Low Rank Approximation Directed by Leverage Scores and Computed at Sub-linear Cost

Qi Luan[1],[a], Victor Y. Pan[1,2],[b], and John Svadlenka[2],[c]

[1] Ph.D. Programs in Computer Science and Mathematics
The Graduate Center of the City University of New York
New York, NY 10036 USA

[2] Department of Computer Science
Lehman College of the City University of New York
Bronx, NY 10468 USA

[a] qi_luan@yahoo.com
[b] victor.pan@lehman.cuny.edu
http://comet.lehman.cuny.edu/vpan/
[c] jsvadlenka@gradcenter.cuny.edu

Abstract

Low rank approximation (*LRA*) of a matrix is a major subject of matrix and tensor computations and data mining and analysis. It is desired (and even imperative in applications to Big Data) to solve the problem at *sub-linear cost*, involving much fewer memory cells and arithmetic operations than an input matrix has entries, but this is impossible even for a small matrix family of our Appendix A. Nevertheless we prove that this is possible with a high probability (hereafter *whp*) for random matrices admitting LRA. Namely we recall the known randomized algorithms that solve the LRA problem whp for any matrix admitting LRA by means of random sampling directed by sampling probabilities called leverage scores. The computation of leverage scores has super-linear cost, but we trivialize that stage and prove that whp the resulting super-linear cost algorithm still outputs accurate LRA of a random input matrix admitting LRA whp.

Key Words: Low-rank approximation, Sub-linear cost, Subspace sampling, Leverage scores

2000 Math. Subject Classification: 65Y20, 65F30, 68Q25, 68W20, 15A52

1 Introduction: LRA at sub-linear cost: background and our progress

LRA of a matrix is one of the most fundamental problems of Numerical Linear and Multilinear Algebra and Data Mining and Analysis, with applications ranging from machine learning theory and neural networks to term document data and DNA SNP data (see surveys [HMT11], [M11], and

[KS16]). Matrices representing Big Data (e.g., unfolding matrices of multidimensional tensors) are usually so immense that realistically one can only access and process a tiny fraction of their entries, but quite typically these matrices admit LRA, that is, are close to low rank matrices,[1] with which one can operate by using *sub-linear arithmetic time and memory space*, that is, much fewer flops and memory cells than the matrix has entries.

Every LRA algorithm running at sub-linear cost fails on the worst case inputs and even on the small families of matrices of our Appendix A. For decades of worldwide computational practice, however, Cross–Approximation algorithms, running at sub-linear cost, routinely compute accurate LRA of matrices admitting LRA.[2] The papers [PLSZ16], [PLSZ17], [PLSZa], [PLSZb], and [PLa] provide some formal support for this empirical observations and extend it to some other (and in some cases even more primitive) LRA algorithms running at sub-linear cost.[3]

Now we present similar formal results on sub-linear cost computation of CUR LRA by means of *subspace sampling* directed by sampling probabilities, known as *leverage scores*.

The known algorithms of this class output nearly optimal LRA with a high probability (hereafter *whp*) and run at sub-linear cost except for the stage of computing leverage scores, performed at linear or super-linear cost. We trivialize that stage decreasing the overall computational cost to sub-linear level and then prove that whp the resulting algorithms still output reasonably close CUR LRA of a random input matrix that admits LRA. Moreover for all such matrices our algorithms can compute close CUR LRA at sub-linear cost whp if they are pre-processed with Gaussian multipliers. The cost of performing such pre-processing is not sub-linear, but empirically pre-processing at sub-linear cost with quasi Gaussian and other sparse multipliers works as efficiently.

Our main result about accuracy of subspace sampling at sub-linear cost applies in a certain neighborhood of low rank matrices specified by a restriction on the perturbation of singular vectors, according to the well-known bounds by Davis-Kahan 1970 and Wedin 1972 (see Theorem 3.2, Remark 3.3, and Section 5).

All known algorithms for the computation of LRA by means of subspace sampling directed by leverage scores sample a large number of rows and columns of an input matrix, and so does our acceleration of these algorithms. [TYUC17, Section 1.7.3] consider such a property an obstacle for practical application of these algorithms (see), but [PLa, Algorithm 3.1] provides partial remedy at sub-linear cost.

We organize our paper as follows. We devote the next section to background for LRA. In Section 3 we recall subspace sampling algorithms of [DMM08], directed by leverage scores. In Section 4 we prove our main result that its variation running at sub-linear cost is accurate whp for a random input. In Section 5, the contribution of the third author, we cover our tests of the perturbations of leverage scores caused by the perturbation of some real world inputs. In Appendix A we describe a small input families that are hard for any LRA algorithm that runs at sub-linear cost. In Appendix B we cover background on random matrices. In Appendix C we recall the auxiliary algorithms for random sampling and re-scaling from [DMM08].

[1] Here and throughout we use such concepts as "low", "small", "nearby", etc. defined in context.

[2] The algorithms output CUR LRA, which is a particularly memory efficient form of LRA and is proved to universal: [PLSZa] specifies transformation of any LRA into CUR LRA at sub-linear cost.

[3] The pioneering papers [PLSZ16] and [PLSZ17] provide first formal support for LRA at sub-linear cost, which they call "superfast" LRA.

2 Background for LRA

2.1 Matrix norms, pseudo inverse, and SVD

For simplicity we assume dealing with real matrices in $\mathbb{R}^{p\times q}$ throughout, but our study can be quite readily extended to complex matrices; in particular see [D88], [E88], [CD05], [ES05], and [TYUC17] for some relevant results about complex Gaussian matrices.

Hereafter M^+ denotes the Moore–Penrose pseudo inverse of M, $||\cdot||$ denotes the spectral norm, $||\cdot||_F$ the Frobenius norm, and $|\cdot|$ is our unified notation for both of these norms.

Lemma 2.1. [The norm of the pseudo inverse of a matrix product.] *Suppose that $A \in \mathbb{R}^{k\times r}$, $B \in \mathbb{R}^{r\times l}$ and the matrices A and B have full rank $r \le \min\{k, l\}$. Then $|(AB)^+| \le |A^+|\ |B^+|$.*

r-top SVD of a matrix M of rank at least r is the decomposition $M_r = U_r\Sigma_r V_r^*$ for the diagonal matrix $\Sigma_r = \mathrm{diag}(\sigma_j)_{j=1}^r$ of the r largest singular values of M and two unitary matrices U_r and V_r of the r associated top left and right singular vectors, respectively.[4]

M_r is said to be the *r-truncation* of M.

For a rank-r matrix its r-top SVD is just its *compact SVD*.

2.2 2-factor LRA

A matrix M has ξ-rank at most r if it admits approximation within an error norm ξ by a matrix M' of rank at most r or equivalently if there exist three matrices A, B and E such that

$$M = M' + E \text{ where } |E| \le \xi,\ M' = AB,\ A \in \mathbb{R}^{m\times r},\ \text{and } B \in \mathbb{R}^{r\times n}. \tag{2.1}$$

The 0-rank is the rank; the ξ-rank of a matrix M for a small tolerance ξ is said to be its *numerical rank*, hereafter denoted $\mathrm{nrank}(M)$. A matrix admits its close approximation by a matrix of rank at most r if and only if it has numerical rank at most r.

Theorem 2.1. [GL13, Theorem 2.4.8].) *Write $\tau_{r+1}(M) := \min_{N:\ \mathrm{rank}(N)=r} |M - N|$. Then $\tau_{r+1}(M) = |M - M_r|$ under both spectral and Frobenius norms: $\tau_{r+1}(M) = \sigma_{r+1}(M)$ under the spectral norm and $\tau_{r+1}(M) = \sigma_{F,r+1}(M) := \sum_{j\ge r}\sigma_j^2(M)$ under the Frobenius norm.*

2.3 Canonical CUR LRA and 3-factor LRA

For two sets $\mathcal{I} \subseteq \{1,\dots,m\}$ and $\mathcal{J} \subseteq \{1,\dots,n\}$ define the submatrices

$$M_{\mathcal{I},:} := (m_{i,j})_{i\in\mathcal{I};j=1,\dots,n}, M_{:,\mathcal{J}} := (m_{i,j})_{i=1,\dots,m;j\in\mathcal{J}},\ \text{and } M_{\mathcal{I},\mathcal{J}} := (m_{i,j})_{i\in\mathcal{I};j\in\mathcal{J}}.$$

Given an $m\times n$ matrix M of rank r and its nonsingular $r\times r$ submatrix $G = M_{\mathcal{I},\mathcal{J}}$ one can readily verify that $M = M'$ for

$$M' = CUR,\ C = M_{:,\mathcal{J}},\ U = G^{-1},\ G = M_{\mathcal{I},\mathcal{J}},\ \text{and } R = M_{\mathcal{I},:}. \tag{2.2}$$

We call the matrices G and U the *generator* and *nucleus* of *CUR decomposition* of M, respectively.

[4] An $m\times n$ matrix M is *unitary* (also called *orthogonal* if it is real) if $M^*M = I_n$ or $MM^* = I_m$ for I_s denoting the $s\times s$ identity matrix.

CUR approximation M' of a matrix M of numerical rank r extends CUR decomposition, although the approximation $M = M' + E$ for M' of (2.2) can be poor if CUR generator G is ill-conditioned.[5]

We generalize CUR LRA by allowing to use $k \times l$ CUR generators for k and l satisfying

$$r \le k \le m,\ r \le l \le n \tag{2.3}$$

and to choose any $l \times k$ nucleus U for which the error matrix $E = CUR - M$ has smaller norm.

Given two matrices C and R, the minimal error norm of CUR LRA

$$||E||_F = ||M - CUR||_F \le ||M - CC^+M||_F + ||M - MR^+R||_F$$

is reached for the nucleus $U = C^+MR^+$ (see [MD09, equation (6)]), whose computation has superlinear cost.

Hereafter we study *canonical CUR LRA* (cf. [DMM08], [CLO16], [OZ18]) with a nucleus of CUR LRA given by the r-truncation of a CUR generator:

$$U := G_r^+ = 1/\sigma_\rho(M).$$

In this case the computation of a nucleus involves kl memory cells and $O(kl \min\{k, l\})$ flops.

Unlike 2-factor LRA of (2.1), CUR LRA is a 3-factor LRA, which can generally be represented as follows:

$$M = M' + E,\ |E| \le \xi,\ M' = ATB,\ A \in \mathbb{R}^{m\times k},\ T \in \mathbb{R}^{k\times l},\ B \in \mathbb{R}^{l\times n}, \tag{2.4}$$

and one typically seeks LRA with $k \ll m$ and/or $l \ll n$. The pairs of maps $AT \to A$ and $B \to B$ as well as $A \to A$ and $TB \to B$ turn a 3-factor LRA ATB of (2.4) into a 2-factor LRA AB of (2.1). The r-top SVD and a CUR LRA of M are two important classes of 3-factor LRA.

3 Computation of LRA with Subspace Sampling Directed by Leverage Scores: the State of the Art

In this section we recall statistical approach to the computation of CUR generators by means of *subspace sampling* directed by leverage scores. The CUR LRA algorithms of [DMM08], implementing this approach, outputs CUR LRA of a matrix M such that whp

$$||M - CUR||_F \le (1 + \epsilon)\sigma_{F,r+1} \tag{3.1}$$

for $\sigma_{F,r+1}$ of Theorem 2.1 and any fixed positive ϵ. The algorithm runs at at sub-linear cost even for the worst case input, except for the stage of computing leverage scores.

Let us supply some details. Let $M_r = U^{(r)}\Sigma^{(r)}V^{(r)*}$ be r-top SVD where $U^{(r)} \in \mathbb{C}^{m\times r}$, $\Sigma^{(r)} \in \mathbb{C}^{r\times r}$, and $V^{(r)^*} = (\mathbf{t}_j^{(r)})_{j=1}^n \in \mathbb{C}^{r\times n}$.

Fix scalars $p_1, \dots, p_n$, and β such that

$$0 < \beta \le 1,\ p_j \ge (\beta/r)||\mathbf{v}_j^{(r)}||^2 \text{ for } j = 1, \dots, n, \text{ and } \sum_{j=1}^n p_j = 1. \tag{3.2}$$

[5]The papers [GZT95], [GTZ97], [GZT97], [GT01], [GT11], [GOSTZ10], [OZ16], and [OZ18] define CGR approximations having nuclei G; "G" can stand, say, for "germ". We use the acronym CUR, more customary in the West. "U" can stand, say, for "unification factor", and we notice the alternatives of CNR, CCR, or CSR with N, C, and S standing for *"nucleus", "core", and "seed"*.

Call the scalars $p_1, \dots, p_n$ the *SVD-based leverage scores* for the matrix M (cf. (C.1)). They stay invariant if we pre-multiply the matrix $V^{(r)*}$ by a unitary matrix. Furthermore

$$p_j = ||\mathbf{v}_j^{(r)}||^2/r \text{ for } j = 1, \dots, n \text{ if } \beta = 1. \tag{3.3}$$

For any $m \times n$ matrix M, [HMT11, Algorithm 5.1] computes the matrix $V^{(r)}$ and leverage scores $p_1, \dots, p_n$ by using mn memory cells and $O(mnr)$ flops.[6]

Given an integer parameter l, $1 \le l \le n$, and leverage scores $p_1, \dots, p_n$, Algorithms C.1 and C.2, reproduced from [DMM08], compute auxiliary sampling and rescaling matrices, $S = S_{M,l}$ and $D = D_{M,l}$, respectively. (In particular Algorithms C.1 and C.2 sample and rescale either exactly l columns of an input matrix M or at most its l columns in expectation – the ith column with probability p_i or $\min\{1, lp_i\}$, respectively.) Then [DMM08, Algorithms 1 and 2] compute a CUR LRA of a matrix M as follows.

Algorithm 3.1. [CUR LRA by using SVD-based leverage scores.]

INPUT: *A matrix $M \in C^{m \times n}$ and a target rank r.*

INITIALIZATION: *Choose two integers $k \ge r$ and $l \ge r$ and real β and $\bar{\beta}$ in the range $(0, 1]$.*

COMPUTATIONS:
1. *Compute the leverage scores $p_1, \dots, p_n$ of (3.2).*
2. *Compute sampling and rescaling matrices S and D by applying Algorithm C.1 or C.2. Compute and output a CUR factor $C := WS$.*
3. *Compute leverage scores $\bar{p}_1, \dots, \bar{p}_m$ satisfying relationships (3.2) under the following replacements: $\{p_1, \dots, p_n\} \leftarrow \{\bar{p}_1, \dots, \bar{p}_m\}$, $M \leftarrow (CD)^T$ and $\beta \leftarrow \bar{\beta}$.*
4. *By applying Algorithm C.1 or C.2 to these leverage scores compute $k \times l$ sampling matrix $\bar{S}$ and $k \times k$ rescaling matrix $\bar{D}$.*
5. *Compute and output a CUR factor $R := \bar{S}^T M$.*
6. *Compute and output a CUR factor $U := DW^+\bar{D}$ for $W := \bar{D}\bar{S}^T MSD$.*

Complexity estimates: Overall Algorithm 3.1 involves $kn+ml+kl$ memory cells and $O((m+k)l^2 + kn)$ flops in addition to mn cells and $O(mnr)$ flops used for computing SVD-based leverage scores at stage 1. Except for that stage the algorithm runs at sub-linear cost if $k+l^2 \ll \min\{m, n\}$.

Bound (3.1) is expected to hold for the output of the algorithm if we bound the integers k and l by combining [DMM08, Theorems 4 and 5] as follows.

Theorem 3.1. *Suppose that*

(i) $M \in C^{m \times n}$, $0 < r \le \min\{m, n\}$, $\epsilon, \beta, \bar{\beta} \in (0, 1]$, and $\bar{c}$ is a sufficiently large constant,

(ii) four integers k, k_-, l, and l_- satisfy the bounds

$$0 < l_- = 3200r^2/(\epsilon^2\beta) \le l \le n \text{ and } 0 < k_- = 3200l^2/(\epsilon^2\bar{\beta}) \le k \le m \tag{3.4}$$

or

$$l_- = \bar{c}\, r \log(r)/(\epsilon^2\beta) \le l \le n \text{ and } k_- = \bar{c}\, l \log(l)/(\epsilon^2\bar{\beta}) \le k \le m, \tag{3.5}$$

(iii) we apply Algorithm 3.1 invoking at stages 2 and 4 either Algorithm C.1 under (3.4) or Algorithm C.2 under (3.5).

Then bound (3.1) holds with a probability at least 0.7.

[6] Here and hereafter "flop" stands for "floating point arithmetic operation".

Remark 3.1. *The bounds $k_- \le m$ and $l_- \le n$ imply that either $\epsilon^6 \ge 3200^3 r^4/(m\beta^2\bar{\beta})$ and $\epsilon^2 \ge 3200r/(n\beta)$ if Algorithm C.1 is applied or $\epsilon^4 \ge \bar{c}^2 r\log(r)\log(\bar{c}r\log(r)/(\epsilon^2\beta))/(m\beta^2\bar{\beta})$ and $\epsilon^2 \ge \bar{c}r\log(r)/(n\beta)$ if Algorithm C.2 is applied for a sufficiently large constant $\bar{c}$.*

Remark 3.2. *The estimates k_- and l_- of (3.4) and (3.5) are minimized for $\beta = \bar{\beta} = 1$ and a fixed ϵ. By decreasing the values of β and $\bar{\beta}$ we increase these two estimates by factors of $1/\beta$ and $1/(\beta^2\bar{\beta})$, respectively, and for any values of the leverage scores p_i in the ranges (3.4) and (3.5) we can ensure randomized error bound (3.1).*

The following result implies that the r-top SVD and hence the leverage scores are stable in perturbation of a matrix M within $0.2(\sigma_r(M) - \sigma_{r+1}(M))$.[7]

Theorem 3.2. [See [GL13, Theorem 8.6.5].] *Suppose that*

$$g =: \sigma_r(M) - \sigma_{r+1}(M) > 0 \text{ and } ||E||_F \le 0.2g.$$

Then, for the left and right singular spaces associated with the r largest singular values of the matrices M and $M+E$, there exist unitary matrix bases $B_{r,\text{left}}(M)$, $B_{r,\text{right}}(M)$, $B_{r,\text{left}}(M+E)$, and $B_{r,\text{right}}(M+E)$, respectively, such that

$$\max\{||B_{r,\text{left}}(M+E) - B_{r,\text{left}}(M)||_F, ||B_{r,\text{right}}(M+E) - B_{r,\text{right}}(M)||_F\} \le 4\frac{||E||_F}{g}.$$

For example, if $\sigma_r(M) \gg \sigma_{r+1}(M)$, which implies that $g \approx \sigma_r(M)$, then the upper bound on the right-hand side is approximately $4||E||_F/\sigma_r(M)$.

Leverage scores are expressed through the singular vectors, and in Section 5 we display the results of our tests that show the impact of input perturbations on the leverage scores.

Remark 3.3. *By choosing parameter $\beta < 1$ in (3.2) we can expand the range of perturbations of low rank input, which can be covered by our study of LRA directed by the leverage scores.*

Remark 3.4. *At stage 6 of Algorithm 3.1 we can alternatively apply the simpler expressions $U := (\bar{S}^T MS)^+ = (S^T C)^+ = (RS)^+$, although this would a little weaken numerical stability of the computation of a nucleus of a perturbed input matrix M.*

4 LRA with leverage scores for random inputs

The computation of leverage scores is the bottleneck stage of the algorithms of [DMM08], but in this section we bypass that stage simply by assigning the uniform leverage scores and then prove that the resulting algorithms still compute accurate CUR LRA of a perturbed factor-Gaussian matrix whp.

Theorem 3.2 reduces our task to the case of factor-Gaussian matrix M. The following theorem further reduces it to the case of a Gaussian matrix.

Theorem 4.1. *Let $M = GH$ for $G \in \mathbb{C}^{m\times r}$ and $H \in \mathbb{C}^{r\times n}$ and let $r = \text{rank}(G) = \text{rank}(H)$. Then the matrices M^T and M share their SVD-based leverage scores with the matrices G^T and H, respectively,*

[7] It is more explicit than the similar results by Davis-Kahan 1970 and Wedin 1972, which involve angles between singular spaces.

Proof. Let $G = S_G \Sigma_G T_G^* \in \mathbb{C}^{m \times r}$ and $H = S_H \Sigma_H T_H^*$ be SVDs.

Write $W := \Sigma_G T_G^* S_H \Sigma_H$ and let $W = S_W \Sigma_W T_W^*$ be SVD.

Notice that Σ_G, T_G^*, S_H, and Σ_H are $r \times r$ matrices.

Consequently so are the matrices W, S_W, Σ_W, and T_W^*.

Hence $M = \bar{S}_G \Sigma_W \bar{T}_H^*$ where $\bar{S}_G = S_G S_W$ and $\bar{T}_H^* = T_W^* T_H^*$ are unitary matrices.

Therefore $M = \bar{S}_G \Sigma_W \bar{T}_H^*$ is SVD.

Consequently the columns of the unitary matrices $\bar{S}_G$ and $\bar{T}_H^{*T}$ span the r top right singular spaces of the matrices M^T and M, respectively, and so do the columns of the matrices S_G and T_H^{*T} as well because $\bar{S}_G = S_G S_W$ and $\bar{T}_H^* = T_W^* T_H^*$ where S_W and T_W^* are $r \times r$ unitary matrices. This proves the theorem. □

If $M = GH$ (resp. $M^T = H^T G^T$) is a right or diagonally scaled factor-Gaussian matrix, then with probability 1 the matrices M and H (resp. M^T and G^T) share their leverage scores by virtue of Theorem 4.1. If we only know that the matrix M is either a left or a right factor-Gaussian matrix, apply Algorithm 3.1 to both matrices M and M^T and in at least one case reduce the computation of the leverage scores to the case of Gaussian matrix.

Now let $r \ll n$ and outline our further steps of the estimation of the leverage scores.

Outline 4.1. *Recall from [E89, Theorem 7.3] or [RV09] that $\kappa(G) \to 1$ as $r/n \to 0$ for $G \in \mathcal{G}^{r \times n}$. It follows that for $r \ll n$ the matrix G is close to a scaled unitary matrix whp, and hence within a factor $\frac{1}{\sqrt{n}}$ it is close to the unitary matrix T_G^* of its right singular space whp. Therefore the leverage scores p_j of a Gaussian matrix $G = (\mathbf{g}_j)_{j=1}^n$ are close to the values $\frac{1}{rn}||\mathbf{g}_j||^2$, $j = 1, \dots, n$. They, however, are invariant in j and close to $1/n$ for all j whp. This choice trivializes the approximation of the leverage scores of a Gaussian matrix and hence of a factor-Gaussian matrix. Since this bottleneck stage of Algorithm 3.1 has been made trivial, the entire algorithm now runs at sub-linear cost while it still outputs accurate CUR LRA whp in the case of a factor-Gaussian input. Theorem 3.2 implies extension to a perturbed factor-Gaussian input.*

Next we elaborate upon this outline.

Lemma 4.1. *Suppose that $G \in \mathcal{G}^{n \times r}$, $\mathbf{u} \in \mathbb{R}^r$, $\mathbf{v} = \frac{1}{\sqrt{n}} G\mathbf{u}$, and $r \le n$. Fix $\bar{\epsilon} > 0$. Then*

$$\text{Probability}\{(1-\bar{\epsilon})||\mathbf{u}||^2 \le ||\mathbf{v}||^2 \le (1+\bar{\epsilon})||\mathbf{u}||^2\} \ge 1 - 2e^{-(\bar{\epsilon}^2 - \bar{\epsilon}^3)\frac{n}{4}}.$$

Proof. See [AV06, Lemma 2]. □

Lemma 4.2. *Fix the spectral or Frobenius norm $|\cdot|$ and let $M = S_M \Sigma_M T_M^*$ be SVD. Then $S_M T_M^*$ is a unitary matrix and*

$$|M - S_M T_M^*|^2 \le |MM^* - I|.$$

Proof. $S_M T_M^*$ is a unitary matrix because both matrices S_M and T_M^* are unitary and at least one of them is a square matrix.

Next observe that $M - S_M T_M^* = S_M \Sigma_M T_M^* - S_M T_M^* = S_M(\Sigma_M - I)T_M^*$, and so

$$|M - S_M T_M^*| = |\Sigma_M - I|.$$

Likewise $MM^* - I = S_M \Sigma_M^2 S_M^* - I = S_M(\Sigma_M^2 - I)S_M^*$, and so

$$|MM^* - I| = |\Sigma_M^2 - I|.$$

Complement these equations for the norms with the inequality

$$|\Sigma_M^2 - I| = |\Sigma_M - I|\ |\Sigma_M + I| \ge |\Sigma_M - I|,$$

which holds because Σ_M is a diagonal matrix having only nonnegative entries. □

Lemma 4.3. *Suppose that $0 < \epsilon < 1$ and that n and $r < n$ are two integers such that $n = O(r^4\epsilon^{-2}\log(n))$ is sufficiently large. Furthermore let $G = (\mathbf{g}_j)_{j=1}^n \in \mathcal{G}^{r\times n}$. Then whp*

$$\Big|\Big|\frac{1}{n}GG^* - I_r\Big|\Big|_F^2 < \epsilon.$$

Proof. Let $\mathbf{e}_j$ denote the jth column of the identity matrix I_r. Apply Lemma 4.1 for $\mathbf{u}$ equal to the vectors $\mathbf{e}_j$ and $\mathbf{e}_i - \mathbf{e}_j$ and to $\mathbf{v} = \frac{1}{\sqrt{n}}\mathbf{g}_j$ and for $i, j = 1, \ldots, r$ where $i \neq j$, substitute $||\mathbf{e}_j|| = 1$ and $||\mathbf{e}_i - \mathbf{e}_j||^2 = 2$ for all j and all $i \neq j$, and deduce that with probability no less than $1 - 2n^2 e^{-(\bar{\epsilon}^2 - \bar{\epsilon}^3)\frac{n}{4}}$

$$1 - \bar{\epsilon} < ||\mathbf{g}_j||^2/n < 1 + \bar{\epsilon} \text{ and } 2 - \bar{\epsilon} < ||\mathbf{g}_i - \mathbf{g}_j||^2/n < 2 + \bar{\epsilon} \tag{4.1}$$

for all j and for all $i \neq j$.

Now, write $\epsilon = \frac{2}{3r^2}\bar{\epsilon}$, and let $n = O(r^4\epsilon^{-2}\log(n))$ such that (4.1) holds. Since the (i, j)th entry of the matrix GG^* is given by $\mathbf{g}_i^*\mathbf{g}_j$, deduce that

$$\Big|\Big|\frac{1}{n}GG^* - I_r\Big|\Big|_F^2 \leq \Big(\frac{3}{2}r^2 - \frac{r}{2}\Big)\bar{\epsilon} < \frac{3}{2}r^2\bar{\epsilon} = \epsilon.$$

□

Combine Lemmas 4.2 and 4.3 for $M = \frac{1}{\sqrt{n}}G$ and obtain the following result.

Corollary 4.1. *Under the assumptions of Lemma 4.3 let $\frac{1}{\sqrt{n}}G = S\Sigma T^*$ be SVD. Then whp*

$$\Big|\Big|\frac{1}{\sqrt{n}}G - ST^*\Big|\Big|_F^2 < \ \epsilon.$$

Remark 4.1. *Under the assumptions of the corollary $\Sigma \to I_r$ as $\epsilon \to 0$, and then the norm $||(\Sigma + I_r)^{-1}||_F$ and consequently the ratio $\frac{||\Sigma - I_r||_F}{||\Sigma^2 - I_r||_F}$ converge to 1/2.*

Theorem 4.2. *Given two integers n and r and a positive ϵ satisfying the assumptions of Lemma 4.3, a Gaussian matrix $G = (\mathbf{g}_j)_{j=1}^n \in \mathcal{G}^{r\times n}$, and SVD $\frac{1}{\sqrt{n}}G = S\Sigma T^*$, write $T^* = (\mathbf{t}_j)_{j=1}^n$ and $\beta = 1$ and define the SVD-based leverage scores of the matrix $\frac{1}{\sqrt{n}}G$, that is, $p_j = ||\mathbf{t}_j||^2/r$ for $j = 1, \ldots, n$ (cf. (3.2) and (3.3)). Then whp*

$$\Big|p_j - \frac{||\mathbf{g}_j||^2}{nr}\Big| \leq \frac{\epsilon}{r} \text{ for } j = 1, ..., n.$$

Proof. Notice that $||S\mathbf{t}_j|| = ||\mathbf{t}_j||$ for all j since S is a square unitary matrix; deduce from Corollary 4.1 that whp

$$\frac{1}{n}||\mathbf{g}_j||^2 - ||S\mathbf{t}_j||^2 < \epsilon \text{ for } i = 1, ..., n.$$

□

Remark 4.2. *The estimate of the theorem is readily extended to the case where the leverage scores are defined by (3.2) rather than (3.3).*

Now observe that the squared norms $||\mathbf{g}_j||^2$ are iid chi-square random variables $\chi^2(r)$ and therefore are quite strongly concentrated in a reasonable range about the expected value of such a variable. Hence we obtain reasonably good approximations to SVD-based leverage scores for a Gaussian matrix $G = (\mathbf{g}_j)_{j=1}^n \in \mathcal{G}^{r\times n}$ by choosing $p_j = 1/n$ for all j, and then we satisfy bounds (3.2) and consequently (3.1) by choosing a reasonably small positive value β.

Let us supply some further details.

Lemma 4.4. *Let $Z = \sum_{i=1}^{r} X_i^2$ for iid Gaussian variables $X_1, \dots, X_r$. Then*

$$\text{Probability}\{Z - r \geq 2\sqrt{rx} + 2rx\} \leq \exp(-x) \text{ for any } x > 0.$$

Proof. See [LM00, Lemma 1]. □

Corollary 4.2. *Given two integers n and r and a positive ϵ satisfying the assumptions of Theorem 4.2, a Gaussian matrix $G = (\mathbf{g}_j)_{j=1}^{n} \in \mathcal{G}^{r \times n}$, and denote its SVD-based leverage scores as p_j for $j = 1, ..., n$. Fix $0 < \beta < r(n\epsilon + r)^{-1}$ such that $\left(\frac{1}{\beta} - \frac{n\epsilon + r}{r}\right) = \Theta(\ln n)$, then whp*

$$\frac{1}{n} > \beta p_j \text{ for } j = 1, ..., n.$$

Proof. Deduce from Lemma 4.4 that

$$\begin{aligned}\text{Probability}\Big\{\frac{1}{n} > \beta\Big(\frac{||\mathbf{g}_j||^2}{nr} + \frac{\epsilon}{r}\Big)\Big\} &= 1 - \text{Probability}\Big\{||\mathbf{g}_j||^2 - r \geq \Big(\frac{1}{\beta} - \frac{n\epsilon + r}{r}\Big)r\Big\} \\ &\geq 1 - \exp\left(-f(\beta)\right)\end{aligned}$$

for $f(\beta)$ being the positive solution of $2\sqrt{rx} + 2rx = \left(\frac{1}{\beta} - \frac{n\epsilon + r}{r}\right)r$ and any $j \in \{1, 2, \dots, n\}$.

Furthermore the random variables $||\mathbf{g}_j||^2$ are independent, and therefore

$$\begin{aligned}\text{Probability}\Big\{\frac{1}{n} > \beta\Big(\frac{||\mathbf{g}_j||^2}{nr} + \frac{\epsilon}{r}\Big) \text{ for all } j = 1, 2, \dots, n\Big\} &\geq \big(1 - \exp(-f(\beta))\big)^n \\ &\geq 1 - \exp\Big(-f(\beta) + \ln n\Big).\end{aligned}$$

It can be readily verified that $f(\beta)$ is dominated by $\frac{1}{\beta} - \frac{n\epsilon + r}{r}$. Combine this result with Theorem 4.2 and conclude that whp

$$\frac{1}{n} > \beta p_j \text{ for } j = 1, ..., n.$$

□

We have completed our formal support for Outline 4.1 and arrived at the following result where one can specify the output errors by using Theorem 3.2 and Corollary 4.2.

Theorem 4.3. *Suppose that the algorithms of [DMM08] have been applied to the computation of CUR LRA of a perturbed factor-Gaussian matrix by using the uniform leverage scores. Then this computation is performed at sub-linear cost and whp outputs reasonably close CUR LRA.*

5 Testing perturbation of leverage scores

Table 5.1 shows the means and standard deviations of the norms of the relative errors of approximation of the input matrix M and of its LRA AB and similar data for the maximum difference between the SVD-based leverage scores of the pairs of these matrices. We computed a close approximation to the leverage scores of input matrices M at sub-linear cost by using their LRA AB. The table also displays numerical ranks of the input matrices M defined up to tolerance 10^{-6}. Our statistics were gathered from 100 runs for each input matrix.

Input matrices. The dense matrices with smaller ratios of "numerical rank/n" from the built-in test problems in Regularization Tools, which came from discretization (based on Galerkin or

quadrature methods) of the Fredholm Integral Equations of the first kind,[8] namely to the following six input classes from the Database:

baart: Fredholm Integral Equation of the first kind,
shaw: one-dimensional image restoration model,
gravity: 1-D gravity surveying model problem,
wing: problem with a discontinuous solution,
foxgood: severely ill-posed problem,
inverse Laplace: inverse Laplace transformation.

We computed the LRA approximations AB by using [PZ16, Algorithm 1.1] with multipliers of Class 5 of [PZ16, Section 5.3].

Our goal was to compare the approximate leverage scores with their true values. The columns "mean(Leverage Score Error)" and "std(Leverage Score Error)" of the table show that these approximations are in good accordance with increasing r.

In addition, the last three lines of Table 5.1 show similar results for perturbed two-sided factor-Gaussian matrices GH of rank r approximating an input matrix M up to perturbations.

			LRA Rel Error		Leverage Score Error	
Input Matrix	r	rank	mean	std	mean	std
baart	4	6	6.57e-04	1.17e-03	1.57e-05	5.81e-05
baart	6	6	7.25e-07	9.32e-07	5.10e-06	3.32e-05
baart	8	6	7.74e-10	2.05e-09	1.15e-06	3.70e-06
foxgood	8	10	5.48e-05	5.70e-05	7.89e-03	7.04e-03
foxgood	10	10	9.09e-06	8.45e-06	1.06e-02	6.71e-03
foxgood	12	10	1.85e-06	1.68e-06	5.60e-03	3.42e-03
gravity	23	25	3.27e-06	1.82e-06	4.02e-04	3.30e-04
gravity	25	25	8.69e-07	7.03e-07	4.49e-04	3.24e-04
gravity	27	25	2.59e-07	2.88e-07	4.64e-04	3.61e-04
laplace	23	25	2.45e-05	9.40e-05	4.85e-04	3.03e-04
laplace	25	25	3.73e-06	1.30e-05	4.47e-04	2.78e-04
laplace	27	25	1.30e-06	4.67e-06	3.57e-04	2.24e-04
shaw	10	12	6.40e-05	1.16e-04	2.80e-04	5.17e-04
shaw	12	12	1.61e-06	1.60e-06	2.10e-04	2.70e-04
shaw	14	12	4.11e-08	1.00e-07	9.24e-05	2.01e-04
wing	2	4	1.99e-02	3.25e-02	5.17e-05	2.07e-04
wing	4	4	7.75e-06	1.59e-05	7.17e-06	2.30e-05
wing	6	4	2.57e-09	1.15e-08	9.84e-06	5.52e-05
factor-Gaussian	25	25	1.61e-05	3.19e-05	4.05e-08	8.34e-08
factor-Gaussian	50	50	2.29e-05	7.56e-05	2.88e-08	6.82e-08
factor-Gaussian	75	75	4.55e-05	1.90e-04	1.97e-08	2.67e-08

Table 5.1: Tests for the perturbation of leverage scores

[8] See http://www.math.sjsu.edu/singular/matrices and http://www2.imm.dtu.dk/∼pch/Regutools
For more details see Chapter 4 of the Regularization Tools Manual at
http://www.imm.dtu.dk/∼pcha/Regutools/RTv4manual.pdf

Appendix

A Small families of hard inputs for LRA at sub-linear cost

Any LRA algorithm that runs at sub-linear cost fails on the following small families of LRA inputs.

Example A.1. *Define the following family of $m \times n$ matrices of rank 1 (we call them δ-*matrices*): $\{\Delta_{i,j},\ i = 1, \dots, m;\ j = 1, \dots, n\}$. Also include the $m \times n$ null matrix $O_{m,n}$ into this family. Now fix an LRA algorithm that runs at sub-linear cost; it does not access the (i, j)th entry of its input matrices for some pair of i and j. Therefore it outputs the same approximation of the matrices $\Delta_{i,j}$ and $O_{m,n}$, with an undetected error at least 1/2. Apply the same argument to the set of $mn + 1$ small-norm perturbations of the matrices of the above family and to the $mn + 1$ sums of the latter matrices with any fixed $m \times n$ matrix of low rank. Finally, the same argument shows that a posteriori output errors of an LRA algorithm applied to the same input families cannot be estimated at sub-linear cost.*

The example actually covers randomized LRA algorithms as well. Indeed suppose that an LRA algorithm does not access a constant fraction of the entries of an input matrix. Then with a constant probability the algorithm misses an entry whose value greatly exceeds those of all other entries, in which case the algorithm can hardly approximate that entry closely. The paper [Pa] shows, however, that close LRA can be computed at sub-linear cost in two successive C-A iterations provided that we avoid choosing degenerating initial submatrix, which is precisely the problem with the matrix families of Example A.1. The sub-linear cost algorithms of [MW17] and [BW18] compute LRA of matrices of two important special matrix classes.

B Background on random matrix computations

B.1 Gaussian and factor-Gaussian matrices of low rank and low numerical rank

$\mathcal{G}^{p\times q}$ denotes the linear space of $p \times q$ matrices filled with iid Gaussian (normal) random variables, which we call *Gaussian* for short.

Theorem B.1. [Nondegeneration of a Gaussian Matrix.] *Let $F \in \mathcal{G}^{r\times m}$, $H \in \mathcal{G}^{n\times r}$, $M \in \mathbb{R}^{m\times n}$ and $r \le \text{rank}(M)$. Then the matrices F, H, FM, and MH have full rank r with probability 1.*

Proof. Fix any of the matrices F, H, FM, and MH and its $r \times r$ submatrix B. Then the equation $\det(B) = 0$ defines an algebraic variety of a lower dimension in the linear space of the entries of the matrix because in this case $\det(B)$ is a polynomial of degree r in the entries of the matrix F or H (cf. [BV88, Proposition 1]). Clearly, such a variety has Lebesgue and Gaussian measures 0, both being absolutely continuous with respect to one another. This implies the theorem. □

Assumption B.1. [Nondegeneration of a Gaussian matrix.] *Hereafter we simplify the statements of our results by assuming that a Gaussian matrix has full rank, ignoring the probability 0 of its degeneration.*

Definition B.1. [Factor-Gaussian matrices.] *Let $\rho \le \min\{m, n\}$ and let $\mathcal{G}^{m\times n}_{\rho,B}$, $\mathcal{G}^{m\times n}_{A,\rho}$, and $\mathcal{G}^{m\times n}_{\rho,C}$ denote the classes of matrices $G_{m,\rho}B$, $AG_{\rho,n}$, and $G_{m,\rho}\Sigma G_{\rho,n}$, respectively, which we call* left, right, *and* two-sided factor-Gaussian matrices of rank ρ, *respectively, provided that $G_{p,q}$ denotes a $p \times q$ Gaussian matrix, $A \in \mathbb{R}^{m\times\rho}$, $B \in \mathbb{R}^{\rho\times n}$, $\Sigma \in \mathbb{R}^{\rho\times\rho}$, and A, B, and Σ are well-conditioned matrices of full rank ρ, and $\Sigma = (\sigma_j)_{j=1}^{\rho}$ such that $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_\rho > 0$.*

Theorem B.2. *The class $\mathcal{G}_{r,C}^{m\times n}$ of two-sided $m\times n$ factor-Gaussian matrices $G_{m,\rho}\Sigma G_{\rho,n}$ does not change in the transition to $G_{m,r}CG_{r,n}$ for a well-conditioned nonsingular $\rho\times\rho$ matrix C.*

Proof. Let $C=U_C\Sigma_C V_C^*$ be SVD. Then $A=G_{m,r}U_C\in\mathcal{G}^{m\times r}$ and $B=V_C^*G_{r,n}\in\mathcal{G}^{r\times n}$ by virtue of orthogonality invariance of Gaussian matrices, and so $G_{m,r}CG_{r,n}=A\Sigma_C B$ for $A\in\mathcal{G}^{m\times r}$ and $B\in\mathcal{G}^{r\times n}$. □

Definition B.2. The relative norm of a perturbation of a Gaussian matrix *is the ratio of the perturbation norm and the expected value of the norm of the matrix (estimated in Theorem B.4).*

We refer to all three matrix classes above as *factor-Gaussian matrices of rank r*, to their perturbations within a relative norm bound ϵ as *factor-Gaussian matrices of ϵ-rank r*, and to their perturbations within a small relative norm as *factor-Gaussian matrices of numerical rank r*, to which we also refer as *perturbations of factor-Gaussian matrices.*

Clearly $||(A\Sigma)^+||\le||\Sigma^{-1}||\,||A^+||$ and $||(\Sigma B)^+||\le||\Sigma^{-1}||\,||B^+||$ for a two-sided factor-Gaussian matrix $M=A\Sigma B$ of rank r of Definition B.1, and so whp such a matrix is both left and right factor-Gaussian of rank r.

We readily verify the following result.

Theorem B.3. *(i) A submatrix of a two-sided (resp. scaled) factor-Gaussian matrix of rank ρ is a two-sided (resp. scaled) factor-Gaussian matrix of rank ρ, (ii) a $k\times n$ (resp. $m\times l$) submatrix of an $m\times n$ left (resp. right) factor-Gaussian matrix of rank ρ is a left (resp. right) factor-Gaussian matrix of rank ρ.*

B.2 Norms of a Gaussian matrix and its pseudo inverse

Hereafter $\Gamma(x)=\int_0^\infty\exp(-t)t^{x-1}dt$ denotes the Gamma function, $\mathbb{E}(v)$ denotes the *expected value* of a random variable v, and we write

$$\mathbb{E}||M||:=\mathbb{E}(||M||),\ \mathbb{E}||M||_F^2:=\mathbb{E}(||M||_F^2),\ \text{and } e:=2.71828\dots. \tag{B.1}$$

Definition B.3. [Norms of a Gaussian matrix and its pseudo inverse.] *Write $\nu_{m,n}=|G|$, $\nu_{\mathrm{sp},m,n}=||G||$, $\nu_{F,m,n}=||G||_F$, $\nu^+_{m,n}=|G^+|$, $\nu^+_{\mathrm{sp},m,n}=||G^+||$, and $\nu^+_{F,m,n}=||G^+||_F$, for a Gaussian $m\times n$ matrix G. ($\nu_{m,n}=\nu_{n,m}$ and $\nu^+_{m,n}=\nu^+_{n,m}$, for all pairs of m and n.)*

Theorem B.4. [Norms of a Gaussian matrix.]

(i) [DS01, Theorem II.7]. Probability$\{\nu_{\mathrm{sp},m,n}>t+\sqrt{m}+\sqrt{n}\}\le\exp(-t^2/2)$ *for $t\ge0$,* $\mathbb{E}(\nu_{\mathrm{sp},m,n})\le\sqrt{m}+\sqrt{n}$.

(ii) $\nu_{F,m,n}$ is the χ-function, with the expected value $\mathbb{E}(\nu_{F,m,n})=mn$ and the probability density

$$\frac{2x^{n-i}exp(-x^2/2)}{2^{n/2}\Gamma(n/2)},$$

Theorem B.5. [Norms of the pseudo inverse of a Gaussian matrix.]

(i) Probability $\{\nu^+_{\mathrm{sp},m,n}\ge m/x^2\}<\frac{x^{m-n+1}}{\Gamma(m-n+2)}$ *for $m\ge n\ge2$ and all positive x,*

(ii) Probability $\{\nu^+_{F,m,n}\ge t\sqrt{\frac{3n}{m-n+1}}\}\le t^{n-m}$ *and* Probability $\{\nu^+_{\mathrm{sp},m,n}\ge t\frac{e\sqrt{m}}{m-n+1}\}\le t^{n-m}$ *for all $t\ge1$ provided that $m\ge4$,*

(iii) $\mathbb{E}((\nu^+_{F,m,n})^2)=\frac{n}{m-n-1}$ *and* $\mathbb{E}(\nu^+_{\mathrm{sp},m,n})\le\frac{e\sqrt{m}}{m-n}$ *provided that $m\ge n+2\ge4$,*

(iv) Probability $\{\nu^+_{\mathrm{sp},n,n}\ge x\}\le\frac{2.35\sqrt{n}}{x}$ *for $n\ge2$ and all positive x, and furthermore $||M_{n,n}+G_{n,n}||^+\le\nu_{n,n}$ for any $n\times n$ matrix $M_{n,n}$ and an $n\times n$ Gaussian matrix $G_{n,n}$.*

Proof. See [CD05, Proof of Lemma 4.1] for claim (i), [HMT11, Proposition 10.4 and equations (10.3) and (10.4)] for claims (ii) and (iii), and [SST06, Theorem 3.3] for claim (iv). □

Theorem B.5 implies reasonable probabilistic upper bounds on the norm $\nu^+_{m,n}$ even where the integer $|m-n|$ is close to 0; whp the upper bounds of Theorem B.5 on the norm $\nu^+_{m,n}$ decrease very fast as the difference $|m-n|$ grows from 1.

C Computation of Sampling and Re-scaling Matrices

We begin with the following simple computations. Given an n vectors $\mathbf{v}_1, \dots, \mathbf{v}_n$ of dimension l, write $V = (\mathbf{v}_i)_{i=1}^n$ and compute n leverage scores

$$p_i = \mathbf{v}_i^T \mathbf{v}_i / ||V||_F^2, i = 1, \dots, n. \tag{C.1}$$

Notice that $p_i \geq 0$ for all i and $\sum_{i=1}^n p_i = 1$.

Next assume that some leverage scores $p_1, \dots, p_n$ are given to us and next recall [DMM08, Algorithms 4 and 5]. For a fixed positive integer l they sample either exactly l columns of an input matrix W (the ith column with probability p_i) or at most l its columns in expectation (the ith column with probability $\min\{1, lp_i\}$), respectively.

Algorithm C.1. (The Exactly(l) Sampling and Re-scaling. [DMM08, Algorithm 4]).

INPUT: *Two integers l and n such that $1 \leq l \leq n$ and n nonnegative scalars $p_1, \dots, p_n$ such that $\sum_{i=1}^n p_i = 1$.*

INITIALIZATION: *Write $S := O_{n,l}$ and $D := O_{l,l}$.*

COMPUTATIONS: *(1) For $t = 1, \dots, l$ do*

Pick $i_t \in \{1, \dots, n\}$ such that Probability$(i_t = i) = p_i$;

$s_{i_t,t} := 1$;

$d_{t,t} = 1/\sqrt{lp_{i_t}}$;

end

(2) Write $s_{i,t} = 0$ for all pairs of i and t unless $i = i_t$.

OUTPUT: *$n \times l$ sampling matrix $S = (s_i, t)_{i,t=1}^{n,l}$ and $l \times l$ re-scaling matrix $D = \mathrm{diag}(d_{t,t})_{t=1}^l$.*

The algorithm performs l searches in the set $\{1, \dots, n\}$, l multiplications, l divisions, and the computation of l square roots.

Algorithm C.2. (The Expected(l) Sampling and Re-scaling. [DMM08, Algorithm 5]).

INPUT, OUTPUT AND INITIALIZATION *are as in Algorithm C.1.*

COMPUTATIONS: *Write $t := 1$;*

for $t = 1, \dots, l-1$ do

for $j = 1, \dots, n$ do

Pick j with probability $\min\{1, lp_j\}$;

if j is picked, then

$s_{j,t} := 1$;

$d_{t,t} := 1/\min\{1, \sqrt{lp_j}\}$;

$t := t + 1$;

end

end

Algorithm C.2 involves nl memory cells. $O((l+1)n)$ flops, and the computation of l square roots.

Acknowledgements: Our work has been supported by NSF Grants CCF–1116736, CCF–1563942 and CCF–1733834 and PSC CUNY Award 69813 00 48. We are also grateful to E. E. Tyrtyshnikov for the challenge of formally supporting empirical power of C–A iterations, to N. L. Zamarashkin for his comments on his work with A. Osinsky on LRA via volume maximization and on the first draft of [PLSZ17], and to S. A. Goreinov, I. V. Oseledets, A. Osinsky, E. E. Tyrtyshnikov, and N. L. Zamarashkin for reprints and pointers to relevant bibliography.

References

[AV06] R.I. Arriaga, S. Vempala, An Algorithmic Theory of Learning: Robust Concepts and Random Projection, *Machine Learning*, **63, 2**, 161–182, May 2006.

[BV88] W. Bruns, U. Vetter, *Determinantal Rings, Lecture Notes in Math.,* **1327**, Springer, Heidelberg, 1988.

[BW17] C. Boutsidis, D. Woodruff, Optimal CUR Matrix Decompositions, *SIAM Journal on Computing*, **46, 2**, 543–589, 2017, DOI:10.1137/140977898.

[BW18] A. Bakshi, D. P. Woodruff: Sublinear Time Low-Rank Approximation of Distance Matrices, *Procs. 32nd Intern. Conf. Neural Information Processing Systems (NIPS'18),* 3786–3796, Montral, Canada, 2018.

[CD05] Z. Chen, J. J. Dongarra, Condition Numbers of Gaussian Random Matrices, *SIAM. J. on Matrix Analysis and Applications*, **27**, 603–620, 2005.

[CLO16] C. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao and D. P. Mandic, "Tensor Networks for Dimensionality Reduction and Large-scale Optimization: Part 1 Low-Rank Tensor Decompositions", Foundations and Trends in Machine Learning: **9, 4-5**, 249–429, 2016. http://dx.doi.org/10.1561/2200000059

[D88] J. Demmel, The Probability That a Numerical Analysis Problem Is Difficult, *Math. of Computation*, **50**, 449–480, 1988.

[DMM08] P. Drineas, M.W. Mahoney, S. Muthukrishnan, Relative-error CUR Matrix Decompositions, *SIAM Journal on Matrix Analysis and Applications*, **30, 2**, 844–881, 2008.

[DS01] K. R. Davidson, S. J. Szarek, Local Operator Theory, Random Matrices, and Banach Spaces, in *Handbook on the Geometry of Banach Spaces* (W. B. Johnson and J. Lindenstrauss editors), pages 317–368, North Holland, Amsterdam, 2001.

[E88] A. Edelman, Eigenvalues and Condition Numbers of Random Matrices, *SIAM J. on Matrix Analysis and Applications*, **9**, **4**, 543–560, 1988.

[E89] A. Edelman, Eigenvalues and Condition Numbers of Random Matrices, Ph.D. thesis, Massachusetts Institute of Technology, 1989.

[ES05] A. Edelman, B. D. Sutton, Tails of Condition Number Distributions, *SIAM J. on Matrix Analysis and Applications*, **27**, **2**, 547–560, 2005.

[GL13] G. H. Golub, C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, Maryland, 2013 (fourth edition).

[GOSTZ10] S. Goreinov, I. Oseledets, D. Savostyanov, E. Tyrtyshnikov, N. Zamarashkin, How to Find a Good Submatrix, in *Matrix Methods: Theory, Algorithms, Applications* (dedicated to the Memory of Gene Golub, edited by V. Olshevsky and E. Tyrtyshnikov), pages 247–256, World Scientific Publishing, New Jersey, ISBN-13 978-981-283-601-4, ISBN-10-981-283-601-2, 2010.

[GT01] S. A. Goreinov, E. E. Tyrtyshnikov, The Maximal-Volume Concept in Approximation by Low Rank Matrices, *Contemporary Mathematics*, **208**, 47–51, 2001.

[GT11] S. A. Goreinov, E. E. Tyrtyshnikov, Quasioptimality of Skeleton Approximation of a Matrix on the Chebyshev Norm, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **83, 3**, 1–2, 2011.

[GTZ97] S. A. Goreinov, E. E. Tyrtyshnikov, N. L. Zamarashkin, A Theory of Pseudo-skeleton Approximations, *Linear Algebra and Its Applications*, **261**, 1–21, 1997.

[GZT95] S. A. Goreinov, N. L. Zamarashkin, E. E. Tyrtyshnikov, Pseudo-skeleton approximations, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **343, 2**, 151–152, 1995.

[GZT97] S. A. Goreinov, N. L. Zamarashkin, E. E. Tyrtyshnikov, Pseudo-skeleton Approximations by Matrices of Maximal Volume, *Mathematical Notes*, **62, 4**, 515–519, 1997.

[HMT11] N. Halko, P. G. Martinsson, J. A. Tropp, Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions, *SIAM Review*, **53, 2**, 217–288, 2011.

[KS16] N. Kishore Kumar, J. Schneider, Literature Survey on Low Rank Approximation of Matrices, arXiv:1606.06511v1 [math.NA] 21 June 2016.

[LM00] B. Laurent, P. Massart, Adaptive Estimation of a Quadratic Functional by Model Selection, *The Annals of Statistics* **28, 5**, 1302–1338, 2000.
Also see http://www.jstor.org/stable/2674095

[M11] M. W. Mahoney, Randomized Algorithms for Matrices and Data, *Foundations and Trends in Machine Learning*, NOW Publishers, **3, 2**, 2011. Preprint: arXiv:1104.5557 (2011) (Abridged version in: *Advances in Machine Learning and Data Mining for Astronomy*, edited by M. J. Way et al., pp. 647–672, 2012.)

[MD09] M. W. Mahoney, and P. Drineas, CUR matrix decompositions for improved data analysis, *Proceedings of the National Academy of Sciences*, **106 3**, 697–702, 2009.

[MW17] Cameron Musco, D. P. Woodruff: Sublinear Time Low-Rank Approximation of Positive Semidefinite Matrices, IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), 672–683, 2017.

[OZ16] A.I. Osinsky, N. L. Zamarashkin, New Accuracy Estimates for Pseudo-skeleton Approximations of Matrices, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **94, 3**, 643–645, 2016.

[OZ18] A.I. Osinsky, N. L. Zamarashkin, Pseudo-skeleton Approximations with Better Accuracy Estimates, *Linear Algebra and Its Applications*, **537**, 221–249, 2018.

[Pa] V. Y. Pan, Low Rank Approximation of a Matrix at Sub-linear Cost, preprint, 2019.

[PLa] V. Y. Pan, Q. Luan, Refinement of Low Rank Approximation of a Matrix at Sublinear Cost, arXiv:1906.04223 (Submitted on 10 Jun 2019).

[PLSZ16] V. Y. Pan, Q. Luan, J. Svadlenka, L.Zhao, Primitive and Cynical Low Rank Approximation, Preprocessing and Extensions, arXiv 1611.01391 (Submitted on 3 November, 2016).

[PLSZ17] V. Y. Pan, Q. Luan, J. Svadlenka, L. Zhao, Superfast Accurate Low Rank Approximation, preprint, arXiv:1710.07946 (Submitted on 22 October, 2017).

[PLSZa] V. Y. Pan, Q. Luan, J. Svadlenka, L. Zhao, Low Rank Approximation by Means of Subspace Sampling at Sub-linear Cost, arXiv:1906.04327 (Submitted on 10 Jun 2019).

[PLSZb] V. Y. Pan, Q. Luan, J. Svadlenka, L. Zhao, CUR Low Rank Approximation at Sublinear Cost, arXiv:1906.04112 (Submitted on 10 Jun 2019).

[PQY15] V. Y. Pan, G. Qian, X. Yan, Random Multipliers Numerically Stabilize Gaussian and Block Gaussian Elimination: Proofs and an Extension to Low-rank Approximation, *Linear Algebra and Its Applications*, **481**, 202–234, 2015.

[PZ16] V. Y. Pan, L. Zhao, Low-rank Approximation of a Matrix: Novel Insights, New Progress, and Extensions, Proc. of the *Eleventh International Computer Science Symposium in Russia (CSR'2016)*, (Alexander Kulikov and Gerhard Woeginger, editors), St. Petersburg, Russia, June 2016, *Lecture Notes in Computer Science (LNCS)*, **9691**, 352–366, Springer International Publishing, Switzerland (2016).

[PZ17a] V. Y. Pan, L. Zhao, New Studies of Randomized Augmentation and Additive Preprocessing, *Linear Algebra and Its Applications*, **527**, 256–305, 2017.
http://dx.doi.org/10.1016/j.laa.2016.09.035. Also arxiv 1412.5864.

[PZ17b] V. Y. Pan, L. Zhao, Numerically Safe Gaussian Elimination with No Pivoting, *Linear Algebra and Its Applications*, **527**, 349–383, 2017.
http://dx.doi.org/10.1016/j.laa.2017.04.007. Also arxiv 1501.05385

[RV07] M. Rudelson, R. Vershynin, Sampling from Large Matrices: an Approach through Geometric Functional Analysis, *Journal of the ACM*, **54, 4**, Article number 1255449, 2007.

[RV09] M. Rudelson, R. Vershynin, Smallest Singular Value of a Random Rectangular Matrix, *Comm. Pure Appl. Math.*, **62, 12**, 1707–1739, 2009.
https://doi.org/10.1002/cpa.20294

[SST06] A. Sankar, D. Spielman, S.-H. Teng, Smoothed Analysis of the Condition Numbers and Growth Factors of Matrices, *SIAM J. on Matrix Analysis and Applics.*, **28**, **2**, 446–476, 2006.

[TYUC17] J. A. Tropp, A. Yurtsever, M. Udell, V. Cevher, Practical Sketching Algorithms for Low-rank Matrix Approximation, *SIAM J. Matrix Anal. Appl.*, **38, 4**, 1454–1485, 2017. Also see arXiv:1609.00048 January 2018.