Structural properties underlying high-quality Randomized Numerical Linear Algebra algorithms

Michael W. Mahoney * Petros Drineas †

This chapter appears as: "Structural properties underlying high-quality Randomized Numerical Linear Algebra algorithms," M. W. Mahoney and P. Drineas, In *Handbook of Big Data*. pp. 137-154, edited by P. Bühlmann, P. Drineas, M. Kane, and M. van de Laan, Chapman and Hall/CRC Press, 2016.

1 Introduction

In recent years, the amount of data that has been generated and recorded has grown enormously, and data are now seen to be at the heart of modern economic activity, innovation, and growth. See, for example, the report by the McKinsey Global Institute [51], which identifies ways in which Big Data have transformed the modern world, as well as the report by the National Research Council [19], which discusses reasons for and technical challenges in massive data analysis. In many cases, these so-called Big Data are modeled as matrices, basically since an $m \times n$ matrix A provides a natural mathematical structure with which to encode information about m objects, each of which is described by n features. As a result, while linear algebra algorithms have been of interest for decades in areas such as Numerical Linear Algebra (NLA) and scientific computing, in recent years there has been renewed interest in developing matrix algorithms that are appropriate for the analysis of large datasets that are represented in the form of matrices. For example, tools such as the Singular Value Decomposition (SVD) and the related Principal Components Analysis (PCA) [38] permit the low-rank approximation of a matrix, and they have have had a profound impact in diverse areas of science and engineering. They have also been studied extensively in large-scale machine learning and data analysis applications, in settings ranging from web search engines and social network analysis to the analysis of astronomical and biological data.

Importantly, the structural and noise properties of matrices that arise in machine learning and data analysis applications are typically very different than those of matrices that arise in scientific computing and NLA. This has led researchers to revisit traditional problems in light of new requirements and to consider novel algorithmic approaches to many traditional matrix problems. One of the more remarkable trends in recent years is a new paradigm that arose in Theoretical Computer Science (TCS) and that involves the use of randomization as a computational resource for the design and analysis of algorithms for fundamental matrix problems. Randomized Numerical Linear Algebra (RandNLA) is the interdisciplinary research area that exploits randomization as a computational resource to develop improved algorithms for large-scale linear algebra problems, *e.g.*, matrix multiplication, linear regression, low-rank matrix approximation, etc. [49]. In this chapter, we will discuss RandNLA, with an emphasis on highlighting how many of the most

*International Computer Science Institute and Department of Statistics, University of California at Berkeley, Berkeley, CA, mmahoney@stat.berkeley.edu.

†Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, drinep@cs.rpi.edu.

interesting RandNLA developments for problems related to improved low-rank matrix approximation boil down to exploiting a particular structural property of Euclidean vector spaces. This structural property is of interest in and of itself (for researchers interested in linear algebra *per se*), but it is also of interest (for researchers interested in *using* linear algebra) since it highlights strong connections between algorithms for many seemingly-unrelated matrix problems.

2 Overview

As background, we note that early work in RandNLA focused on low-rank approximation problems and led to results that were primarily of theoretical interest in idealized models of data access [34, 57, 23, 24, 25, 63]. An overview of RandNLA for readers not familiar with the area has recently been provided [49]. Subsequent work on very over-determined linear regression problems, *e.g.*, least-squares regression problems with an input matrix $A \in \mathbb{R}^{m \times n}$, with $m \gg n$, led to several remarkable successes for RandNLA: theoretical results for worst-case inputs for the running time in the RAM model that improve upon the 200 year old Gaussian elimination [31, 26, 17, 53, 56]; high-quality implementations that are competitive with or better than traditional deterministic implementations, *e.g.*, as provided by LAPACK, on a single machine [62, 2, 18]; and high-quality implementations in parallel and distributed environments on up to terabyte-sized input matrices [55, 16, 54, 70, 71].

As has been described in detail, *e.g.*, in [49], both the more theoretical as well as the more applied successes of RandNLA for these very over-determined linear regression problems were achieved by using, implicitly or explicitly, the so-called statistical leverage scores of the tall input matrix A.[1] In some cases, the use of leverage scores was explicit, in that one used exact or approximate leverage scores to construct a nonuniform importance sampling probability distribution with respect to which to sample rows from the input matrix, thereby constructing a data-aware subspace embedding [28, 26]. In other cases, the use of leverage scores was implicit, in that one performed a random projection, thereby implementing a data-oblivious subspace embedding [31, 68].[2] In both cases, the improved theoretical and practical results for over-determined linear regression problems were obtained by coupling the original, rather theoretical, RandNLA ideas more closely with structural properties of the input data [49].

In parallel with these successes on over-determined regression problems, there have also been several impressive successes on applying RandNLA methods to a wide range of seemingly-different low-rank matrix approximation problems.[3] For example, consider the following problems (which are described in more detail in Section 4):

- the Column Subset Selection Problem (CSSP), in which one seeks to select the most informative subset of exactly k columns from a matrix;
- the problem of using random projections to approximate low-rank matrix approximations faster than traditional SVD-based or QR-based deterministic methods either for worst-case input matrices and/or for inputs that are typical in scientific computing applications;

[1] The statistical leverage scores of a tall matrix $A \in \mathbb{R}^{m \times n}$ with $m \gg n$ are equal to the diagonal elements of the projection matrix onto the column span of A [14, 50, 48]. Thus, they capture a subtle but important structural property of the Euclidean vector space from which the data were drawn.

[2] Random projections can be applied in more general metric spaces, but in a Euclidean vector space a random projection essentially amounts to rotating to a random basis, where the leverage scores are uniform and thus where uniform sampling can be applied [49].

[3] By "low-rank matrix approximation problems," we informally mean problems where the input is a general matrix $A \in \mathbb{R}^{m \times n}$, where both m and n are large, and a rank parameter $k \ll \min\{m, n\}$; the output is a low-rank approximation to A, not necessarily the optimal one, that is computed via the SVD.

- the problem of developing improved Nyström-based low-rank matrix approximations of symmetric positive definite matrices; and
- the problem of developing improved machine learning and data analysis methods to identify interesting features in the data (feature selection).

These problems often arise in very different research areas, and they are—at least superficially—quite different. Relatedly, it can be difficult to tell what—if anything—improved algorithms for one problem mean for the possibility of improved algorithms for the others.

In this chapter, we highlight and discuss a particular deterministic structural property of Euclidean vector spaces that underlies the recent improvements in RandNLA algorithms for all of the above low-rank matrix approximation problems.[4] (See Lemma 1 below for a statement of this result.) This structural property characterizes the interaction between the singular subspaces of the input matrix A and *any* (deterministic or randomized) "sketching" matrix. In particular, this structural property is deterministic, in the sense that it is a statement about the (fixed) input data A and not the (randomized) algorithm. Moreover, it holds for arbitrary matrices A, *i.e.*, matrices that have an arbitrarily large number of columns and rows and not necessarily just tall-and-thin or short-and-fat matrices A, as was the case in the over- or under-determined least-squares problems. The structural property thus applies most directly to problems where one is interested in low-rank approximation with respect to a low-rank space of dimension $k \ll \min\{m, n\}$.

In RandNLA, the sketching matrix is typically either a matrix representing the operation of sampling columns or rows from the input matrix, or a matrix representing the random projection operation. In that case, this structural property has an interpretation in terms of how the sampling or projection operation interacts with the subspaces defined by the top and bottom part of the spectrum of A. We emphasize, however, that this structural property holds more generally: in particular, it holds for *any* (deterministic or randomized) "sketching" matrix and thus it is a property of independent interest. For example, while it is outside the scope of this chapter to discuss in detail, one can easily imagine using this property to derandomize RandNLA algorithms or to develop other deterministic matrix algorithms for these and related matrix problems or to develop improved heuristics in machine learning and data analysis applications. In the remainder of this chapter, we highlight this structural property, stating and presenting an analysis of a more general version of it than has been previously available. We also describe how it is used in several of the recent improvements to various RandNLA algorithms for low-rank matrix approximation problems.

3 Our main technical result

In this section, we state and prove our main technical result. This technical result is a structural condition that characterizes the interaction between the singular subspaces of the input matrix A and *any* deterministic or randomized "sketching" matrix.

[4] Although this structural property is central to all of the above problems, its role is typically obscured since it is often secondary to the main result of interest in a given paper, and thus it is hidden deep within the analysis of each of the superficially-different methods that use it. This property was first introduced by Boutsidis *et al.* [10] in the context of the CSSP, it was subsequently used by Halko *et al.* [44] to simplify the description of several related random projection algorithms, and it was then used—typically knowingly, but sometimes unknowingly—by many researchers working on these and other problems.

3.1 Statement of the main technical result

Recall that, given a matrix $A \in \mathbb{R}^{m \times n}$, many RandNLA algorithms seek to construct a "sketch" of A by post-multiplying A by some "sketching" matrix $Z \in \mathbb{R}^{n \times r}$, where r is much smaller than n. (For example, Z could represent the action of random sampling or random projection.) Thus, the resulting matrix $AZ \in \mathbb{R}^{m \times r}$ is matrix that is much smaller than the original matrix A, and the interesting question is what kind of approximation guarantees does it offer for A.

A common approach is to explore how well AZ spans the principal subspace of A, and one metric of accuracy is the error matrix, $A - P_{AZ}A$, where $P_{AZ}A$ is the projection of A onto the subspace spanned by the columns of AZ. Formally,

$$P_{AZ} = (AZ)(AZ)^{+} = U_{AZ}U_{AZ}^{T}.$$

Recall that $X^{+} \in \mathbb{R}^{n \times m}$ is the Moore-Penrose pseudoinverse of any matrix $X \in \mathbb{R}^{m \times n}$ and that it can be computed via the SVD of X; see [38] for details. Similarly, $U_{AZ} \in \mathbb{R}^{m \times \rho}$ is the matrix of the left singular vectors of AZ, where ρ is the rank of AZ. The following structural result offers a means to bound any unitarily invariant norm of the error matrix $A - P_{AZ}A$.

Lemma 1 *Given $A \in \mathbb{R}^{m \times n}$, let $Y \in \mathbb{R}^{n \times k}$ be any matrix such that $Y^{T}Y = I_k$. Let $Z \in \mathbb{R}^{n \times r}$ ($r \geq k$) be any matrix such that $Y^{T}Z$ and AY have full rank. Then, for any unitarily invariant norm ξ, we have that*

$$\|A - P_{AZ}A\|_{\xi} \leq \left\|A - AYY^{T}\right\|_{\xi} + \left\|\left(A - AYY^{T}\right) Z(Y^{T}Z)^{+}\right\|_{\xi}. \tag{1}$$

Three comments about this lemma, one regarding Z, one regarding Y, and one regarding the interaction between Z and Y, are in order.

- Lemma 1 holds for any matrix Z, regardless of whether Z is constructed deterministically or randomly. In the context of RandNLA, typical constructions of Z would represent a random sampling or random projection operation.
- The orthogonal matrix Y in the above lemma is also arbitrary. In the context of RandNLA, one can think of Y either as $Y = V_k$, where $V_k \in \mathbb{R}^{n \times k}$ is the matrix of the top k right singular vectors of A, or as some other orthogonal matrix that approximates V_k; but Lemma 1 holds more generally.
- As stated in Lemma 1, Y must satisfy two conditions: the matrix $Y^{T}Z$ must have full rank, equal to k, since $r \geq k$, and the matrix AY must also have full rank, again equal to k. If $Y = V_k$, then the constraint that AY must have full rank is trivially satisfied, assuming that A has rank at least k. Additionally, the sampling and random projection approaches that are used in high-quality RandNLA algorithms with sufficiently large values of r guarantee that the rank condition on $Y^{T}Z$ is satisfied [26, 49]. More generally, though, one could perform an *a posteriori* check that these two conditions hold.

3.2 A popular special case

Before providing a proof of this structural result, we will now consider a popular special case of Lemma 1. To do so, we will let $Y = V_k \in \mathbb{R}^{n \times k}$, namely the orthogonal matrix of the top k right singular vectors of A. (Actually, any orthogonal matrix spanning that same subspace would do in this discussion.) For notational convenience, we will let $V_{k,\perp}\mathbb{R}^{n \times (\rho - k)}$ (respectively, $\Sigma_{k,\perp} \in \mathbb{R}^{(n-\rho) \times (n-\rho)}$) be the matrix of the bottom $\rho - k$ right singular vectors (respectively,

singular values) of A. Let $A_k \in \mathbb{R}^{m\times n}$ be the best rank k approximation to A as computed by the SVD. It is well known that

$$A_k = AV_kV_k^T.$$

Assuming that V_k^TZ has full rank, then Lemma 1 implies that:

$$\|A - P_{AZ}A\|_\xi \leq \|A - A_k\|_\xi + \left\|(A - A_k)\, Z\left(V_k^TZ\right)^+\right\|_\xi.$$

Note here that

$$A - A_k = U_{k,\perp}\Sigma_{k,\perp}V_{k,\perp}^T$$

and if we drop, using unitary invariance, the matrix $U_{k,\perp}$ from the second norm at the right-hand side of the above inequality, then we get:

$$\|A - P_{AZ}A\|_\xi \leq \|A - A_k\|_\xi + \left\|\Sigma_{k,\perp}\left(V_{k,\perp}^TZ\right)\left(V_k^TZ\right)^+\right\|_\xi.$$

For the special case of $\xi \in \{2, F\}$, this is exactly the structural condition underlying the randomized low-rank projection algorithms of [44] that was first introduced in the context of the CSSP [10]. We summarize the above discussion in the following lemma.

Lemma 2 *Given $A \in \mathbb{R}^{m\times n}$, let $V_k \in \mathbb{R}^{n\times k}$ be the matrix of the top k right singular vectors of A. Let $Z \in \mathbb{R}^{n\times r}$ $(r \geq k)$ be any matrix such that Y^TZ has full rank. Then, for any unitarily invariant norm ξ,*

$$\|A - P_{AZ}A\|_\xi \leq \|A - A_k\|_\xi + \left\|\Sigma_{k,\perp}\left(V_{k,\perp}^TZ\right)\left(V_k^TZ\right)^+\right\|_\xi. \tag{2}$$

Eqn. (2) immediately suggests a proof strategy for bounding the error RandNLA algorithms for low-rank matrix approximation: identify a sketching matrix Z such that V_k^TZ has full rank; and, at the same time, bound the relevant norms of $\left(V_k^TZ\right)^+$ and $V_{k,\perp}Z$.

Lemma 1 generalizes the prior use of Lemma 2 in several important ways.

- First, it is not necessary to focus on random sampling matrices or random projection matrices, but instead we consider arbitrary sketching matrices Z. This was actually implicit in the analysis of the original version of Lemma 2 [10], but it seems worth making that explicit here. It does, however, require the extra condition that AZ also has full rank.
- Second, it is not necessary to focus on V_k and so we consider the more general case of any arbitrary orthogonal matrix $Y \in \mathbb{R}^{n\times k}$ instead of V_k.
- Third, it is not necessary to focus on the spectral or Frobenius norm, as it is straightforward to prove this result for any unitarily invariant matrix norm.

3.3 Proof of the main technical result

This proof of Lemma 1 follows our previous proof of Lemma 2 from [10], simplifying it and generalizing it at appropriate places. We start by noting that

$$\|A - P_{AZ}A\|_\xi = \left\|A - (AZ)\,(AZ)^+A\right\|_\xi. \tag{3}$$

Then, for any unitarily invariant norm ξ [45],

$$(AZ)^+A = \arg\min_{X\in R^{r\times n}} \|A - (AZ)\,X\|_\xi.$$

This implies that in Eqn. (3) we can replace $(AZ)^+ A$ with any other $r \times n$ matrix and the equality with an inequality. In particular we replace $(AZ)^+ A$ with $\left(AYY^TZ\right)^+ AYY^T$, where AYY^T is a rank-k approximation to A (not necessarily the best rank-k approximation to A):

$$\begin{aligned} \|A - P_{AZ}A\|_\xi &= \left\|A - AZ\,(AZ)^+ A\right\|_\xi \\ &\leq \left\|A - AZ\left(AYY^TZ\right)^+ AYY^T\right\|_\xi . \end{aligned}$$

This suboptimal choice for X is essentially the "heart" of our proof: it allows us to manipulate and further decompose the error term, thus making the remainder of the analysis feasible. Use $A = A - AYY^T + AYY^T$ and the triangle inequality to get

$$\begin{aligned} &\|A - P_{AZ}A\|_\xi \\ &\leq \left\|A - AYY^T + AYY^T - \left(A - AYY^T + AYY^T\right) Z \left(AYY^TZ\right)^+ AYY^T\right\|_\xi \\ &\leq \left\|A - AYY^T\right\|_\xi + \left\|AYY^T - AYY^TZ(AYY^TZ)^+AYY^T\right\|_\xi \\ &+ \left\|\left(A - AYY^T\right) Z(AYY^TZ)^+AYY^T\right\|_\xi . \end{aligned}$$

We now prove that the second term in the last inequality is equal to zero. Indeed,

$$\begin{aligned} &\left\|AYY^T - AYY^TZ(AYY^TZ)^+AYY^T\right\|_\xi \\ &\quad = \left\|AYY^T - AYY^TZ(Y^TZ)^+(AY)^+AYY^T\right\|_\xi \qquad (4) \\ &\quad = \left\|AYY^T - AYY^T\right\|_\xi = 0. \end{aligned}$$

In Eqn. (4), we replaced $\left(AYY^TZ\right)^+$ by $\left(Y^TZ\right)^+ (AY)^+$, using the fact that both matrices Y^TZ and AY have full rank. The fact that both matrices have full rank also implies

$$Y^TZ\left(Y^TZ\right)^+ = I_k \quad \text{and} \quad (AY)^+ AY = I_k,$$

which concludes the derivation. Using the same manipulations and dropping Y^T using unitary invariance, we get:

$$\left\|\left(A - AYY^T\right) Z(AYY^TZ)^+AYY^T\right\|_\xi = \left\|\left(A - AYY^T\right) Z(Y^TZ)^+\right\|_\xi ,$$

which concludes the proof.

4 Applications of our main technical result

In this section, we discuss several settings where exploiting the structural result highlighted in Lemma 1 results in improved analyses of RandNLA algorithms for low-rank matrix approximation problems.

4.1 The Column Subset Selection Problem (CSSP): theory

The special case of Lemma 2 corresponding to the spectral and Frobenius norm was first identified and established in our prior work on the CSSP [10]. The CSSP is the problem of choosing the "best" (in a sense that we will make precise shortly) set of r columns from an $m \times n$ matrix A. Given the importance of the CSSP in both NLA as well as TCS applications of RandNLA, here

we will describe in some detail the role of Eqn. (1) in this context, as well as related work. In the next section, we will describe applied aspects of the CSSP.

First of all, the CSSP can be formally defined as follows: given a matrix $A \in \mathbb{R}^{m \times n}$, one seeks a matrix $C \in \mathbb{R}^{m \times r}$ consisting of r columns of A such that

$$\left\| A - CC^{+}A \right\|_{\xi}$$

is minimized. While one could use any norm to measure the error $A - CC^{+}A$, the most common choices are $\xi = 2$ or $\xi = F$. Most of the early work on CSSP in the NLA literature focused on error bounds of the form

$$\left\| A - CC^{+}A \right\|_{\xi} \leq \alpha \left\| A - A_k \right\|_{\xi},$$

where A_k is the best rank k approximation to A. The objective was to make the multiplicative error factor α as small as possible. In this setting, the choice of r is critical, and almost all early work focused on $r = k$, namely the setting where *exactly* k columns of A are chosen in order to approximate the best rank-k approximation to the matrix. The first result in this domain goes back to Golub in the 1960's [37]. It was quickly followed by numerous papers in the NLA community studying algorithms and bounds for the CSSP, with a primary focus on the spectral norm ($\xi = 2$). Almost all the early papers analyzed deterministic, greedy approaches for the CSSP, including the landmark work by Gu and Eisenstat [41], which provided essentially optimal algorithms (in terms of α) for the spectral norm variant of the CSSP.

The work of [10, 8] was the first attempt to design a randomized algorithm for both the spectral and the Frobenius norm version of the CSSP. The fundamental contribution of [10, 8] was an early, simple version of the structural result of Eqn. (1), which allowed us to combine in a non-trivial way deterministic and randomized methods from the NLA and TCS communities for the CSSP. More specifically, Algorithm 1 (see also Section 4.2 where we will discuss this algorithm from a more applied perspective) is a two-phase approach that was proposed in order to identify k columns of A to be included in C: first, sample $O(k \log k)$ columns of A with respect to the leverage scores, a highly informative probability distribution over the columns of A that biases the sampling process towards important columns; and, second, using deterministic column subset selection algorithms, choose *exactly* k columns out of the $O(k \log k)$ columns sampled in the first phase. Deriving the error bounds for the proposed two-phase approach was done by bounding the second term of Eqn. (1) as follows: first, one bounds the relevant norm of $\left(A - YY^{T}A\right)Z$, where Y was equal to V_k and Z was a sampling matrix encoding both the randomized and the deterministic phase of the proposed algorithm; and then, a lower bound on the smallest singular value of the matrix $Y^{+}Z$ was also proven. The latter bound was derived by properties of the leverage score sampling as well as by properties of the deterministic column selection algorithm applied in the second phase. Submultiplicativity of unitarily invariant norms was finally used to conclude the proof. The work of [10, 8] provided a major improvement on previous bounds for the Frobenius norm error of the CSSP, showing that the proposed randomized algorithm achieves $\alpha = O\left(k \log^{\frac{1}{2}} k\right)$ with constant probability. Prior work had exponential dependencies on k.

The above bound motivated Deshpande and Rademacher [20] to look at the CSSP using the so-called *volume sampling* approach. They designed and analyzed an approximation algorithm that guaranteed $\alpha = \sqrt{k+1}$ for the Frobenius norm, running in time $O(knm^3 \log m)$. This algorithm matched a lower bound for the CSSP presented in [21]. It is worth noting that [20] also presented faster versions of the above algorithm. The current state-of-the-art approach (in terms of speed) appeared in the work of [42], who presented a randomized algorithm that runs in $O\left(knm^2\right)$ time and guarantees $\alpha = \sqrt{k+1}$, with constant probability. Neither of these papers use the inequality that we discuss here. It would be interesting to understand how one could leverage structural results in order to prove the above bounds.

Input: $A \in \mathbb{R}^{m \times n}$, integer $k \ll \min\{m, n\}$.
Output: $C \in \mathbb{R}^{m \times k}$ with k columns of A.

1. **Randomized Stage:**
   - Let $V_k \in \mathbb{R}^{n \times k}$ be *any* orthogonal basis spanning the top-k right singular subspace of A.
   - Compute the sampling probabilities p_i for all $i = 1 \dots n$:
   $$p_i = \frac{1}{k} \left\| \left(V_k^T\right)^{(i)} \right\|_2^2, \tag{5}$$
   where $\left(V_k^T\right)^{(i)}$ denotes the i-th column of V_k^T as a column vector.
   - Randomly select and rescale $c = O(k \log k)$ columns of V_k^T according to these probabilities.
2. **Deterministic Stage:**
   - Let $\tilde{V}^T$ be the $k \times c$ non-orthogonal matrix consisting of the down-sampled and rescaled columns of V_k^T.
   - Run a deterministic QR algorithm on $\tilde{V}^T$ to select exactly k columns of $\tilde{V}^T$.
   - Return the corresponding columns of A.

Algorithm 1: A two-stage algorithm for the CSSP.

We now consider the relaxation of the CSSP where r is allowed to be greater than k. In this framework, when $r = \Omega(k \log k/\epsilon)$, relative-error approximations, namely approximations where $\alpha = 1 + \epsilon$, are known. For example, [29, 30] presented the first result that achieved such a bound, using random sampling of the columns of A according to the Euclidean norms of the rows of V_k, which are the leverage scores that we mentioned earlier in this chapter. More specifically, a $(1+\epsilon)$-approximation was proven by setting $r = \Omega\left(k\epsilon^{-2} \log\left(k\epsilon^{-1}\right)\right)$. Subsequently, [63] argued that the same technique gives a $(1+\epsilon)$-approximation using $r = \Omega\left(k \log k + k\epsilon^{-1}\right)$ columns, and this improved the running time by essentially computing approximations to the singular vectors of A. It is precisely in this context that the matrix Y of Eqn. (1) would be useful, since it would allow us to work with approximation to the singular vectors of A. While neither of these papers used the structural result of Eqn. (1) explicitly, they both implicitly had to follow similar derivations. As a matter of fact, Eqn. (1) could be a starting point for both papers and with little additional work could result in constant factor approximations. However, in order to get relative error bounds, additional care and more technical details are necessary. A long line of work followed [29, 63] showing alternative algorithms, often with improved running times, that achieve comparable relative error bounds [22, 32, 33, 65].

A major open question on the CSSP was whether one could derive meaningful error bounds for values of r that are larger than k but smaller than $O(k \log k)$. Towards that end, the first major breakthrough allowing sampling of fewer than $O(k \log k)$ columns appeared in [6, 7], where it was proven that by setting $r = 2k/\epsilon$ (up to lower order terms) one can achieve relative error approximations to the CSSP. Once more, structural inequalities along the lines of Eqn. (1) were at the forefront, combined with a novel column selection procedure invented by Batson *et al.* [3].

Using the structural inequality *per se* would only result in a constant factor approximation, but an additional adaptive sampling step guaranteed the required relative error approximation. Followup work by Guruswami and Sinop [42] presented algorithms based on volume sampling that set $r = k/\epsilon$ (up to lower order terms), thus exactly matching known lower bounds for the CSSP when $r > k$. The running time of all these algorithms is at least linear in the dimensions of the input matrix, but recent progress on subspace preserving embeddings that run in input sparsity time has removed this dependency. We refer the interested reader to [17, 53, 56] for RandNLA algorithms that run in input sparsity time, plus lower-order terms.

4.2 The Column Subset Selection Problem (CSSP): data analysis and machine learning

The CSSP algorithm of [10] has also been applied in several machine learning and data analysis applications, *e.g.*, see [9, 11, 12, 60, 59, 46]. In this section, we informally describe our experiences when using such approaches in data analysis and machine learning tasks. Our objective here is to provide some insight as to what is going on "under the hood" with this method as well as provide some speculation to justify its success in applications.

Recall Algorithm 1, our two-stage hybrid algorithm for the CSSP [10], and note that both the original choice of columns in the first phase, as well as the application of the QR algorithm in the second phase, involve the matrix V_k^T rather than the matrix A itself. In words, V_k^T is the matrix defining the relevant non-uniformity structure over the columns of A [30, 50]. The analysis of this algorithm (a large part of which boiled down to the proof of Lemma 2) makes critical use of exact or approximate versions of the importance sampling probabilities given in Eqn. (5). These are a generalization of the concept of *statistical leverage scores*; see [50, 49] as well as [66, 15, 14] for a detailed discussion. Here, we note informally that leverage scores capture a notion of "outlierness" *or* the extent to which data points are "spread out" *or* the "influence" of data points in low-rank and least-squares approximation problems.

Observe that the second stage of Algorithm 1 involves a QR computation. It is critical to the success of this algorithm to apply this QR procedure on the randomly-sampled version of V_k^T, *i.e.*, the matrix defining the worst-case non-uniformity structure in A, rather than on A itself. We have also observed the importance of this empirically. To understand this, recall that an important aspect of different QR algorithms is how they make so-called pivot rule decisions about which columns to keep [39]; and recall also that such decisions can be tricky when the columns in the matrix that is input to the QR algorithm are not orthogonal or spread out in similarly "nice" ways (*e.g.*, when it is the case that two columns are approximately, but not exactly, collinear). With this in mind, here are several empirical observations we have made that shed light on the inner workings of the CSSP algorithm and its usefulness in applications.

- Since the QR decomposition can be used to solve directly the CSSP, we investigated several alternative algorithms for the QR decomposition; and we also compared each QR alternative to the CSSP using that version of QR in the second phase. An initial observation was that "off-the-shelf" implementations of alternative algorithms for the QR decomposition behave quite differently—*e.g.*, some versions such as the Low-RRQR algorithm of [13] tend to perform much better than other versions such as the qrxp algorithm of [5, 4]. Although not surprising to NLA practitioners, this observation indicates that "off-the-shelf" implementations in large-scale data applications should be used carefully. A second, less obvious, observation is that preprocessing with the randomized first phase tends to improve worse-performing variants of QR more than better variants. Part of this is simply due to the fact that the worse-performing variants have more room to improve, but part of

this is also due to the fact that more sophisticated versions of QR tend to make elaborate pivot rule decisions. This sophistication is relatively less important after the randomized phase has selected columns that are already spread out and biased towards the important or outlying directions.

- To understand better the role of randomness in the algorithm, we also investigated the effect of applying algorithms for the QR decomposition directly on V_k^T (without running the randomized phase first) and then keeping the corresponding columns of A. Interestingly, with this "preprocessing" we tended to get better columns than if we ran QR decomposition algorithms directly on the original matrix A. Again, the interpretation seems to be that, since the norms of the columns of V_k^T define the relevant nonuniformity structure of A, working directly with those columns tends to avoid (even in traditional deterministic settings) situations where pivot rules fail to choose good columns.

- Of course, we also observed that randomization further improves the results, assuming that care is taken in choosing the rank parameter k and the sampling parameter c. In practice, the choice of k should be viewed as a "model selection" question. By choosing $c = k, 1.5k, 2k, \ldots$, we often observed a "sweet spot," in a bias-variance sense, as a function of increasing c. That is, for a fixed k, the behavior of the deterministic QR algorithms improves by choosing somewhat more than k columns, but that improvement is degraded by choosing too many columns in the randomized phase.

4.3 Random projections for low-rank matrix approximation

There has been massive interest recently in implementing random projection algorithms for use in scientific computing applications. One thing that has enabled this is that the structural condition identified in Lemma 2 makes it easier to parameterize RandNLA algorithms in terms more familiar to the NLA and scientific computing communities (and thus this was a very important step in the development of practically-useful RandNLA methods for low-rank matrix approximation.) To see how this relates to our main technical result, consider the following basic random projection algorithm. Given a matrix $A \in \mathbb{R}^{m \times n}$ and a rank parameter k:

- Construct an $n \times \ell$, with $\ell = O(k/\epsilon)$, structured random projection matrix Ω, *e.g.*, uniformly sample a few rows from a randomized Hadamard transform (see, for example, [31] for a precise definition of the randomized Hadamard transform).

- Return $B = A\Omega$.

This algorithm, which amounts to choosing uniformly at random a small number ℓ of columns in a randomly rotated basis, was introduced in [63], where it is proven that

$$\|A - P_{B_k} A\|_F \leq (1+\epsilon) \|A - P_{U_k} A\|_F \,, \tag{6}$$

where $P_{B_k} A$ is the projection of A onto the best rank-k approximation of B, holds with high probability. This bound, which is the random projection analogue of the relative-error CUR matrix approximations of [30, 50], provides a bound only on the reconstruction error of the top part of the spectrum of the input matrix. Additionally, it necessitates sampling a relatively large number of columns $\ell = O(k/\epsilon)$.

In many practical applications, *e.g.*, when providing high-quality numerical implementations, it is preferable to parameterize the problem in order to choose some number $\ell = k + p$ columns, where p is a modest additive oversampling factor, *e.g.*, p is equal to 10 or 20 or k. When attempting

to be this aggressive at minimizing the size of the sample, the choice of the oversampling factor p is quite sensitive to the input. That is, whereas the bound of Eqn. (6) holds for any worst-case input, here the proper choice for the oversampling factor p could depend on the matrix dimensions, the decay properties of the spectrum, and the particular choice made for the random projection matrix [52, 69, 47, 61, 44, 43].

To deal with these issues, the best numerical implementations of RandNLA algorithms for low-rank matrix approximation, and those that obtain the strongest results in terms of minimizing p, take advantage of Lemma 2 in a somewhat different way than was originally used in the analysis of the CSSP. For example, rather than choosing $O(k \log k)$ dimensions and then filtering them through *exactly* k dimensions, one can choose some number ℓ of dimensions and project onto a k'-dimensional subspace, where $k < k' \leq \ell$, while exploiting Lemma 2 to bound the error, as appropriate for the computational environment at hand [44].

Consider, for example, the following random projection algorithm. Given $A \in \mathbb{R}^{m \times n}$, a rank parameter k, and an oversampling factor p:

- Set $\ell = k + p$.
- Construct an $n \times \ell$ random projection matrix Ω, either with i.i.d. Gaussian entries or in the form of a structured random projection such as uniformly sampling a few rows from a randomized Hadamard transform.
- Return $B = A\Omega$.

Although this approach is quite similar to the algorithms of [58, 63], algorithms parameterized in this form were first introduced in [52, 69, 47], where a suite of bounds of the form

$$\|A - Z\|_2 \lesssim 10\sqrt{\ell \min\{m, n\}}\, \|A - A_k\|_2$$

were shown to hold with high probability. Here, Z is a rank-k-or-greater matrix, easily-constructed from B. Such results can be used to obtain the so-called *interpolative decomposition*, a variant of the basic CSSP with explicit numerical conditioning properties, and [52, 69, 47] also provided *a posteriori* error estimates that are useful in situations where one wants to choose the rank parameter k to be the numerical rank, as opposed to *a priori* specifying k as part of the input. Such *a priori* choices were more common in TCS algorithms for the same problem that predated the aforementioned approach.

Consider, in addition, how the following random projection algorithm addresses the issue that the decay properties of the spectrum can be important when it is of interest to aggressively minimize the oversampling parameter p. Given a matrix $A \in \mathbb{R}^{m \times n}$, a rank parameter k, an oversampling factor p, and an iteration parameter q:

- Set $\ell = k + p$.
- Construct an $n \times \ell$ random projection matrix Ω, either with i.i.d. Gaussian entries or in the form of a structured random projection such as uniformly sampling a few rows from a randomized Hadamard transform.
- Return $B = (AA^T)^q A\Omega$.

This algorithm, as well as a numerically-stable variant of it, was introduced in [61], where it was shown that bounds of the form

$$\|A - Z\|_2 \lesssim \left(10\sqrt{\ell \min\{m, n\}}\right)^{1/(4q+2)}\, \|A - A_k\|_2$$

hold with high probability. Again, Z is a rank-k-or-greater matrix easily-constructed from B; and this bound should be compared with the bound of the previous algorithm. Basically, this random projection algorithm modifies the previous algorithm by coupling a form of the power iteration method within the random projection step and, in many cases, it leads to improved performance [61, 44].

In their review, [44] used Lemma 2 to clarify and simplify these and other prior random projection methods. (Subsequent work, *e.g.*, that of [40] which develops RandNLA algorithms within the subspace iteration framework, has continued to use Lemma 2 in somewhat different ways.) Lemma 2 was explicitly reproven (with squares in the norms) in [44], using a proof based on the perturbation theory of orthogonal projectors, thus providing an elegant alternative to the original proof of the inequality. Our inequality in Lemma 2 was an essential ingredient of their work, allowing the authors of [44] to bound the performance of their algorithms based on the relationship between the singular vectors corresponding to the large singular values of A and their counterparts corresponding to the small singular values of A. As the authors of [44] observe, "when a substantial proportion of the mass of A appears in the small singular values, the constructed basis may have low accuracy. Conversely, when the large singular values dominate, it is much easier to identify a good low-rank basis." Our main inequality, originally developed within the context of the CSSP, precisely quantifies this tradeoff in a strong sense and it serves as a starting point and foundation for the RandNLA theory reviewed in [44].

4.4 Improved results for Nyström-based machine learning.

Symmetric positive semi-definite (SPSD) matrices are of interest in many applications, in particular for so-called kernel-based machine learning methods [64]. In many situations, matrices of interest are moderately well approximated by low-rank matrices, and in many of these cases one is interested in so-called Nyström-based low-rank matrix approximation [67, 27, 36]. These are low-rank matrix approximations that are expressed in terms of actual columns and rows, *i.e.*, they are essentially CSSP methods for SPSD matrices that preserve the SPSD property. A challenge here is that, while CSSP methods provide high-quality bounds for general matrices, it is difficult to preserve the SPSD property and thus extend these to provide high-quality SPSD low-rank approximation of SPSD matrices. Indeed, early work on Nyström methods was either heuristic [67] or provided rigorous but weak worst-case theory [27].

A qualitative improvement in this area occurred with Gittens and Mahoney [36], which used a result from Gittens [35] to preserve the SPSD property, while working with leverage-based column sampling and related random projection methods. A critical component of the analysis of [36] involved providing structural decompositions which are variants of Lemma 2 for SPSD matrices for the spectral, Frobenius, and trace norms. Subsequent to this, Anderson *et al.* [1] introduced the so-called spectral gap error bound method to provide still finer results in a common case: namely, when one performs a very modest amount of oversampling for input kernel matrices that *do not have* a large spectral gap, but that *do have* a spectrum that decays rapidly. The analysis of [1] used a result from Gu [40] that extended Lemma 2 by providing an analogous structural statement when one is interested in splitting the matrix into three parts: the top, middle, and bottom (rather than just top and bottom) parts of the spectrum. In each of these cases, increasingly finer results are derived for several related problems by exploiting structural properties having to do with the interaction of sampling/projection operators in the RandNLA algorithms with various parts of the vector space defined by the input matrix.

5 Conclusions

The interdisciplinary history of RandNLA has seen a gradual movement toward providing increasingly-finer bounds for a range of low-rank (and other) matrix problems. In this chapter, we have highlighted, described, and extended a deterministic structural result underlying many state-of-the-art RandNLA algorithms for low-rank matrix approximation problems. A general theme in this development is that this is accomplished by using general algorithmic and statistical tools and specializing them to account for the fine-scale structure of the Euclidean vector space defined by the data matrix. For example, while a vanilla application of the Johnson-Lindenstrauss lemma, which is applicable to vectors in general metric spaces, leads to interesting results (*e.g.*, additive-error bounds on the top part of the spectrum of the matrix being approximated), much stronger results (*e.g.*, relative-error bounds, as well as the CSSP results that first introduced the predecessor of Lemma 1, as well as the other results we have reviewed here) can be obtained by exploiting the vector space structure of the Euclidean spaces defined by the top and bottom parts of the spectrum of A.

A challenge in interdisciplinary research areas such as RandNLA is that algorithms solving seemingly different problems use similar structural results in various ways. At the same time, diverse research areas study those problems from many different perspectives. As a result, highlighting structural commonalities is rare and such structural results usually get "buried" deep inside the technical analysis of the proposed methods. Highlighting the central role of such structural results is important, especially as RandNLA methods are increasingly being applied to data analysis tasks in applications ranging from genetics [60, 59, 46] to astronomy [73] and mass spectrometry imaging [72] and as RandNLA algorithms are increasingly being implement in large-scale parallel and distributed computational environments [55, 54, 70, 71].

References

[1] D. G. Anderson, S. S. Du, M. W. Mahoney, C. Melgaard, K. Wu, , and M. Gu. Spectral gap error bounds for improving CUR matrix decomposition and the Nyström method. In *Proceedings of the 18th International Workshop on Artificial Intelligence and Statistics*, pages 19–27, 2015.

[2] H. Avron, P. Maymounkov, and S. Toledo. Blendenpik: Supercharging LAPACK's least-squares solver. *SIAM Journal on Scientific Computing*, 32:1217–1236, 2010.

[3] 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, 2009.

[4] C. H. Bischof and G. Quintana-Ortí. Algorithm 782: Codes for rank-revealing QR factorizations of dense matrices. *ACM Transactions on Mathematical Software*, 24(2):254–257, 1998.

[5] C. H. Bischof and G. Quintana-Ortí. Computing rank-revealing QR factorizations of dense matrices. *ACM Transactions on Mathematical Software*, 24(2):226–253, 1998.

[6] C. Boutsidis, P. Drineas, and M. Magdon-Ismail. Near-optimal column-based matrix reconstruction. In *Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science*, pages 305–314, 2011.

[7] C. Boutsidis, P. Drineas, and M. Magdon-Ismail. Near-optimal column-based matrix reconstruction. *SIAM J. Comput.*, 43(2):687–717, 2014.

[8] C. Boutsidis, M. W. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. Technical report. Preprint: arXiv:0812.4293 (2008).

[9] C. Boutsidis, M. W. Mahoney, and P. Drineas. Unsupervised feature selection for principal components analysis. In *Proceedings of the 14th Annual ACM SIGKDD Conference*, pages 61–69, 2008.

[10] C. Boutsidis, M. W. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 968–977, 2009.

[11] C. Boutsidis, M. W. Mahoney, and P. Drineas. Unsupervised feature selection for the k-means clustering problem. In *Annual Advances in Neural Information Processing Systems 22: Proceedings of the 2009 Conference*, 2009.

[12] C. Boutsidis, A. Zouzias, M. W. Mahoney, and P. Drineas. Randomized dimensionality reduction for k-means clustering. *IEEE Transactions on Information Theory*, 61(2):1045–1062, 2015.

[13] T. F. Chan and P.C. Hansen. Low-rank revealing QR factorizations. *Numerical Linear Algebra with Applications*, 1:33–44, 1994.

[14] S. Chatterjee and A. S. Hadi. Influential observations, high leverage points, and outliers in linear regression. *Statistical Science*, 1(3):379–393, 1986.

[15] S. Chatterjee and A.S. Hadi. *Sensitivity Analysis in Linear Regression*. John Wiley & Sons, New York, 1988.

[16] K. L. Clarkson, P. Drineas, M. Magdon-Ismail, M. W. Mahoney, X. Meng, and D. P. Woodruff. The Fast Cauchy Transform and faster robust linear regression. In *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 466–477, 2013.

[17] K. L. Clarkson and D. P. Woodruff. Low rank approximation and regression in input sparsity time. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, pages 81–90, 2013.

[18] E. S. Coakley, V. Rokhlin, and M. Tygert. A fast randomized algorithm for orthogonal projection. *SIAM Journal on Scientific Computing*, 33(2):849–868, 2011.

[19] National Research Council. *Frontiers in Massive Data Analysis*. The National Academies Press, Washington, D. C., 2013.

[20] A. Deshpande and L. Rademacher. Efficient volume sampling for row/column subset selection. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science*, pages 329–338, 2010.

[21] A. Deshpande, L. Rademacher, S. Vempala, and G. Wang. Matrix approximation and projective clustering via volume sampling. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1117–1126, 2006.

[22] A. Deshpande and S. Vempala. Adaptive sampling and fast low-rank matrix approximation. In *Proceedings of the 10th International Workshop on Randomization and Computation*, pages 292–303, 2006.

[23] P. Drineas, R. Kannan, and M. W. Mahoney. Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication. *SIAM Journal on Computing*, 36:132–157, 2006.

[24] P. Drineas, R. Kannan, and M. W. Mahoney. Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix. *SIAM Journal on Computing*, 36:158–183, 2006.

[25] P. Drineas, R. Kannan, and M. W. Mahoney. Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition. *SIAM Journal on Computing*, 36:184–206, 2006.

[26] P. Drineas, M. Magdon-Ismail, M. W. Mahoney, and D. P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13:3475–3506, 2012.

[27] P. Drineas and M. W. Mahoney. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6:2153–2175, 2005.

[28] 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.

[29] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Subspace sampling and relative-error matrix approximation: Column-based methods. In *Proceedings of the 10th International Workshop on Randomization and Computation*, pages 316–326, 2006.

[30] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30:844–881, 2008.

[31] P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, 2010.

[32] D. Feldman and M. Langberg. A unified framework for approximating and clustering data. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing*, pages 569–578, 2011.

[33] D. Feldman, M. Monemizadeh, C. Sohler, and D. P. Woodruff. Coresets and sketches for high dimensional subspace approximation problems. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 630–649, 2010.

[34] A. Frieze, R. Kannan, and S. Vempala. Fast Monte-Carlo algorithms for finding low-rank approximations. In *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science*, pages 370–378, 1998.

[35] A. Gittens. The spectral norm error of the naive Nyström extension. Technical report, 2011. Preprint: arXiv:1110.5305.

[36] A. Gittens and M. W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. Technical report, 2013. Preprint: arXiv:1303.1849.

[37] G. Golub. Numerical methods for solving linear least squares problems. *Numerische Mathematik*, 7:206–216, 1965.

[38] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 1989.

[39] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 1996.

[40] M. Gu. Subspace iteration randomization and singular value problems. Technical report, 2014. Preprint: arXiv:1408.2208.

[41] M. Gu and S. C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal on Scientific Computing*, 17:848–869, 1996.

[42] V. Guruswami and A. K. Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1207–1214, 2012.

[43] N. Halko, P.-G. Martinsson, Y. Shkolnisky, and M. Tygert. An algorithm for the principal component analysis of large data sets. Technical report. Preprint: arXiv:1007.5510 (2010).

[44] N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[45] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, 1985.

[46] A. Javed, P. Drineas, M. W. Mahoney, and P. Paschou. Efficient genomewide selection of PCA-correlated tSNPs for genotype imputation. *Annals of Human Genetics*, 75(6):707–722, 2011.

[47] E. Liberty, F. Woolfe, P.-G. Martinsson, V. Rokhlin, and M. Tygert. Randomized algorithms for the low-rank approximation of matrices. *Proc. Natl. Acad. Sci. USA*, 104(51):20167–20172, 2007.

[48] P. Ma, M. W. Mahoney, and B. Yu. A statistical perspective on algorithmic leveraging. *In press at: Journal of Machine Learning Research*.

[49] M. W. Mahoney. *Randomized algorithms for matrices and data*. Foundations and Trends in Machine Learning. NOW Publishers, Boston, 2011.

[50] M. W. Mahoney and P. Drineas. CUR matrix decompositions for improved data analysis. *Proc. Natl. Acad. Sci. USA*, 106:697–702, 2009.

[51] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers. Big data: The next frontier for innovation, competition, and productivity. Technical report, McKinsey Global Institute, 2011.

[52] P.-G. Martinsson, V. Rokhlin, and M. Tygert. A randomized algorithm for the decomposition of matrices. *Applied and Computational Harmonic Analysis*, 30:47–68, 2011.

[53] X. Meng and M. W. Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, pages 91–100, 2013.

[54] X. Meng and M. W. Mahoney. Robust regression on MapReduce. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[55] X. Meng, M. A. Saunders, and M. W. Mahoney. LSRN: A parallel iterative solver for strongly over- or under-determined systems. *SIAM Journal on Scientific Computing*, 36(2):C95–C118, 2014.

[56] J. Nelson and N. L. Huy. OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science*, pages 117–126, 2013.

[57] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: a probabilistic analysis. In *Proceedings of the 17th ACM Symposium on Principles of Database Systems*, pages 159–168, 1998.

[58] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: a probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235, 2000.

[59] P. Paschou, J. Lewis, A. Javed, and P. Drineas. Ancestry informative markers for fine-scale individual assignment to worldwide populations. *Journal of Medical Genetics*, page doi:10.1136/jmg.2010.078212, 2010.

[60] P. Paschou, E. Ziv, E. G. Burchard, S. Choudhry, W. Rodriguez-Cintron, M. W. Mahoney, and P. Drineas. PCA-correlated SNPs for structure identification in worldwide human populations. *PLoS Genetics*, 3:1672–1686, 2007.

[61] V. Rokhlin, A. Szlam, and M. Tygert. A randomized algorithm for principal component analysis. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1100–1124, 2009.

[62] V. Rokhlin and M. Tygert. A fast randomized algorithm for overdetermined linear least-squares regression. *Proc. Natl. Acad. Sci. USA*, 105(36):13212–13217, 2008.

[63] T. Sarlós. Improved approximation algorithms for large matrices via random projections. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 143–152, 2006.

[64] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[65] N. D. Shyamalkumar and K. Varadarajan. Efficient subspace approximation algorithms. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 532–540, 2007.

[66] P.F. Velleman and R.E. Welsch. Efficient computing of regression diagnostics. *The American Statistician*, 35(4):234–242, 1981.

[67] C.K.I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Annual Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference*, pages 682–688, 2001.

[68] D. P. Woodruff. *Sketching as a Tool for Numerical Linear Algebra*. Foundations and Trends in Theoretical Computer Science. NOW Publishers, Boston, 2014.

[69] F. Woolfe, E. Liberty, V. Rokhlin, and M. Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, 2008.

[70] J. Yang, X. Meng, and M. W. Mahoney. Quantile regression for large-scale applications. *SIAM Journal on Scientific Computing*, 36:S78–S110, 2014.

[71] J. Yang, X. Meng, and M. W. Mahoney. Implementing randomized matrix algorithms in parallel and distributed environments. Technical report, 2015. Preprint: arXiv:1502.03032.

[72] J. Yang, O. Rubel, Prabhat, M. W. Mahoney, and B. P. Bowen. Identifying important ions and positions in mass spectrometry imaging data using CUR matrix decompositions. *Accepted for publication in: Analytical Chemistry*.

[73] C.-W. Yip, M. W. Mahoney, A. S. Szalay, I. Csabai, T. Budavari, R. F. G. Wyse, and L. Dobos. Objective identification of informative wavelength regions in galaxy spectra. *The Astronomical Journal*, 147(110):15pp, 2014.