_e$.

Moreover, by definition of $\mathbf{T}$, the e-th coordinate of $\mathbf{x}_e$ is 0 if e does not occur in G. Otherwise, it is $\mathbf{b}_e\mathbf{K}^\dagger\mathbf{b}_e = \ell_e$. We in general could have that $\ell_e \ll \|\mathbf{x}_e\|_2$, that is, there can be many other non-zero entries among the coordinates of $\mathbf{x}_e$ other than the e-th entry.

This is where sub-sampling and Theorem 66 come to the rescue. At a given level in the Leverage Score Sampling hierarchy, that is, when trying to construct $\mathbf{K}(\ell+1)$ from $\mathbf{K}(\ell)$, we have a C-approximation $\tilde{\mathbf{K}}$ to $\mathbf{K}(\ell)$ for a given ℓ, and would like a $(1+\varepsilon)$-approximation to $\mathbf{K}(\ell)$. Here $C > 0$ is a fixed constant. To do this, suppose we sub-sample the edges of G at rates $1, 1/2, 1/4, 1/8, \ldots, 1/n^2$, where sub-sampling at rate $1/2^i$ means we randomly decide to keep each edge independently with probability $1/2^i$. Given $\tilde{\mathbf{K}}$, if $\hat{\ell}_e$ is our C-approximation to ℓ_e, if we sub-sample at rate $1/2^i$ where 2^i is within a factor of 2 of $\hat{\ell}_e$, then we would expect $\|\mathbf{x}_e\|_2^2$ to drop by a factor of $\Theta(\ell_e)$ to $\Theta(\ell_e^2)$. Moreover, if edge e is included in the sub-sampling at rate $1/2^i$, then we will still have $\mathbf{x}_e = \ell_e$. Now we can apply Theorem 66 on the sub-sampled vector $\mathbf{x}_e$ and we have that $\mathbf{x}_e = \Omega(\|\mathbf{x}_e\|_2)$, which implies that in the discussion after Theorem 66, we will be able to find edge e. What's more is that the process of dropping each edge with probability $1/2^i = 1/\Theta(\ell_e)$ can serve as the leverage score sampling step itself. Indeed, this process sets the e-th coordinate of $\mathbf{x}_e$ to 0 with probability $1 - \Theta(\ell_e)$, that is, it finds edge e with probability $\Theta(\ell_e)$, which is exactly the probability that we wanted to sample edge e with in the first place.

Thus, the algorithm is to sub-sample the edges of G at rates $1, 1/2, \ldots, 1/n^2$, and for each rate of sub-sampling, maintain the linear sketch given by Theorem 66. This involves computing $\mathbf{\Phi}^i\mathbf{T}\mathbf{B}_n$ where $\mathbf{\Phi}^i$ is a linear sketch of the form $\mathbf{\Phi} \cdot \mathbf{D}^i$, where $\mathbf{\Phi}$ is as in Theorem 66, and $\mathbf{D}^i$ is a diagonal matrix with each diagonal entry set to 1 with probability $1/2^i$ and set to 0

otherwise. We do this entire process independently $O(\log n)$ times, as each independent repetition will allow us to build a $\mathbf{K}(\ell+1)$ from a $\mathbf{K}(\ell)$ for one value of ℓ. Then, for each level of the Leverage Score Sampling hierarchy of Theorem 65, we have a $\tilde{\mathbf{K}}$. For each possible edge e, we compute $\hat{\ell}_e$ using $\tilde{\mathbf{K}}^\dagger$ which determines a sub-sampling rate $1/2^i$. By linearity, we can compute $(\mathbf{\Phi}^i \mathbf{T}\mathbf{B}_n)\tilde{\mathbf{K}}^\dagger \mathbf{b}_e$, which is the sub-sampled version of $\mathbf{x}_e$. We sample edge e if it is found by the algorithm $\mathbf{D}$ in the discussion surrounding Theorem 66, for that sub-sampling level. We can thus use these sketches to walk up the Leverage Score Sampling hierarchy of Theorem 65 and obtain a $(1+\varepsilon)$-approximate spectral sparsifier to G. Our discussion has omitted a number of details, but hopefully gives a flavor of the result. We refer the reader to [69] for futher details on the algorithm.

6 Sketching Lower Bounds for Linear Algebra

While sketching, and in particular subspace embeddings, have been used for a wide variety of applications, there are certain limitations. In this chapter we explain some of them.

Chapter Overview: In §6.1 we introduce the Schatten norms as a natural family of matrix norms including the Frobenius and operator norms, and show that they can be approximated pretty efficiently given non-oblivious methods and multiple passes over the data. In §6.2 we ask what we can do with just a single oblivious sketch of the data matrix, and show that unlike the Frobenius norm, where it can be compressed to a vector of a constant number of dimensions, for approximating the operator norm of an $n \times n$ matrix from the sketch one cannot compress to fewer than $\Omega(n^2)$ dimensions. In §6.3 we discuss streaming lower bounds for numerical linear algebra problems, such as approximate matrix product, ℓ_2-regression, and low rank approximation. In §6.4 we mention lower bounds on the dimension of ℓ_2-subspace embeddings themselves. Finally, in §6.5 we show how algorithms which sketch input data, then use the same sketch to adaptively query properties about the input, typically cannot satisfy correctness. That is, we show broad impossibility results for sketching basic properties such as the Euclidean norm of an input vector when faced with adaptively chosen queries. Thus, when using sketches inside of complex algorithms, one should make sure they are not queried adaptively, or if they are, that the algorithm will still succeed.

6.1 Schatten norms

A basic primitive is to be able to use a sketch to estimate a norm. This is a very well-studied problem with inspirations from the data stream literature, where sketching ℓ_p-norms has been extensively studied.

For problems on matrices one is often interested in error measures that depend on a matrix norm. An appealing class of such norms is the Schatten p-norms of a matrix $\mathbf{A}$, which we shall denote $\|\mathbf{A}\|_p$.

Definition 67 *For $p \geq 1$, the p-th Schatten norm $\|\mathbf{A}\|_p$ of a rank-ρ matrix $\mathbf{A}$ is defined to be*

$$\|\mathbf{A}\|_p = \left(\sum_{i=1}^{\rho} \sigma_i^p \right)^{1/p},$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_\rho > 0$ are the singular values of $\mathbf{A}$. For $p = \infty$, $\|\mathbf{A}\|_\infty$ is defined to be σ_1.

Two familiar norms immediately stand out: the Schatten 2-norm is just the Frobenius norm of $\mathbf{A}$, while the Schatten ∞-norm is the operator norm of $\mathbf{A}$. Note that typical convention is to let $\|\mathbf{A}\|_2$ denote the operator norm of $\mathbf{A}$, but in this chapter we shall use $\|\mathbf{A}\|_\infty$ to denote the operator norm to distinguish it from the Schatten 2-norm, which is the Frobenius norm.

The Schatten norms are particularly useful in that they are rotationally invariant. That is, if $\mathbf{A}$ is an $m \times n$ matrix, and if $\mathbf{J}$ is an $m \times m$ orthonormal matrix while $\mathbf{K}$ is an $n \times n$ orthonormal matrix, then $\|\mathbf{JAK}\|_p = \|\mathbf{A}\|_p$ for any $p \geq 1$. To see this, we may write $\mathbf{A} = \mathbf{U\Sigma V}^T$ in its SVD. Then $\mathbf{JU}$ has orthonormal columns, while $\mathbf{V}^T\mathbf{K}$ has orthonormal rows. It follows that the SVD of the matrix $\mathbf{JAK}$ is given in factored form as $(\mathbf{JU})\mathbf{\Sigma}(\mathbf{V}^T\mathbf{K})$, and so it has the same singular values as $\mathbf{A}$, and therefore the same Schatten p-norm for any $p \geq 1$.

One reason one is interested in estimating a matrix norm is to evaluate the quality of an approximation. For instance, suppose one finds a matrix $\tilde{\mathbf{A}}$ which is supposed to approximate $\mathbf{A}$ in a certain norm, e.g., one would like $\|\mathbf{A} - \tilde{\mathbf{A}}\|_p$ to be small. To evaluate the quality of the approximation directly one would need to compute $\|\mathbf{A} - \tilde{\mathbf{A}}\|_p$. This may be difficult to do if one is interested in a very fast running time or using a small amount of space and a small number of passes over the data. For instance, for $p \notin \{2, \infty\}$ it isn't immediately clear there is an algorithm other than computing the SVD of $\mathbf{A} - \tilde{\mathbf{A}}$.

While our focus in this chapter is on lower bounds, we mention that for integers $p \geq 1$, there is the following simple algorithm for estimating Schatten norms which has a good running time but requires multiple passes over the data. This is given in [79].

Theorem 68 *For any integer $p \geq 1$, given an $n \times d$ matrix $\mathbf{A}$, there is an $O(p \cdot \mathrm{nnz}(\mathbf{A})/\varepsilon^{-2})$ time algorithm for obtaining a $(1+\varepsilon)$-approximation to $\|\mathbf{A}\|_p^p$ with probability at least $9/10$. Further, the algorithm makes $\lceil p/2 \rceil$ passes over the data.*

Proof: Let $r = C/\varepsilon^2$ for a positive constant $C > 0$. Suppose $\mathbf{g}^1, \ldots, \mathbf{g}^r$ are independent $N(0,1)^d$ vectors, that is, they are independent vectors of i.i.d. normal random variables with mean 0 and variance 1.

We can assume $\mathbf{A}$ is symmetric by replacing $\mathbf{A}$ with the matrix

$$\mathbf{B} = \begin{pmatrix} 0 & \mathbf{A}^T \\ \mathbf{A} & 0. \end{pmatrix}$$

A straightforward calculation shows $\|\mathbf{B}\|_p = 2^{1/p}\|\mathbf{A}\|_p$ for all Schatten p-norms, and that the rank of $\mathbf{B}$ is 2ρ, where ρ is the rank of $\mathbf{A}$.

In the first pass we compute $\mathbf{B}\mathbf{g}^1, \ldots, \mathbf{B}\mathbf{g}^r$. In the second pass we compute $\mathbf{B}(\mathbf{B}\mathbf{g}^1), \ldots, \mathbf{B}(\mathbf{B}\mathbf{g}^r)$, and in general in the i-th pass we compute $\mathbf{B}^i\mathbf{g}^1, \ldots, \mathbf{B}^i\mathbf{g}^r$.

If $\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^T$ is the SVD of the symmetric matrix $\mathbf{B}$, then after $s = \lceil p/2 \rceil$ passes we will have computed $\mathbf{U}\mathbf{\Sigma}^s\mathbf{U}^T\mathbf{g}^i$ for each i, as well as $\mathbf{U}\mathbf{\Sigma}^t\mathbf{U}^T\mathbf{g}^i$ for each i where $t = \lfloor p/2 \rfloor$. Using that $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, we can compute $(\mathbf{g}^i)^T\mathbf{U}\mathbf{\Sigma}^p\mathbf{U}^T\mathbf{g}^i$ for each i. By rotational invariance, these r values are equal $(\mathbf{h}^1)^T\mathbf{\Sigma}^p\mathbf{h}^1, \ldots, (\mathbf{h}^r)^T\mathbf{\Sigma}^p\mathbf{h}^r$, where $\mathbf{h}^1, \ldots, \mathbf{h}^r$ are independent vectors of independent $N(0,1)$ random variables.

For every i, we have

$$\mathbf{E}[(\mathbf{h}^i)^T\mathbf{\Sigma}^p\mathbf{h}^i] = \sum_{j=1}^{2\rho} \mathbf{E}[(\mathbf{h}^i)_j^2\sigma_j^p] = \|\mathbf{B}\|_p^p,$$

where we use that $\mathbf{E}[(\mathbf{h}^i)_i^2] = 1$ for all i. We also have that

$$\begin{aligned}
\mathbf{E}[((\mathbf{h}^i)^T\mathbf{\Sigma}^p\mathbf{h}^i)^2] &= \sum_{j,j'} \Sigma_{j,j}^p\Sigma_{j',j'}^p\mathbf{E}[(\mathbf{h}_j^i)^2(\mathbf{h}_{j'}^i)^2] \\
&= 3\sum_j \Sigma_{j,j}^{2p} + \sum_{j\neq j'} \Sigma_{j,j}^p\Sigma_{j',j'}^p\mathbf{E}[(\mathbf{h}_j^i)^2]\mathbf{E}[(\mathbf{h}_{j'}^i)^2] \\
&= 3\sum_j \Sigma_{j,j}^{2p} + \sum_{j\neq j'} \Sigma_{j,j}^p\Sigma_{j',j'}^p \\
&\leq 4\|\mathbf{B}\|_p^{2p},
\end{aligned}$$

where the second equality uses independence of the coordinates of $\mathbf{h}^i$ and that the 4-th moment of an $N(0,1)$ random variable is 3, while the third equality uses that the variance of an $N(0,1)$ random variable is 1. It follows by Chebyshev's inequality that if $r \geq 40/\varepsilon^2$ and let $Z = \frac{1}{r}\sum_{i\in[r]}((\mathbf{h}^i)^T\mathbf{\Sigma}^p\mathbf{h}^i)^2$, then

$$\Pr[|Z - \|\mathbf{B}\|_p^p| > \varepsilon\|\mathbf{B}\|_p^p] \leq \frac{4\|\mathbf{B}\|_p^{2p}}{\varepsilon^2\|\mathbf{B}\|_p^{2p}} \cdot \frac{\varepsilon^2}{40} \leq \frac{1}{10}.$$

This shows correctness. The running time follows from our bound on r and the number s of passes. ■

6.2 Sketching the operator norm

The algorithm in the previous section has the drawback that it is not a linear sketch, and therefore requires multiple passes over the data. This is prohibitive in certain applications. We now turn our focus to linear sketches.

A first question is what it means to have a linear sketch of a matrix $\mathbf{A}$. While some applications could require a sketch of the form $\mathbf{S} \cdot \mathbf{A} \cdot \mathbf{T}$ for random matrices $\mathbf{S}$ and $\mathbf{T}$, we will not restrict ourselves to such sketches and instead consider treating an $n \times d$ matrix $\mathbf{A}$ as an nd-dimensional vector, and computing $\mathbf{L}(\mathbf{A})$, where $\mathbf{L} : \mathbb{R}^{nd} \to \mathbb{R}^k$ is a random linear operator, i.e., a $k \times nd$ matrix which multiplies $\mathbf{A}$ on the left, where $\mathbf{A}$ is treated as an nd-dimensional vector. Since we will be proving lower bounds, this generalization is appropriate.

While the Frobenius norm is easy to approximate up to a $(1+\varepsilon)$-factor via a sketch, e.g., by taking $\mathbf{L}$ to be a random Gaussian matrix with $k = O(1/\varepsilon^2)$ rows, another natural, very important Schatten norm is the Schatten-∞, or operator norm of $\mathbf{A}$. Can the operator norm of $\mathbf{A}$ be sketched efficiently?

Here we will prove a lower bound of $k = \Omega(\min(n,d)^2)$ for obtaining a fixed constant factor approximation. Note that this is tight, up to a constant factor, since if $\mathbf{S}$ is an ℓ_2-subspace embedding with $O(d)$ rows, then $\mathbf{SA}$ preserves all the singular values of $\mathbf{A}$ up to a $(1 \pm 1/3)$ factor. We prove this formally with the following lemma.

The idea is to use the min-max principle for singular values.

Lemma 69 *Suppose $\mathbf{S}$ is a $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for $\mathbf{A}$. Then, it holds that $(1-\varepsilon)\sigma_i(\mathbf{SA}) \leq \sigma_i(\mathbf{A}) \leq (1+\varepsilon)\sigma_i(\mathbf{SA})$ for all $1 \leq i \leq d$.*

Proof: The min-max principle for singular values says that

$$\sigma_i(\mathbf{A}) = \max_{Q_i} \min_{\substack{\mathbf{x} \in Q_i \\ \|\mathbf{x}\|_2 = 1}} \|\mathbf{Ax}\|,$$

where Q_i runs through all i-dimensional subspaces. Observe that the range of $\mathbf{A}$ is a subspace of dimension at most d. It follows from the definition of a subspace embedding that

$$(1-\varepsilon)\|\mathbf{Ax}\| \leq \|\mathbf{SAx}\| \leq (1+\varepsilon)\|\mathbf{Ax}\|, \quad \forall \mathbf{x} \in \mathbb{R}^d.$$

The lemma now follows from the min-max principle for singular values, since every vector in the range has its norm preserved up to a factor of $1+\varepsilon$, and so this also holds for any i-dimensional subspace of the range, for any i. ■

Similarly, if $\mathbf{T}$ is an ℓ_2 susbpace embedding with $O(d)$ columns, then $\mathbf{AT}$ preserves all the singular values of $\mathbf{A}$ up to a $(1 \pm 1/3)$ factor, so $O(\min(n,d)^2)$ is achievable.

Hence, we shall, for simplicity focus on the case when $\mathbf{A}$ is a square $n \times n$ matrix. The following $\Omega(n^2)$ lower bound on the sketching dimension t was

shown by Oded Regev [102], improving an earlier $\Omega(n^{3/2})$ lower bound of Li, Nguyễn, and the author [79]. We will need to describe several tools before giving its proof.

Define two distributions:

- μ is the distribution on $n \times n$ matrices with i.i.d. $N(0,1)$ entries.
- ν is the distribution on $n \times n$ matrices obtained by (1) sampling $\mathbf{G}$ from μ, (2) sampling $\mathbf{u}, \mathbf{v} \sim N(0, \mathbf{I}_n)$ to be independent n-dimensional vectors with i.i.d. $N(0,1)$ entries, and (3) outputting $\mathbf{G} + \frac{5}{n^{1/2}}\mathbf{u}\mathbf{v}^T$.

We will show that any linear sketch $\mathbf{L}$ for distinguishing a matrix drawn from μ from a matrix drawn from ν requires $\Omega(n^2)$ rows. For this to imply a lower bound for approximating the operator norm of a matrix, we first need to show that with good probability, a matrix drawn from μ has an operator norm which differs by a constant factor from a matrix drawn from ν.

Lemma 70 *Let $\mathbf{X}$ be a random matrix drawn from distribution μ, while $\mathbf{Y}$ is a random matrix drawn from distribution ν. With probability $1 - o(1)$, $\|\mathbf{Y}\|_\infty \geq \frac{4}{3}\|\mathbf{X}\|_\infty$.*

Proof: It suffices to show that for $\mathbf{G}$ drawn from μ and $u, \mathbf{v} \sim N(0, \mathbf{I}_n)$, that $\|\mathbf{G}\|_\infty$ and $\|\mathbf{G} + \frac{5}{n^{1/2}}\mathbf{u}\mathbf{v}^T\|_\infty$ differ by a constant factor with probability $1 - o(1)$. We use the following tail bound.

Fact 7 (Operator norm of Gaussian Random Matrix [121]) *Suppose that $\mathbf{G} \sim \mu$. Then with probability at least $1 - e^{-t^2/2}$, it holds that $\|\mathbf{X}\|_\infty \leq 2n^{1/2} + t$.*

By Fact 7, with probability $1 - e^{-\Theta(n)}$, $\|\mathbf{G}\|_\infty \leq 2.1n^{1/2}$.

Let $\mathbf{X} = \frac{5}{n^{1/2}}\mathbf{u}\mathbf{v}^T$. Since $\mathbf{X}$ is of rank one, the only non-zero singular value of $\mathbf{X}$ is equal to $\|\mathbf{X}\|_{\mathrm{F}}$. We also have $\|\mathbf{X}\|_{\mathrm{F}} \geq 4.9 \cdot n^{1/2}$ with probability $1 - 1/n$, since $\|\mathbf{u}\mathbf{v}^T\|_{\mathrm{F}}^2 \sim (\chi^2(n))^2$, where $\chi^2(n)$ is the χ^2-distribution with n degrees of freedom, which is tightly concentrated around n.

It follows with probability $1 - O(1/n)$, by the triangle inequality

$$\|\mathbf{G} + \frac{5}{n^{1/2}}\mathbf{u}\mathbf{v}^T\|_\infty \geq 4.9n^{1/2} - 2.1n^{1/2} \geq 2.8n^{1/2} \geq \frac{4}{3}\|\mathbf{G}\|_\infty.$$

■

In our proof we need the following tail bound due to Latała Suppose that $g_{i_1}, \ldots, g_{i_d}$ are i.i.d. $N(0,1)$ random variables. The following result, due to

Latała [73], bounds the tails of Gaussian chaoses $\sum a_{i_1} \cdots a_{i_d} g_{i_1} \cdots g_{i_d}$. The proof of Latała's tail bound was later simplified by Lehec [76].

Suppose that $\mathbf{A} = (a_{\mathbf{i}})_{1 \le i_1, \ldots, i_d \le n}$ is a finite multi-indexed matrix of order d. For $\mathbf{i} \in [n]^d$ and $I \subseteq [d]$, define $i_I = (i_j)_{j \in I}$. For disjoint nonempty subsets $I_1, \ldots, I_k \subseteq [d]$ define $\|\mathbf{A}\|_{I_1,\ldots,I_k}$ to be:

$$\sup \left\{ \sum_{\mathbf{i}} a_{\mathbf{i}} \mathbf{x}^{(1)}_{i_{I_1}} \cdots \mathbf{x}^{(k)}_{i_{I_k}} : \sum_{i_{I_1}} \left(\mathbf{x}^{(1)}_{i_{I_1}} \right)^2 \le 1, \ldots, \sum_{i_{I_k}} \left(\mathbf{x}^{(1)}_{i_{I_k}} \right)^2 \le 1 \right\}.$$

Also denote by $S(k,d)$ the set of all partitions of $\{1, \ldots, d\}$ into k nonempty disjoint sets $I_1, \ldots, I_k$. It is not hard to show that if a partition $\{I_1, \ldots, I_k\}$ is finer than another partition $\{J_1, \ldots, J_\ell\}$, then $\|\mathbf{A}\|_{I_1,\ldots,I_k} \le \|\mathbf{A}\|_{J_1,\ldots,J_\ell}$.

Theorem 71 *For any $t > 0$ and $d \ge 2$,*

$$\Pr \left[\left| \sum_{\mathbf{i}} a_{\mathbf{i}} \prod_{j=1}^{d} g^{(j)}_{i_j} \right| \ge t \right] \le C_d$$
$$\cdot \exp \left\{ -c_d \min_{1 \le k \le d} \min_{(I_1,\ldots,I_k) \in S(k,d)} \left(\frac{t}{\|\mathbf{A}\|_{I_1,\ldots,I_k}} \right)^{\frac{2}{k}} \right\},$$

where $C_d, c_d > 0$ are constants depending only on d.

To prove the main theorem of this section, we need a few facts about distances between distributions.

Suppose μ and ν are two probability measures on $\mathbb{R}^n$. For a convex function $\phi : \mathbb{R} \to \mathbb{R}$ such that $\phi(1) = 0$, we define the ϕ-divergence

$$D_\phi(\mu||\nu) = \int \phi \left(\frac{d\mu}{d\nu} \right) d\nu.$$

In general $D_\phi(\mu||\nu)$ is not a distance because it is not symmetric.

The *total variation distance* between μ and ν, denoted by $d_{TV}(\mu, \nu)$, is defined as $D_\phi(\mu||\nu)$ for $\phi(x) = |x-1|$. It can be verified that this is indeed a distance. It is well known that if $d_{TV}(\mu, \nu) \le c < 1$, then the probability that any, possibly randomized algorithm, can distinguish the two distributions is at most $(1+c)/2$.

The *χ^2-divergence* between μ and ν, denoted by $\chi^2(\mu||\nu)$, is defined as $D_\phi(\mu||\nu)$ for $\phi(x) = (x-1)^2$ or $\phi(x) = x^2 - 1$. It can be verified that these two choices of ϕ give exactly the same value of $D_\phi(\mu||\nu)$.

We can upper bound total variation distance in terms of the χ^2-divergence using the next proposition.

Fact 8 ([117, p90]) $d_{TV}(\mu,\nu) \leq \sqrt{\chi^2(\mu||\nu)}$.

The next proposition gives a convenient upper bound on the χ^2-divergence between a Gaussian distribution and a mixture of Gaussian distributions.

Fact 9 ([65, p97]) $\chi^2(N(0,\mathbf{I}_n)*\mu||N(0,\mathbf{I}_n)) \leq \mathbf{E}[e^{\langle \mathbf{x},\mathbf{x}'\rangle}-1]$, *where* $\mathbf{x},\mathbf{x}' \sim \mu$ *are independent.*

We can now prove the main impossibility theorem about sketching the operator norm up to a constant factor.

Theorem 72 *[102] Let* $\mathbf{L} \in \mathbb{R}^{k\times n^2}$ *be drawn from a distribution on matrices for which for any fixed* $n\times n$ *matrix* $\mathbf{A}$*, with probability at least* $9/10$ *there is an algorithm which given* $\mathbf{L}(\mathbf{A})$*, can estimate* $\|\mathbf{A}\|_\infty$ *up to a constant factor* C*, with* $1 \leq C < \frac{2}{\sqrt{3}}$*. Recall here that* $\mathbf{L}(\mathbf{A})$ *is the image (in* $\mathbb{R}^k$*) of the linear map* $\mathbf{L}$ *which takes as input* $\mathbf{A}$ *represented as a vector in* $\mathbb{R}^{n^2}$*. Under these conditions, it holds that* $k = \Omega(n^2)$.

Proof: We can assume the rows of $\mathbf{L}$ are orthonormal vectors in $\mathbb{R}^{n^2}$. Indeed, this just corresponds to a change of basis of the row space of $\mathbf{L}$, which can be performed in post-processing. That is, given $\mathbf{L}(\mathbf{A})$ one can compute $\mathbf{R}\cdot\mathbf{L}(\mathbf{A})$ where $\mathbf{R}$ is a $k\times k$ change of basis matrix.

Let the orthonormal rows of $\mathbf{L}$ be denoted $\mathbf{L}_1,\ldots,\mathbf{L}_k$. Although these are vectors in $\mathbb{R}^{n^2}$, we will sometimes think of these as $n\times n$ matrices with the orthonormal property expressed by $\mathtt{tr}(\mathbf{L}_i^T\mathbf{L}_j) = \delta_{ij}$.

Suppose $\mathbf{A}\sim\mu$. Then, since the rows of $\mathbf{L}$ are orthonormal, it follows by rotational invariance that $\mathbf{L}(\mathbf{A})$ is distributed as a k-dimensional Gaussian vector $N(0,\mathbf{I}_k)$. On the other hand, if $\mathbf{A}\sim\nu$, then $\mathbf{L}(\mathbf{A})$ is distributed as a k-dimensional Gaussian vector with a *random mean*, that is, as $N(\mathbf{X}_{\mathbf{u},\mathbf{v}},\mathbf{I}_k)$ where

$$\mathbf{X}_{\mathbf{u},\mathbf{v}} =: \frac{5}{n^{1/2}}\left(\mathbf{u}^T\mathbf{L}_1\mathbf{v},\quad \mathbf{u}^T\mathbf{L}_2\mathbf{v},\quad \cdots\quad ,\mathbf{u}^T\mathbf{L}_k\mathbf{v}\right) =: \frac{5}{n^{1/2}}\mathbf{Y}_{\mathbf{u},\mathbf{v}}.$$

We denote the distribution of $N(\mathbf{X}_{\mathbf{u},\mathbf{v}},\mathbf{I}_k)$ by $\mathcal{D}_{n,k}$. By Lemma 70 and the definition of total variation distance, to prove the theorem it suffices to upper bound $d_{TV}(N(0,\mathbf{I}_n),\mathcal{D}_{n,k})$ by a constant $C \leq 4/5$. We shall do so for $C = 1/4$.

Without loss of generality we may assume that $k \geq 16$. Consider the event $\mathcal{E}_{\mathbf{u},\mathbf{v}} = \{\|\mathbf{Y}_{\mathbf{u},\mathbf{v}}\|_2 \leq 4\sqrt{k}\}$. Since $\mathbf{E}\|\mathbf{Y}_{\mathbf{u},\mathbf{v}}\|_2^2 = k$, it follows by Markov's inequality that $\Pr_{\mathbf{u},\mathbf{v}}\{\mathcal{E}_{\mathbf{u},\mathbf{v}}\} \geq 15/16$. Let $\widetilde{\mathcal{D}}_{n,k}$ be the marginal distribution of $\mathcal{D}_{n,k}$ conditioned on $\mathcal{E}_{\mathbf{u},\mathbf{v}}$. Then

$$d_{TV}(\widetilde{\mathcal{D}}_{n,k},\mathcal{D}_{n,k}) \leq \frac{1}{8},$$

and it suffices to bound $d_{TV}(N(0,\mathbf{I}_n),\widetilde{\mathcal{D}}_{n,k})$. Resorting to χ^2-divergence by invoking Proposition 8 and Proposition 9, we have that

$$d_{TV}(N(0,\mathbf{I}_n),\widetilde{\mathcal{D}}_{n,k}) \leq \sqrt{\mathbf{E}e^{\langle \mathbf{X}_{\mathbf{u},\mathbf{v}},\mathbf{X}_{\mathbf{u}',\mathbf{v}'}\rangle}-1},$$

where $\mathbf{u},\mathbf{v},\mathbf{u}',\mathbf{v}' \sim N(0,\mathbf{I}_n)$ conditioned on $\mathcal{E}_{\mathbf{u},\mathbf{v}}$ and $\mathcal{E}_{\mathbf{u}',\mathbf{v}'}$. We first see that

$$\begin{aligned}\langle \mathbf{X}_{u,v},\mathbf{X}_{u',v'}\rangle &= \frac{25}{n}\sum_{a,b,c,d=1}^{n}\sum_{i}(\mathbf{L}^i)_{ab}(\mathbf{L}^i)_{cd}\mathbf{u}_a\mathbf{u}'_b\mathbf{v}_c\mathbf{v}'_d \\ &=: D\sum_{a,b,c,d}\mathbf{A}_{a,b,c,d}\mathbf{u}_a\mathbf{u}'_b\mathbf{v}_c\mathbf{v}'_d,\end{aligned}$$

where $D=\frac{25}{n}$ and $\mathbf{A}_{a,b,c,d}$ is an array of order 4 such that

$$\mathbf{A}_{a,b,c,d}=\sum_{i=1}^{k}\mathbf{L}^i_{ab}\mathbf{L}^i_{cd}.$$

We shall compute the partition norms of $\mathbf{A}_{a,b,c,d}$ as needed in Latała's tail bound Theorem 71.

Partition of size 1. The only possible partition is $\{1,2,3,4\}$. We have

$$\begin{aligned}\|\mathbf{A}\|_{\{1,2,3,4\}} &= \left(\sum_{a,b,c,d}\left(\sum_{i=1}^{k}\mathbf{L}^i_{a,b}\mathbf{L}^i_{c,d}\right)^2\right)^{1/2} \\ &= \left(\sum_{a,b,c,d}\sum_{i,j=1}^{k}\mathbf{L}^i_{a,b}\mathbf{L}^i_{c,d}\mathbf{L}^j_{a,b}\mathbf{L}^j_{c,d}\right)^{1/2} \\ &= \left(\sum_{a,b,c,d}\sum_{i=1}^{k}(\mathbf{L}^i_{a,b})^2(\mathbf{L}^i_{c,d})^2\right)^{1/2} \\ &= \sqrt{k}\end{aligned}$$

Partitions of size 2. The norms are automatically upper-bounded by $\|\mathbf{A}\|_{\{1,2,3,4\}}=\sqrt{k}$.

Partitions of size 3. Up to symmetry, there are only two partitions to consider: $\{1,2\},\{3\},\{4\}$, and $\{1,3\},\{2\},\{4\}$. We first consider the partition

$\{1,2\},\{3\},\{4\}$. We have

$$\begin{aligned}\|\mathbf{A}\|_{\{1,2\},\{3\},\{4\}} &= \sup_{\mathbf{W}\in\mathbb{S}^{n^2-1},\mathbf{u}^T,\mathbf{v}\in\mathbb{S}^{n-1}} \sum_{i=1}^{k}\langle L^i W\rangle\cdot \mathbf{u}^T\mathbf{L}^i\mathbf{v}\\ &\leq \left(\sum_{i=1}^{k}(\langle \mathbf{L}^i,\mathbf{W}\rangle\right)^{1/2}\cdot\left(\sum_{i=1}^{k}(\mathbf{u}^T\mathbf{L}^i\mathbf{v})^2\right)^{1/2}\\ &\leq 1,\end{aligned}$$

where the first inequality follows from Cauchy-Schwarz. We now consider the partition $\{1,3\},\{2\},\{4\}$. We have

$$\begin{aligned}\|\mathbf{A}\|_{\{1,3\},\{2\},\{4\}} &= \sup_{\mathbf{W}\in\mathbb{S}^{n^2-1},\mathbf{u}^T,\mathbf{v}\in\mathbb{S}^{n-1}} \textstyle\sum_{i=1}^{k}\langle W,((\mathbf{L}^i\mathbf{u})\otimes(\mathbf{L}^i\mathbf{v}))\rangle\\ &= \|\textstyle\sum_{i=1}^{k}((\mathbf{L}^i\mathbf{u})\otimes(\mathbf{L}^i\mathbf{v}))\|_F, \qquad (65)\end{aligned}$$

where the second equality follows by Cauchy-Schwarz.

Let $\mathbf{T}$ be the $n^2\times k$ matrix whose columns are the $\mathbf{L}^i$, interpreted as column vectors in $\mathbf{R}^{n^2}$. Let $\mathbf{T}'$ be the $n\times k$ matrix whose columns are $\mathbf{L}^i\mathbf{u}$. Similarly, $\mathbf{T}''$ is the $n\times k$ matrix whose columns are $\mathbf{L}^i\mathbf{v}$. Then

$$\|\sum_{i=1}^{k}((\mathbf{L}^i\mathbf{u})\otimes(\mathbf{L}^i\mathbf{v}))\|_\infty = \|\mathbf{T}'(\mathbf{T}'')^T\|_\infty.$$

Since $\mathbf{T}'$ and $\mathbf{T}''$ are obtained from $\mathbf{T}$ by applying a contraction, we have that $\|\mathbf{T}'\|_\infty\leq\|\mathbf{T}\|_\infty\leq 1$, and $\|\mathbf{T}''\|_\infty\leq\|\mathbf{T}\|_\infty\leq 1$. Therefore, $\|\mathbf{T}'(\mathbf{T}'')^T\|_\infty\leq 1$. Consequently, since $\mathbf{T}'(\mathbf{T}'')^T$ is an $n\times n$ matrix, $\|\mathbf{T}'(\mathbf{T}'')^T\|_{\mathrm{F}}\leq\sqrt{n}$.

Partition of size 4. The only partition is $\{1\},\{2\},\{3\},\{4\}$. Using that for integers a,b, $a\cdot b\leq(a^2+b^2)/2$, we have

$$\begin{aligned}\|\mathbf{A}\|_{\{1\},\{2\},\{3\},\{4\}} &= \sup_{\mathbf{u},\mathbf{v},\mathbf{u}',\mathbf{v}'\in\mathbb{S}^{n-1}}\sum_{i=1}^{k}\mathbf{u}^T\mathbf{L}^i\mathbf{v}\mathbf{u}'^T\mathbf{L}^i\mathbf{v}'\\ &\leq \sup_{\mathbf{u},\mathbf{v},\mathbf{u}',\mathbf{v}'}\frac{1}{2}\left(\sum_{i=1}^{k}\langle\mathbf{u}\mathbf{v}^T,\mathbf{L}^i\rangle^2+\langle\mathbf{u}'\mathbf{v}'^T,\mathbf{L}^i\rangle^2\right)\\ &\leq 1\end{aligned}$$

The last inequality follows the fact that $\mathbf{u}\mathbf{v}^T$ is a unit vector in $\mathbb{R}^{n^2}$ and $\mathbf{L}^i$'s are orthonormal vectors in $\mathbb{R}^{n^2}$.

Latała's inequality (Theorem 71) states that for $t > 0$,

$$\begin{aligned}\Pr\left[\left|\sum_{a,b,c,d} \mathbf{A}_{a,b,c,d}\mathbf{u}_a\mathbf{u}'_b\mathbf{v}_c\mathbf{v}'_d\right| > t\right] &\leq C_1 \\ &\cdot \exp\left(-c\min\left\{\frac{t}{\sqrt{k}}, \frac{t^2}{k}, \frac{t^{\frac{2}{3}}}{n^{\frac{1}{3}}}, \sqrt{t}\right\}\right)\end{aligned}$$

The above holds with no conditions imposed on $\mathbf{u}, \mathbf{v}, \mathbf{u}', \mathbf{v}'$. For convenience, we let

$$f(t) = \min\left\{\frac{t}{\sqrt{k}}, \frac{t^2}{k}, \frac{t^{\frac{2}{3}}}{n^{\frac{1}{3}}}, \sqrt{t}\right\}.$$

It follows that

$$\begin{aligned}\Pr\left[|\langle \mathbf{Y}_{\mathbf{u},\mathbf{v}}, \mathbf{Y}_{\mathbf{u}',\mathbf{v}'}\rangle| > t \middle| \mathcal{E}_{\mathbf{u},\mathbf{v}}\mathcal{E}_{\mathbf{u}',\mathbf{v}'}\right] &\leq \frac{\Pr\left[|\langle \mathbf{Y}_{\mathbf{u},\mathbf{v}}, \mathbf{Y}_{\mathbf{u}',\mathbf{v}'}\rangle| > t\right]}{\Pr[\mathcal{E}_{\mathbf{u}',\mathbf{v}'}\}\Pr\{\mathcal{E}_{\mathbf{u},\mathbf{v}}]} \\ &\leq C_2\exp\left(-c\cdot f(t)\right).\end{aligned}$$

Note that conditioned on $\mathcal{E}_{\mathbf{u},\mathbf{v}}$ and $\mathcal{E}_{\mathbf{u}',\mathbf{v}'}$,

$$|\langle Y_{\mathbf{u},\mathbf{v}}, Y_{\mathbf{u}',\mathbf{v}'}\rangle| \leq \|Y_{\mathbf{u},\mathbf{v}}\|_2\|Y_{\mathbf{u}',\mathbf{v}'}\|_2 \leq 16k.$$

We now claim that $tD = \frac{25t}{n} \leq cf(t)/2$ for all $\sqrt{k} < t < 16k$, provided $k = o(n^2)$. First, note that since $t < 16k$, $\frac{t}{\sqrt{k}} = O(\sqrt{t})$. Also, since $t > \sqrt{k}$, $\frac{t}{\sqrt{k}} \leq \frac{t^2}{k}$. Hence, $f(t) = \Theta(\min(t/\sqrt{k}, t^{2/3}/n^{1/3}))$. Since $k = o(n^2)$, if $t/\sqrt{k}$ achieves the minimum, then it is larger than $\frac{25t}{n}$. On the other hand, if $t^{2/3}/n^{1/3}$ achieves the minimum, then $cf(t)/2 \geq \frac{25t}{n}$ whenever $t = o(n^2)$, which since $t < 16k = o(n^2)$, always holds.

Integrating the tail bound gives that

$$\begin{aligned}\mathbf{E}[e^{\mathbf{X}_{\mathbf{u},\mathbf{v}},\mathbf{X}_{\mathbf{u}',\mathbf{v}'}}] &= 1 + D\int_0^{16k} e^{tD}\Pr[|\langle \mathbf{Y}_{\mathbf{u},\mathbf{v}}, \mathbf{Y}_{\mathbf{u}',\mathbf{v}'}\rangle| > t]dt \\ &\leq 1 + D\int_0^{\sqrt{k}} e^{tD}dt + D\int_{\sqrt{k}}^{16k} e^{tD - cf(t)}dt \\ &\leq 1 + D\sqrt{k}e^{\sqrt{k}D} + D\int_{\sqrt{k}}^{16k} e^{-cf(t)/2}dt \\ &\leq 1 + o(1),\end{aligned}$$

where the first inequality uses that $\Pr[|\langle \mathbf{Y}_{\mathbf{u},\mathbf{v}}, \mathbf{Y}_{\mathbf{u}',\mathbf{v}'}\rangle| > t] \leq 1$, the second inequality uses the above bound that $tD \leq cf(t)/2$, and the first part of the third inequality uses that $k = o(n^2)$. For the second part of the third inequality, since $\sqrt{k} < t < 16k$, we have that $f(t) = \Theta(\min(t/\sqrt{k}, t^{2/3}/n^{1/3}))$. Also, $f(t) \geq 1$ (assuming $k \geq n$, which we can assume, since if there is a linear sketch with $k < n$ there is also one with $k > n$), and so $D\int_{\sqrt{k}}^{16k} e^{-cf(t)/2}dt \leq D\sqrt{k}$ since the integral is dominated by a geometric series. Also, since $k = o(n^2)$, $D\sqrt{k} = o(1)$.

It follows that $d_{TV}(N(0, \mathbf{I}_n), \widetilde{\mathcal{D}}_{n,k}) \leq 1/10$ and thus

$$d_{TV}(N(0, \mathbf{I}_n), \mathcal{D}_{n,k}) \leq 1/10 + 1/8 < 1/4.$$

■

6.3 Streaming lower bounds

In this section, we explain some basic communication complexity, and how it can be used to prove bit lower bounds for the space complexity of linear algebra problems in the popular *streaming model* of computation. We refer the reader to Muthukrishnan's survey [96] for a comprehensive overview on the streaming model. We state the definition of the model that we need as follows. These results are by Clarkson and the author [28], and we follow the exposition in that paper.

In the *turnstile model* of computation for linear algebra problems, there is an input matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$ which is initialized to all zeros. We then see a finite sequence of additive updates to the coordinates of $\mathbf{A}$, where each update has the form (i, j, δ) for $i \in [n], j \in [d]$, and $\delta \in \mathbb{R}$, with the meaning that $\mathbf{A}_{i,j} \leftarrow \mathbf{A}_{i,j} + \delta$. We will restrict ourselves to the case when at all times, all entries of $\mathbf{A}$ are integers bounded by $M \leq \text{poly}(nd)$, for some fixed polynomial in n and d. We note that the sequence of updates is adversarially chosen, and multiple updates to each entry $\mathbf{A}_{i,j}$ may occur and in an arbitrary order (interleaved with other updates to entries $\mathbf{A}_{i',j'}$). One of the main goals in this computational model is to compute or approximate a function of $\mathbf{A}$ using as little space in bits as possible.

6.3.1 Communication complexity

For lower bounds in the turnstile model, we use a few definitions and basic results from two-party communication complexity, described below. We refer the reader to the book by Kushilevitz and Nisasn for more information [72]. We will call the two parties Alice and Bob.

For a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, we use $R_\delta^{1-way}(f)$ to denote the randomized communication complexity with two-sided error at most δ in which only a single message is sent from Alice to Bob. Here, only a single message $M(X)$ is sent from Alice to Bob, where M is Alice's message function of her input X and her random coins. Bob computes $f(M(X), Y)$, where f is a possibly randomized function of $M(X)$ and Bob's input Y. For every input pair (X, Y), Bob should output a correct answer with probability at least $1 - \delta$, where the probability is taken over the joint space of Alice and Bob's random coin tosses. If this holds, we say the protocol is correct. The communication complexity $R_\delta^{1-way}(f)$ is then the minimum over correct protocols, of the maximum length of Alice's message $M(X)$, over all inputs and all settings to the random coins.

We also use $R_{\mu,\delta}^{1-way}(f)$ to denote the minimum communication of a protocol, in which a single message from Alice to Bob is sent, for solving f with probability at least $1 - \delta$, where the probability now is taken over both the coin tosses of the protocol and an input distribution μ.

In the augmented indexing problem, which we call $AIND$, Alice is given $\mathbf{x} \in \{0,1\}^n$, while Bob is given both an $i \in [n]$ together with $\mathbf{x}_{i+1}, \mathbf{x}_{i+2}, \ldots, \mathbf{x}_n$. Bob should output $\mathbf{x}_i$.

Theorem 73 *([93]) $R_{1/3}^{1-way}(AIND) = \Omega(n)$ and also $R_{\mu,1/3}^{1-way}(AIND) = \Omega(n)$, where μ is uniform on $\{0,1\}^n \times [n]$.*

6.3.2 Matrix product

We start with an example showing how to use Theorem 73 for proving space lower bounds for the Matrix Product problem, which is the same as given in Definition 11. Here we also include in the definition the notions relevant for the streaming model.

Definition 74 *In the* Matrix Product *Problem matrices $\mathbf{A}$ and $\mathbf{B}$ are presented as an arbitrary stream of additive updates to their entries, where $\mathbf{A}$ and $\mathbf{B}$ each have n rows and a total of c columns. At all times in the stream we assume the entries of $\mathbf{A}$ and $\mathbf{B}$ are specified by $O(\log nc)$-bit numbers. The goal is to output a matrix $\mathbf{C}$ so that*

$$\|\mathbf{A}^T\mathbf{B} - \mathbf{C}\|_F \leq \varepsilon \|\mathbf{A}\|_F \|\mathbf{B}\|_F.$$

Theorem 75 *Suppose $n \geq \beta(\log_{10} cn)/\varepsilon^2$ for an absolute constant $\beta > 0$, and that the entries of $\mathbf{A}$ and $\mathbf{B}$ are represented by $O(\log(nc))$-bit numbers. Then any randomized 1-pass algorithm which solves Matrix Product with probability at least $4/5$ uses $\Omega(c\varepsilon^{-2}\log(nc))$ bits of space.*

Proof: Throughout we shall assume that $1/\varepsilon$ is an integer, and that c is an even integer. These conditions can be removed with minor modifications. Let Alg be a 1-pass algorithm which solves Matrix Product with probability at least $4/5$. Let $r = \log_{10}(cn)/(8\varepsilon^2)$. We use Alg to solve instances of $AIND$ on strings of size $cr/2$. It will follow by Theorem 73 that the space complexity of Alg must be $\Omega(cr) = \Omega(c\log(cn))/\varepsilon^2$.

Suppose Alice has $\mathbf{x} \in \{0,1\}^{cr/2}$. She creates a $c/2 \times n$ matrix $\mathbf{U}$ as follows. We will have that $\mathbf{U} = (\mathbf{U}^0, \mathbf{U}^1, \ldots, \mathbf{U}^{\log_{10}(cn)-1}, \mathbf{0}_{c/2\times(n-r)})$, where for each $k \in \{0,1,\ldots,\log_{10}(cn)-1\}$, $\mathbf{U}^k$ is a $c/2 \times r/(\log_{10}(cn))$ matrix with entries in the set $\{-10^k, 10^k\}$. Also, $\mathbf{0}_{c/2\times(n-r)}$ is a $c/2 \times (n-r)$ matrix consisting of all zeros.

Each entry of $\mathbf{x}$ is associated with a unique entry in a unique $\mathbf{U}^k$. If the entry in $\mathbf{x}$ is 1, the associated entry in $\mathbf{U}^k$ is 10^k, otherwise it is -10^k. Recall that $n \geq \beta(\log_{10}(cn))/\varepsilon^2$, so we can assume that $n \geq r$ provided that $\beta > 0$ is a sufficiently large constant.

Bob is given an index in $[cr/2]$, and suppose this index of $\mathbf{x}$ is associated with the (i^*, j^*)-th entry of $\mathbf{U}^{k^*}$. By the definition of the AIND problem, we can assume that Bob is given all entries of $\mathbf{U}^k$ for all $k > k^*$. Bob creates a $c/2 \times n$ matrix $\mathbf{V}$ as follows. In $\mathbf{V}$, all entries in the first $k^* r/(\log_{10}(cn))$ columns are set to 0. The entries in the remaining columns are set to the negation of their corresponding entry in $\mathbf{U}$. This is possible because Bob has $\mathbf{U}^k$ for all $k > k^*$. The remaining $n - r$ columns of $\mathbf{V}$ are set to 0. We define $\mathbf{A}^T = \mathbf{U} + \mathbf{V}$. Bob also creates the $n \times c/2$ matrix $\mathbf{B}$ which is 0 in all but the $((k^*-1)r/(\log_{10}(cn)) + j^*, 1)$-st entry, which is 1. Then,

$$\|\mathbf{A}\|_F^2 = \|\mathbf{A}^T\|_F^2 = \left(\frac{c}{2}\right)\left(\frac{r}{\log_{10}(cn)}\right)\sum_{k=1}^{k^*} 100^k \leq \left(\frac{c}{16\varepsilon^2}\right)\frac{100^{k^*+1}}{99}.$$

Using that $\|\mathbf{B}\|_F^2 = 1$,

$$\varepsilon^2\|\mathbf{A}\|_F^2\|\mathbf{B}\|_F^2 \leq \varepsilon^2\left(\frac{c}{16\varepsilon^2}\right)\frac{100^{k^*+1}}{99} = \frac{c}{2}\cdot 100^{k^*}\cdot\frac{25}{198}.$$

$\mathbf{A}^T\mathbf{B}$ has first column equal to the j^*-th column of $\mathbf{U}^{k^*}$, and remaining columns equal to zero. Let $\mathbf{C}$ be the $c/2 \times c/2$ approximation to the matrix $\mathbf{A}^T\mathbf{B}$. We say an entry $\mathbf{C}_{\ell,1}$ is *bad* if its sign disagrees with the sign of $(\mathbf{A}^T\mathbf{B})_{\ell,1}$. If an entry $\mathbf{C}_{\ell,1}$ is bad, then $((\mathbf{A}^T\mathbf{B})_{\ell,1} - \mathbf{C}_{\ell,1})^2 \geq 100^{k^*}$. Thus, the fraction of bad entries is at most $\frac{25}{198}$. Since we may assume that i^*, j^*, and k^* are chosen independently of $\mathbf{x}$, with probability at least $173/198$, $\mathrm{sign}(\mathbf{C}_{i^*,1}) = \mathrm{sign}(\mathbf{U}^{k^*}_{i^*,j^*})$.

Alice runs *Alg* on $\mathbf{U}$ in an arbitrary order, transmitting the state to Bob, who continues the computation on $\mathbf{V}$ and then on $\mathbf{B}$, again feeding the entries into *Alg* in an arbitrary order. Then with probability at least $4/5$, over *Alg*'s internal coin tosses, *Alg* outputs a matrix $\mathbf{C}$ for which $\|\mathbf{A}^T\mathbf{B} - \mathbf{C}\|_F^2 \leq \varepsilon^2 \|\mathbf{A}\|_F^2 \|\mathbf{B}\|_F^2$.

It follows that the parties can solve the AIND problem with probability at least $4/5 - 25/198 > 2/3$. The theorem now follows by Theorem 73. ■

6.3.3 Regression and low rank approximation

One can similarly use communication complexity to obtain lower bounds in the streaming model for Regression and Low Rank Approximation. The results are again obtained by reduction from Theorem 73. They are a bit more involved than those for matrix product, and so we only state several of the known theorems regarding these lower bounds. We begin with the formal statement of the problems, which are the same as defined earlier, specialized here to the streaming setting.

Definition 76 *In the ℓ_2-Regression Problem, an $n \times d$ matrix $\mathbf{A}$ and an $n \times 1$ column vector $\mathbf{b}$ are presented as a sequence of additive updates to their coordinates. We assume that at all points in the stream, the entries of $\mathbf{A}$ and $\mathbf{b}$ are specified by $O(\log nd)$-bit numbers. The goal is to output a vector $\mathbf{x}$ so that*

$$\|\mathbf{A}\mathbf{x} - \mathbf{b}\| \leq (1+\varepsilon) \min_{\mathbf{x}' \in \mathbb{R}^d} \|\mathbf{A}\mathbf{x}' - \mathbf{b}\|.$$

Theorem 77 *([28]) Suppose $n \geq d(\log_{10}(nd))/(36\varepsilon)$ and d is sufficiently large. Then any randomized 1-pass algorithm which solves the ℓ_2-Regression Problem with probability at least $7/9$ needs $\Omega(d^2\varepsilon^{-1}\log(nd))$ bits of space.*

Definition 78 *In the Rank-k Approximation Problem, we are given an integer k, value $\varepsilon > 0$, and $n \times d$ matrix $\mathbf{A}$ which is presented as a sequence of additive updates to its coordinates. The goal is to find a matrix $\tilde{\mathbf{A}}_k$ of rank at most k so that*

$$\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_F \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_F,$$

where $\mathbf{A}_k$ is the best rank-k approximation to $\mathbf{A}$.

Theorem 79 *([28]) Let $\varepsilon > 0$ and $k \geq 1$ be arbitrary. Then,*

(1) Suppose $d > \beta k/\varepsilon$ for an absolute constant $\beta > 0$. Then any randomized 1-pass algorithm which solves the Rank-k Approximation Problem with probability at least 5/6, and which receives the entries of $\mathbf{A}$ in row-order, must use $\Omega(nk/\varepsilon)$ bits of space.

(2) Suppose $n > \beta k/\varepsilon$ for an absolute constant $\beta > 0$. Then any randomized 1-pass algorithm which solves the Rank-k Approximation Problem with probability at least 5/6, and which receives the entries of $\mathbf{A}$ in column-order must use $\Omega(dk/\varepsilon)$ bits of space.

6.4 Subspace embeddings

We have seen that ℓ_2-subspace embeddings have a number of important applications, especially ones that are oblivious to the matrix $\mathbf{A}$ they are being applied to. A natural question is what the minimum dimension of such subspace embeddings needs to be. That is, we seek to design a distribution Π over $r \times n$ matrices $\mathbf{S}$, with r as small as possible, so that for any fixed $n \times d$ matrix $\mathbf{A}$, we have with constant probability over $\mathbf{S}$ drawn from Π,

$$\forall \mathbf{x} \in \mathbb{R}^d, (1-\varepsilon)\|\mathbf{A}\mathbf{x}\|_2 \leq \|\mathbf{S}\mathbf{A}\mathbf{x}\|_2 \leq (1+\varepsilon)\|\mathbf{A}\mathbf{x}\|_2. \tag{66}$$

We have seen that by choosing $\mathbf{S}$ to be a matrix of i.i.d. Gaussians, it suffices to set $r = O(d/\varepsilon^2)$, which also achieves (66) with probability $1 - \exp(-d)$.

A theorem of Nelson and Nguyễn [98] shows that the above setting of r is best possible for achieving (66), even if one desires only constant error probability.

Theorem 80 *For $n \geq Cd/\varepsilon^2$ for a sufficiently large constant $C > 0$, any distribution Π satisfying (66) with constant probability over $\mathbf{S}$ drawn from Π, satisfies $r = \Omega(d/\varepsilon^2)$.*

While Theorem 80 gives a nice dimension lower bound for subspace embeddings, it turns out that often one can do better than it in specific applications, such as the ℓ_2 Regression Problem, where it is possible to achieve a dependence on $1/\varepsilon$ that is linear. This is because in the analysis, only a subspace embedding with constant ε is needed, while additional other properties of the sketching matrix $\mathbf{S}$ are used that only incur a $1/\varepsilon$ factor in the dimension.

6.5 Adaptive algorithms

In this section we would like to point out a word of caution of using a sketch for multiple, adaptively chosen tasks.

Suppose, for example, that we have a $k \times n$ sketching matrix $\mathbf{S}$, with $k \ll n$, drawn from some distribution with the property that there is a, possibly randomized *reconstruction function* f such that for any fixed vector $\mathbf{x} \in \mathbb{R}^n$,

$$(1-\varepsilon)\|\mathbf{x}\|_2^2 \leq \|f(\mathbf{S}\mathbf{x})\|_2^2 \leq (1+\varepsilon)\|\mathbf{x}\|_2^2, \tag{67}$$

with probability at least $1-\delta$ for some parameter $\delta > 0$. In this section we will focus on the case in which $\delta < n^{-c}$ for every constant $c > 0$, that is, δ shrinks faster than any inverse polynomial in n

The property in (67) is a basic property that one could ask for a sketching matrix $\mathbf{S}$ to satisfy, and we will refer to an $(\mathbf{S}, f)$ pair satisfying this property as an ℓ_2-*sketch*. It is clear, for example, that an ℓ_2-subspace embedding has this property for certain ε and δ, where the function $f(\mathbf{S}\mathbf{x}) = \|\mathbf{S}\mathbf{x}\|_2^2$. As we have seen, such embeddings have a number of applications in linear algebra.

A natural question is if an ℓ_2-sketch can be reused, in the following sense. Suppose we compute

$$\mathbf{S} \cdot \mathbf{x}^1, \mathbf{S} \cdot \mathbf{x}^2, \mathbf{S} \cdot \mathbf{x}^3, \ldots, \mathbf{S} \cdot \mathbf{x}^r,$$

where $\mathbf{x}^1, \ldots, \mathbf{x}^r$ is an adaptively chosen sequence of vectors in $\mathbb{R}^n$. We will also assume $r \leq n^c$ for some fixed constant $c > 0$. For each $\mathbf{S} \cdot \mathbf{x}^i$, we obtain $f(\mathbf{S} \cdot \mathbf{x}^i)$. Note that if the $\mathbf{x}^i$ were fixed before the choice of $\mathbf{S}$, by a union bound over the r vectors $\mathbf{x}^1, \ldots, \mathbf{x}^r$, we should have that with probability at least $1 - \delta n^c$, simultaneously for all i,

$$(1-\varepsilon)\|\mathbf{x}^i\|_2^2 \leq \|f(\mathbf{S}\mathbf{x}^i)\|_2^2 \leq (1+\varepsilon)\|\mathbf{x}^i\|_2^2.$$

A natural question though, is what happens in the adaptive case, where the $\mathbf{x}^i$ can depend on $f(\mathbf{S}\mathbf{x}^1), f(\mathbf{S}\mathbf{x}^2), \ldots, f(\mathbf{S}\mathbf{x}^{i-1})$. As an illustrative example that this is a nontrivial issue, we first consider the standard estimator $f(\mathbf{S}\mathbf{x}) = \|\mathbf{S}\mathbf{x}\|_2^2$. An ℓ_2 sketch with this choice of estimator is often called a *Johnson-Lindenstrauss transform*.

Theorem 81 *For any $\varepsilon > 0$, and any Johnson-Lindenstrauss transform $\mathbf{S}$ with k rows and n columns, $k < n$, there is an algorithm which makes $r = \binom{k+1}{2} + (k+1) + 1$ query vectors $\mathbf{x}^1, \ldots, \mathbf{x}^r$, for which with probability 1,*

$$f(\mathbf{S}\mathbf{x}^r) \notin \left[(1-\varepsilon)\|\mathbf{x}^r\|_2^2,\ (1+\varepsilon)\|\mathbf{x}^r\|_2^2\right].$$

Further, the algorithm runs in $O(k^3)$ time and the first $r-1$ queries can be chosen non-adaptively (so the algorithm makes a single adaptive query, namely, $\mathbf{x}^r$).

Proof: The algorithm first queries the sketch on the vectors

$$\mathbf{e}_i, \mathbf{e}_i + \mathbf{e}_j \text{ for all } i, j \in [k+1],$$

where the $\mathbf{e}_i$ are the standard unit vectors in $\mathbb{R}^n$. Since $\mathbf{S}$ is a Johnson-Lindenstrauss transform, it learns $\|\mathbf{S}_i\|_2^2$ and $\|\mathbf{S}_i + \mathbf{S}_j\|_2^2$ for all $i, j \in [k+1]$, where $\mathbf{S}_i$ denotes the i-th column of $\mathbf{S}$. Since $\|\mathbf{S}_i + \mathbf{S}_j\|_2^2 = \|\mathbf{S}_i\|_2^2 + \|\mathbf{S}_j\|_2^2 + 2\langle \mathbf{S}_i, \mathbf{S}_j \rangle$, the algorithm learns $\langle \mathbf{S}_i, \mathbf{S}_j \rangle$ for all i, j.

Now consider the $n \times n$ matrix $\mathbf{A} = \mathbf{S}^T\mathbf{S}$. This matrix has rank at most k, since $\mathbf{S}$ has rank at most k. By definition, $\mathbf{A}_{i,j} = \langle \mathbf{S}_i, \mathbf{S}_j \rangle$. It follows that the algorithm has learned the upper $(k+1) \times (k+1)$ submatrix of $\mathbf{A}$, let us call this submatrix $\mathbf{B}$. As $\mathbf{B}$ has rank at most k, it follows there is a non-zero vector $\mathbf{u} \in \mathbb{R}^{k+1}$ in the kernel of the span of the rows of $\mathbf{B}$.

Consider the non-zero vector $\mathbf{v} \in \mathbb{R}^n$ obtained by padding $\mathbf{u}$ with $n - (k+1)$ zero coordinates. We claim that $\mathbf{S}\mathbf{v} = 0$, which would imply that $\|\mathbf{S}\mathbf{v}\|_2^2$ cannot provide a relative error approximation to $\|\mathbf{v}\|_2^2$ for any $\varepsilon > 0$.

To prove the claim, write $\mathbf{S} = [\mathbf{C}, \mathbf{D}]$ as a block matrix, where $\mathbf{C}$ consists of the first $k+1$ columns of $\mathbf{S}$, and $\mathbf{D}$ consists of the remaining $n - (k+1)$ columns of $\mathbf{S}$. Then

$$\mathbf{S}^T\mathbf{S} = \begin{pmatrix} \mathbf{C}^T\mathbf{C} & \mathbf{C}^T\mathbf{D} \\ \mathbf{D}^T\mathbf{C} & \mathbf{D}^T\mathbf{D} \end{pmatrix},$$

where $\mathbf{B} = \mathbf{C}^T\mathbf{C}$. Writing $\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ in its SVD, we see that u is orthogonal to the row space of $\mathbf{V}^T$, which implues $\mathbf{S}\mathbf{v} = 0$, as desired.

To conclude, note that the query algorithm makes r queries, the only adaptive one being $\mathbf{v}$, and runs in $O(k^3)$ time to compute the SVD of $\mathbf{B}$. ■

While Theorem 81 rules out using a Johnson-Lindenstrauss transform as an ℓ_2 sketch which supports adaptively chosen query vectors, one could ask if a different, more carefully designed ℓ_2 sketch, could support adaptively chosen query vectors. Unfortunately, the answer is no, in a very strong sense, as shown by Hardt and the author [60]. Namely, the authors show that for *any* ℓ_2 sketch, there is an efficient query algorithm which can find a distribution on queries $\mathbf{x}^i$ for which with constant probability, $f(\mathbf{S}\mathbf{x}^i) \notin [(1-\varepsilon)\|\mathbf{x}^i\|_2^2, (1+\varepsilon)\|\mathbf{x}^i\|_2^2$. To avoid introducing additional notation, we state the theorem of [60] informally, and refer the reader to the paper for more details.

Theorem 82 *[Informal version] There is a randomized algorithm which, given a parameter $B \geq 2$ and oracle access to an ℓ_2 sketch that uses at*

most $r = n - O(\log(nB))$ rows, with high probability finds a distribution over queries on which the linear sketch fails to satisfy (67) with constant probability.

The algorithm makes at most $\mathrm{poly}(rB)$ *adaptively chosen queries to the oracle and runs in time* $\mathrm{poly}(rB)$*. Moreover, the algorithm uses only r "rounds of adaptivity" in that the query sequence can be partitioned into at most r sequences of non-adaptive queries.*

We state some of the intuition behind the proof of Theorem 82 below.

The problem of approximating the Euclidean norm of $\mathbf{x}$ is captured by the following game between two players, Alice and Bob. Alice chooses an $r \times n$ sketching matrix $\mathbf{S}$ from distribution π. Bob makes a sequence of queries $\mathbf{x}^1, \ldots, \mathbf{x}^r \in \mathbb{R}^n$ to Alice, who only sees $\mathbf{S}\mathbf{x}^i$ on query i. This captures the fact that a sketching algorithm only has access to $\mathbf{S}\mathbf{x}^i$, rather than to $\mathbf{x}^i$ itself. The multiple queries $\mathbf{x}^i$ of Bob are the vectors whose Euclidean norm one would like to approximate using the sketching matrix $\mathbf{S}$. Alice responds by telling Bob the value $f(\mathbf{S}\mathbf{x}^i)$, which is supposed to be a $(1+\varepsilon)$-approximation to the Euclidean norm of $\mathbf{x}^i$.

Here f is an arbitrary function that need not be efficiently computable. For simplicity of presentation, we'll just focus on the case in which f uses no randomness, though Theorem 82 holds also for randomized functions f. Bob will try to learn the row space $R(\mathbf{A})$ of Alice, namely the at most r-dimensional subspace of $\mathbb{R}^n$ spanned by the rows of $\mathbf{A}$. If Bob knew $R(\mathbf{A})$, he could, with probability $1/2$ query 0^n and with probability $1/2$ query a vector in the kernel of $\mathbf{A}$. Since Alice cannot distinguish the two cases, and since the norm in one case is 0 and in the other case non-zero, she cannot provide a relative error approximation.

Theorem 82 provides an algorithm (which can be executed efficiently by Bob) that learns $r - O(1)$ orthonormal vectors that are almost contained in $R(\mathbf{A})$. While this does not give Bob a vector in the kernel of $\mathbf{A}$, it effectively reduces Alice's row space to be constant dimensional thus forcing her to make a mistake on sufficiently many queries (since the variance is large).

The conditional expectation lemma. In order to learn $R(\mathbf{A})$, Bob's initial query is drawn from the multivariate normal distribution $N(0, \tau\mathbf{I}_n)$, where $\tau\mathbf{I}_n$ is the covariance matrix, which is just a scalar τ times the identity matrix $\mathbf{I}_n$. This ensures that Alice's view of Bob's query $\mathbf{x}$, namely, the projection $P_{\mathbf{A}}\mathbf{x}$ of $\mathbf{x}$ onto $R(\mathbf{A})$, is spherically symmetric, and so only depends on $\|P_{\mathbf{A}}\mathbf{x}\|_2$. Given $\|P_{\mathbf{A}}\mathbf{x}\|_2$, Alice needs to output 0 or 1 depending on what she thinks the norm of $\mathbf{x}$ is. Since Alice has a proper subspace of

$\mathbb{R}^n$, she will be confused into thinking $\mathbf{x}$ has larger norm than it does when $\|P_{\mathbf{A}}\mathbf{x}\|_2$ is slightly larger than its expectation (for a given τ), that is, when $\mathbf{x}$ has a non-trivial correlation with $R(\mathbf{A})$.

Formally, Theorem 82 makes use of a conditional expectation lemma showing that there exists a choice of τ for which

$$\mathbf{E}_{\mathbf{x}\sim N(0,\tau\mathbf{I}_r)}\left[\|P_{\mathbf{A}}\mathbf{x}\|_2^2 \mid f(\mathbf{A}\mathbf{x})=1\right] - \mathbf{E}_{\mathbf{x}\sim N(0,\tau\mathbf{I}_r)}\left[\|P_{\mathbf{A}}\mathbf{x}\|_2^2\right]$$

is non-trivially large. This is done by showing that the sum of this difference over all possible τ in a range $[1, B]$ is noticeably positive. Here B is the approximation factor. In particular, there exists a τ for which this difference is large. To show the sum is large, for each possible condition $v = \|P_{\mathbf{A}}\mathbf{x}\|_2^2$, there is a probability $q(v)$ that the algorithm outputs 1, and as we range over all τ, $q(v)$ contributes both positively and negatively to the above difference based on v's weight in the χ^2-distribution with mean $r \cdot \tau$. The overall contribution of v can be shown to be zero. Moreover, by correctness of the sketch, $q(v)$ must typically be close to 0 for small values of v, and typically close to 1 for large values of v. Therefore $q(v)$ zeros out some of the negative contributions that v would otherwise make and ensures some positive contributions in total.

Boosting a small correlation. Given the conditional expectation lemma, one then finds many independently chosen $\mathbf{x}^i$ for which each $\mathbf{x}^i$ has a slightly increased expected projection onto Alice's space $R(\mathbf{A})$. At this point, however, it is not clear how to proceed unless one can aggregate these slight correlations into a single vector which has very high correlation with $R(\mathbf{A})$. This is accomplished by arranging all $m = \text{poly}(n)$ positively labeled vectors $\mathbf{x}^i$ into an $m \times n$ matrix $\mathbf{G}$ and computing the top right singular vector $\mathbf{v}^*$ of G. Note that this can be done efficiently. One can then show that $\|P_{\mathbf{A}}\mathbf{v}^*\| \geq 1 - 1/\text{poly}(n)$. In other words $\mathbf{v}^*$ is almost entirely contained in $R(\mathbf{A})$. This step is crucial as it gives a way to effectively reduce the dimension of Alice's space by 1.

Iterating the attack. After finding one vector inside Alice's space, one must iterate the argument. In fact Alice might initially use only a small fraction of her rows and switch to a new set of rows after Bob learned her initial rows. An iteration of the previously described attack is performed as follows. Bob makes queries from a multivariate normal distribution inside of the subspace orthogonal to the the previously found vector. In this way one effectively reduces the dimension of Alice's space by 1, and repeats the

attack until her space is of constant dimension, at which point a standard non-adaptive attack is enough to break the sketch. Several complications arise at this point. For example, each vector that we find is only approximately contained in $R(\mathbf{A})$. One needs to rule out that this approximation error could help Alice. This is done by adding a sufficient amount of global Gaussian noise to the query distribution. This has the effect of making the distribution statistically indistinguishable from a query distribution defined by vectors that are exactly contained in Alice's space. A generalized conditional expectation lemma is then shown for such distributions.

7 Open Problems

We have attempted to cover a number of examples where sketching techniques can be used to speed up numerical linear algebra applications. We could not cover everything and have of course missed out on some great material. We encourage the reader to look at other surveys in this area, such as the one by Mahoney [85], for a treatment of some of the topics that we missed.

Here we conclude with some open problems.

Open Question 1 (Spectral Low Rank Approximation) We have seen in Theorem 46 that it is possible to achieve a running time of $O(\text{nnz}(\mathbf{A})) + n \cdot \text{poly}(k/\varepsilon)$ for solving the low rank approximation problem with Frobenius norm error, namely, given an $n \times n$ matrix $\mathbf{A}$, finding a (factorization of a) rank-k matrix $\tilde{\mathbf{A}}_k = \mathbf{LUR}$, where $\mathbf{U}$ is a $k \times k$ matrix, for which

$$\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_{\text{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}.$$

On the other hand, in Theorem 59 we see that it is possible to find a projection matrix $\mathbf{ZZ}^T$ for which

$$\|\mathbf{A} - \mathbf{ZZ}^T\mathbf{A}\|_2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_2, \tag{68}$$

that is the error is with respect to the spectral rather than the Frobenius norm. The latter error measure can be much stronger than Frobenius norm error. Is it possible to achieve $O(\text{nnz}(\mathbf{A})) + n \cdot \text{poly}(k/\varepsilon)$ time and obtain the guarantee in (68)?

Open Question 2. (Robust Low Rank Approximation) We have seen very efficient, $O(\text{nnz}(\mathbf{A})) + n \cdot \text{poly}(k/\varepsilon)$ time algorithms for low rank approximation with Frobenius norm error, that is, for finding a factorization of a rank-k matrix $\tilde{\mathbf{A}}_k = \mathbf{LUR}$, where $\mathbf{U}$ is a $k \times k$ matrix, for which

$$\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_{\text{F}} \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_{\text{F}}.$$

As we have seen for regression, often the ℓ_1-norm is a more robust error measure than the ℓ_2-norm, and so here one could ask instead for finding a factorization of a rank-k matrix $\tilde{\mathbf{A}}_k$ for which

$$\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_1 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_1, \tag{69}$$

where here for an $n \times n$ matrix $\mathbf{B}$, $\|\mathbf{B}\|_1 = \sum_{i,j \in [n]} |\mathbf{B}_{i,j}|$ is the entry-wise 1-norm of $\mathbf{B}$. We are not aware of any polynomial time algorithm for this

problem, nor are we aware of an NP-hardness result. Some work in this direction is achieved by Shyamalkumar and Varadarajan [107] (see also the followup papers [34, 49, 48, 120]) who give an algorithm which are polynomial for fixed k, ε, for the weaker error measure $\|\mathbf{B}\|_V = \sum_{i=1}^{n} \|\mathbf{B}_{i*}\|_2$, that is, the V-norm denotes the sum of Euclidean norms of rows of $\mathbf{B}$, and so is more robust than the Frobenius norm, though not as robust as the entry-wise 1-norm.

Open Question 3. (Distributed Low Rank Approximation) In §4.4 we looked at the arbitrary partition model. Here there are s players, each locally holding an $n \times d$ matrix $\mathbf{A}^t$. Letting $\mathbf{A} = \sum_{t \in [s]} \mathbf{A}^t$, we would like for each player to compute the same rank-k projection matrix $\mathbf{W}\mathbf{W}^T \in \mathbb{R}^{d \times d}$, for which

$$\|\mathbf{A} - \mathbf{A}\mathbf{W}\mathbf{W}^T\|_F^2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_k\|_F^2.$$

We presented a protocol due to Kannan, Vempala, and the author [68] which obtained $O(sdk/\varepsilon) + \text{poly}(sk/\varepsilon)$ words of communication for solving this problem. In [68] a lower bound of $\Omega(sdk)$ bits of communication is also presented. Is it possible to prove an $\Omega(sdk/\varepsilon)$ communication lower bound, which would match the leading term of the upper bound? Some possibly related work is an $\Omega(dk/\varepsilon)$ space lower bound for outputting such a $\mathbf{W}$ given one pass over the rows of $\mathbf{A}$ presented one at a time in an arbitrary order [123], i.e., in the streaming model of computation. In that model, this bound is tight up to the distinction between words and bits.

Open Question 4. (Sketching the Schatten-1 Norm) In Section §6.1 we looked at the Schatten norms of an $n \times n$ matrix $\mathbf{A}$. Recall that for $p \geq 1$, the p-th Schatten norm $\|\mathbf{A}\|_p$ of a rank-ρ matrix $\mathbf{A}$ is defined to be

$$\|\mathbf{A}\|_p = \left(\sum_{i=1}^{\rho} \sigma_i^p \right)^{1/p},$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_\rho > 0$ are the singular values of $\mathbf{A}$. For $p = \infty$, $\|\mathbf{A}\|_\infty$ is defined to be σ_1. Some of the most important cases of a Schatten norm is when $p \in \{1, 2, \infty\}$, in which case it corresponds to the nuclear, Frobenius, and spectral norm, respectively. For constant factor approximation, for $p = 2$ one can sketch $\mathbf{A}$ using a constant number of dimensions, while for $p = \infty$, we saw that $\Omega(n^2)$ dimensions are needed. For $p = 1$, there is a lower bound of $\Omega(n^{1/2})$ for constant factor approximation, which can be improved to $\Omega(n^{1-\gamma})$, for an arbitrarily small constant $\gamma > 0$, if the sketch

is of a particular form called a "matrix sketch" [79]. There is no non-trivial (better than $O(n^2)$) upper bound known. What is the optimal sketching dimension for approximating $\|\mathbf{A}\|_1$ up to a constant factor?

Acknowledgements: I would like to thank Jaroslaw Blasiok, Christos Boutsidis, T.S. Jayram, Jelani Nelson, and the anonymous reviewer for a careful reading and giving many helpful comments.

References

[1] Dimitris Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *J. Comput. Syst. Sci.*, 66(4):671–687, 2003.

[2] Nir Ailon and Bernard Chazelle. Approximate nearest neighbors and the fast johnson-lindenstrauss transform. In *ACM Symposium on Theory of Computing (STOC)*, 2006.

[3] Nir Ailon and Edo Liberty. Fast dimension reduction using rademacher series on dual bch codes. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008.

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

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

[6] Sanjeev Arora, Elad Hazan, and Satyen Kale. A fast random sampling algorithm for sparsifying matrices. In *APPROX-RANDOM*, pages 272–279, 2006.

[7] H. Auerbach. *On the Area of Convex Curves with Conjugate Diameters*. PhD thesis, University of Lwów, Lwów, Poland, 1930. (in Polish).

[8] Haim Avron, Petar Maymounkov, and Sivan Toledo. Blendenpik: Supercharging lapack's least-squares solver. *SIAM J. Scientific Computing*, 32(3):1217–1236, 2010.

[9] Haim Avron, Huy L. Nguyễn, and David P. Woodruff. Subspace embeddings for the polynomial kernel. In *NIPS*, 2014.

[10] Haim Avron, Vikas Sindhwani, and David P. Woodruff. Sketching structured matrices for faster nonlinear regression. In *NIPS*, pages 2994–3002, 2013.

[11] Baruch Awerbuch and Robert Kleinberg. Online linear optimization and adaptive routing. *J. Comput. Syst. Sci.*, 74(1):97–114, 2008.

[12] Z. Bai and Y.Q. Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *The Annals of Probability 21*, 3:1275–1294, 1993.

[13] Maria-Florina Balcan, Vandana Kanchanapally, Yingyu Liang, and David P. Woodruff. Fast and communication efficient algorithms for distributd pca. In *NIPS*, 2014.

[14] J.D. Batson, D.A. Spielman, and N. Srivastava. Twice-ramanujan sparsifiers. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 255–262. ACM, 2009.

[15] Michael W Berry, Shakhina A Pulatova, and GW Stewart. Algorithm 844: Computing sparse reduced-rank approximations to sparse matrices. *ACM Transactions on Mathematical Software (TOMS)*, 31(2):252–269, 2005.

[16] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 2003.

[17] J. Bourgain, J. Lindenstrauss, and V. Milman. Approximation of zonoids by zonotopes. *Acta Math*, 162:73–141, 1989.

[18] Jean Bourgain and Jelani Nelson. Toward a unified theory of sparse dimensionality reduction in euclidean space. *CoRR*, abs/1311.2542, 2013.

[19] Christos Boutsidis, Petros Drineas, and Malik Magdon-Ismail. Near optimal column based matrix reconstruction. *SIAM Journal on Computing (SICOMP)*, 2013.

[20] Christos Boutsidis, Michael W. Mahoney, and Petros Drineas. An improved approximation algorithm for the column subset selection problem. In *Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms (SODA)*, pages 968–977, 2009.

[21] Christos Boutsidis and David P. Woodruff. Optimal cur matrix decompositions. In *STOC*, pages 353–362, 2014.

[22] J.P. Brooks and J.H. Dulá. The ℓ_1-norm best-fit hyperplane problem. Technical report, Optimization Online, 2009.

[23] J.P. Brooks, J.H. Dulá, and E.L. Boone. A pure ℓ_1-norm principal component analysis. Technical report, Optimization Online, 2010.

[24] Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. *Theor. Comput. Sci.*, 312(1):3–15, 2004.

[25] K. Clarkson. Subgradient and sampling algorithms for ℓ_1 regression. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2005.

[26] Kenneth L. Clarkson, Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, Xiangrui Meng, and David P. Woodruff. The fast cauchy transform and faster robust linear regression. In *SODA*, pages 466–477, 2013.

[27] Kenneth L Clarkson and David P Woodruff. Low rank approximation and regression in input sparsity time. In *In STOC*, 2013.

[28] K.L. Clarkson and D.P. Woodruff. Numerical linear algebra in the streaming model. In *Proceedings of the 41st annual ACM symposium on Theory of computing (STOC)*, 2009.

[29] Michael Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. *Arxiv preprint arXiv:1410.6801*, 2014.

[30] Michael Cohen, Jelani Nelson, and David P. Woodruff. Optimal approximate matrix product in terms of stable rank. *Manuscript*, 2014.

[31] Michael B. Cohen, Yin Tat Lee, Cameron Musco, Christopher Musco, Richard Peng, and Aaron Sidford. Uniform sampling for matrix approximation. *Arxiv preprint arXiv:1408.5099*, 2014.

[32] Anirban Dasgupta, Petros Drineas, Boulos Harb, Ravi Kumar, and Michael W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. *SIAM J. Comput.*, 38(5):2060–2078, 2009.

[33] Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós. A sparse johnson: Lindenstrauss transform. In *STOC*, pages 341–350, 2010.

[34] A. Deshpande and K. Varadarajan. Sampling-based dimension reduction for subspace approximation. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 641–650. ACM, 2007.

[35] Amit Deshpande, Luis Rademacher, Santosh Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. *Theory of Computing*, 2(12):225–247, 2006.

[36] D. Donoho and P. Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 1989.

[37] P. Drineas and R. Kannan. Pass efficient algorithms for approximating large matrices. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 223–232, 2003.

[38] P. Drineas, R. Kannan, and M.W. Mahoney. Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition. *SIAM Journal of Computing*, 36(1):184–206, 2006.

[39] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Subspace sampling and relative-error matrix approximation: Column-based methods. In *APPROX-RANDOM*, pages 316–326, 2006.

[40] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Subspace sampling and relative-error matrix approximation: Column-row-based methods. In *Algorithms - ESA 2006, 14th Annual European Symposium, Zurich, Switzerland, September 11-13, 2006, Proceedings*, volume 4168 of *Lecture Notes in Computer Science*, pages 304–314. Springer, 2006.

[41] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Relative-error cur matrix decompositions. *SIAM Journal Matrix Analysis and Applications*, 30(2):844–881, 2008.

[42] P. Drineas, M.W. Mahoney, and S. Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1127–1136, 2006.

[43] P. Drineas, M.W. Mahoney, S. Muthukrishnan, and T. Sarlos. Faster least squares approximation, Technical Report, arXiv:0710.1435, 2007.

[44] 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*, 13:3475–3506, 2012.

[45] Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Subspace sampling and relative-error matrix approximation: Column-based methods. In *APPROX-RANDOM*, pages 316–326, 2006.

[46] Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Subspace sampling and relative-error matrix approximation: Column-row-based methods. In *ESA*, pages 304–314, 2006.

[47] Uriel Feige and Eran Ofek. Spectral techniques applied to sparse random graphs. *Random Struct. Algorithms*, 27(2):251–275, 2005.

[48] D. Feldman and M. Langberg. A unified framework for approximating and clustering data. In *Proc. 41th Annu. ACM Symp. on Theory of Computing (STOC), to appear*, 2011.

[49] Dan Feldman, Morteza Monemizadeh, Christian Sohler, and David P. Woodruff. Coresets and sketches for high dimensional subspace approximation problems. In *SODA*, pages 630–649, 2010.

[50] Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, 2013.

[51] Anna C. Gilbert, Yi Li, Ely Porat, and Martin J. Strauss. Approximate sparse recovery: optimizing time and measurements. In *STOC*, pages 475–484, 2010.

[52] Alex Gittens and Michael W Mahoney. Revisiting the nystrom method for improved large-scale machine learning. *arXiv preprint arXiv:1303.1849*, 2013.

[53] Gene H. Golub and Charles F. van Loan. *Matrix computations (3. ed.)*. Johns Hopkins University Press, 1996.

[54] S.A. Goreinov, EE Tyrtyshnikov, and NL Zamarashkin. A theory of pseudoskeleton approximations. *Linear Algebra and its Applications*, 261(1-3):1–21, 1997.

[55] S.A. Goreinov, N.L. Zamarashkin, and E.E. Tyrtyshnikov. Pseudo-skeleton approximations by matrices of maximal volume. *Mathematical Notes*, 62(4):515–519, 1997.

[56] M. Gu and L. Miranian. Strong rank revealing Cholesky factorization. *Electronic Transactions on Numerical Analysis*, 17:76–92, 2004.

[57] Venkatesan Guruswami and Ali Kemal Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1207–1214. SIAM, 2012.

[58] Uffe Haagerup. The best constants in the khintchine inequality. *Studia Mathematica*, 70(3):231–283, 1981.

[59] 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.

[60] Moritz Hardt and David P. Woodruff. How robust are linear sketches to adaptive inputs? In *STOC*, pages 121–130, 2013.

[61] T.M. Hwang, W.W. Lin, and D. Pierce. Improved bound for rank revealing LU factorizations. *Linear algebra and its applications*, 261(1-3):173–186, 1997.

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

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

[64] Piotr Indyk. Uncertainty principles, extractors, and explicit embeddings of l2 into l1. In *STOC*, pages 615–620, 2007.

[65] Yuri Ingster and I. A. Suslina. *Nonparametric Goodness-of-Fit Testing Under Gaussian Models*. Springer, 1st edition, 2002.

[66] William Johnson and Gideon Schechtman. Very tight embeddings of subspaces of l_p, $1 = p < 2$, into ℓ_p^n. *Geometric and Functional Analysis*, 13(4):845–851, 2003.

[67] Daniel M. Kane and Jelani Nelson. Sparser johnson-lindenstrauss transforms. *J. ACM*, 61(1):4, 2014.

[68] Ravi Kannan, Santosh Vempala, and David P. Woodruff. Principal component analysis and higher correlations for distributed data. In *COLT*, pages 1040–1057, 2014.

[69] Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. *CoRR*, abs/1407.1289, 2014.

[70] Q. Ke and T. Kanade. Robust subspace computation using ℓ_1 norm, 2003. Technical Report CMU-CS-03-172, Carnegie Mellon University, Pittsburgh, PA.

[71] Qifa Ke and Takeo Kanade. Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *CVPR (1)*, pages 739–746, 2005.

[72] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[73] Rafał Latała. Estimates of moments and tails of Gaussian chaoses. *Ann. Probab.*, 34(6):2315–2331, 2006.

[74] Béatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Ann. Statist.*, 28(5):1302–1338, 2000.

[75] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 2001.

[76] Joseph Lehec. Moments of the Gaussian chaos. In *Séminaire de Probabilités XLIII*, volume 2006 of *Lecture Notes in Math.*, pages 327–340. Springer, Berlin, 2011.

[77] D. Lewis. Finite dimensional subspaces of l_p. *Studia Math*, 63:207–211, 1978.

[78] Mu Li, Gary L. Miller, and Richard Peng. Iterative row sampling. In *FOCS*, pages 127–136, 2013.

[79] Yi Li, Huy L. Nguyễn, and David P. Woodruff. On sketching matrix norms and the top singular vector. In *SODA*, 2014.

[80] D.G. Luenberger and Y. Ye. *Linear and nonlinear programming*, volume 116. Springer Verlag, 2008.

[81] M. Magdon-Ismail. Row Sampling for Matrix Algorithms via a Non-Commutative Bernstein Bound. *Arxiv preprint arXiv:1008.0587*, 2010.

[82] Malik Magdon-Ismail. Using a non-commutative bernstein bound to approximate some matrix algorithms in the spectral norm. *CoRR*, abs/1103.5453, 2011.

[83] A. Magen and A. Zouzias. Low Rank Matrix-valued Chernoff Bounds and Approximate Matrix Multiplication. *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2011.

[84] Avner Magen. Dimensionality reductions in l_2 that preserve volumes and distance to affine spaces. *Discrete & Computational Geometry*, 38(1):139–153, 2007.

[85] Michael W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011.

[86] Michael W. Mahoney and Petros Drineas. Cur matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences (PNAS)*, 106:697–702, 2009.

[87] Michael W Mahoney and Petros Drineas. Cur matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.

[88] O. L. Mangasarian. Arbitrary-norm separating plane. *Operations Research Letters*, 24:15–23, 1997.

[89] Jiri Matousek. *Lectures on Discrete Geometry*. Springer, 2002.

[90] A. Maurer. A bound on the deviation probability for sums of non-negative random variables. *Journal of Inequalities in Pure and Applied Mathematics*, 4(1:15), 2003.

[91] X. Meng, M. A. Saunders, and M. W. Mahoney. LSRN: A Parallel Iterative Solver for Strongly Over- or Under-Determined Systems. *ArXiv e-prints*, September 2011.

[92] Xiangrui Meng and Michael W Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *In STOC*, pages 91–100. ACM, 2013.

[93] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. *J. Comput. Syst. Sci.*, 57(1):37–49, 1998.

[94] Thomas P. Minka. A comparison of numerical optimizers for logistic regression. Technical report, Microsoft, 2003.

[95] L. Miranian and M. Gu. Strong rank revealing LU factorizations. *Linear Algebra and its Applications*, 367(C):1–16, July 2003.

[96] S. Muthukrishnan. *Data streams: algorithms and applications*. Foundations and Trends in Theoretical Computer Science, 2005.

[97] Jelani Nelson and Huy L Nguyên. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *In FOCS*, 2013.

[98] Jelani Nelson and Huy L. Nguyên. Lower bounds for oblivious subspace embeddings. In *ICALP (1)*, pages 883–894, 2014.

[99] Jelani Nelson and Huy L. Nguyễn. Sparsity lower bounds for dimensionality reducing maps. In *STOC*, pages 101–110, 2013.

[100] Huy Le Nguyễn. Personal communication, 2013.

[101] C.T. Pan. On the existence and computation of rank-revealing LU factorizations. *Linear Algebra and its Applications*, 316(1-3):199–222, 2000.

[102] Oded Regev. Personal communication, 2014.

[103] V. Rokhlin and M. Tygert. A fast randomized algorithm for overdetermined linear least-squares regression. *Proceedings of the National Academy of Sciences*, 105(36):13212, 2008.

[104] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *CoRR*, abs/1003.2990v2, 2010.

[105] T. Sarlós. Improved approximation algorithms for large matrices via random projections. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2006.

[106] Schechtman. More on embedding subspaces of l_p into ℓ_r^n. *Composition Math*, 61:159–170, 1987.

[107] N.D. Shyamalkumar and K. Varadarajan. Efficient subspace approximation algorithms. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 532–540, 2007.

[108] Christian Sohler and David P. Woodruff. Subspace embeddings for the l_1-norm with applications. In *STOC*, pages 755–764, 2011.

[109] Daniel A. Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *STOC*, pages 81–90, 2004.

[110] Daniel A. Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *SIAM J. Comput.*, 40(4):981–1025, 2011.

[111] N. Srivastava and D.A. Spielman. Graph sparsifications by effective resistances. In *Proceedings of the 40th ACM Symposium on Theory of Computing (STOC)*, 2008.

[112] G.W. Stewart. Four algorithms for the efficient computation of truncated QR approximations to a sparse matrix. *Numerische Mathematik*, 83:313–323, 1999.

[113] Michel Talagrand. Embedding subspaces of l_1 into ℓ_1^n. *Proceedings of the American Mathematical Society*, 108(2):363–369, 1990.

[114] Zhihui Tang. *Fast Transforms Based on Structured Matrices With Applications to The Fast Multipole Method*. PhD thesis, PhD Thesis, University of Maryland College Park, 2004.

[115] Mikkel Thorup and Yin Zhang. Tabulation-based 5-independent hashing with applications to linear probing and second moment estimation. *SIAM J. Comput.*, 41(2):293–331, 2012.

[116] Joel Tropp. Improved analysis of the subsampled randomized hadamard transform. *Adv. Adapt. Data Anal., special issue, "Sparse Representation of Data and Images*, 2011.

[117] Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 1st edition, 2008.

[118] E. Tyrtyshnikov. Mosaic-skeleton approximations. *Calcolo*, 33(1):47–57, 1996.

[119] E. Tyrtyshnikov. Incomplete cross approximation in the mosaic-skeleton method. *Computing*, 64(4):367–380, 2000.

[120] Kasturi Varadarajan and Xin Xiao. On the sensitivity of shape fitting problems. In *FSTTCS*, pages 486–497, 2012.

[121] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Y. C. Eldar and G. Kutyniok, editors, *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2011.

[122] S. Wang and Z. Zhang. Improving cur matrix decomposition and the nystrom approximation via adaptive sampling. *Journal of Machine Learning Research*, 2013.

[123] David P. Woodruff. Low rank approximation lower bounds in row-update streams. In *NIPS*, 2014.

[124] David P. Woodruff and Qin Zhang. Subspace embeddings and ℓ_p-regression using exponential random variables. *CoRR*, 2013.

[125] Dean Foster Yichao Lu, Paramveer Dhillon and Lyle Ungar. Faster ridge regression via the subsampled randomized hadamard transform. In *Proceedings of the Neural Information Processing Systems (NIPS) Conference*, 2013.

[126] Anastasios Zouzias. A matrix hyperbolic cosine algorithm and applications. In *ICALP (1)*, pages 846–858, 2012.