Block Subsampled Randomized Hadamard Transform for Nyström Approximation on Distributed Architectures

Oleg Balabanov [1] **Matthias Beaupère** [1] **Laura Grigori** [2 3] **Victor Lederer** [1]

Abstract

This article introduces a novel structured random matrix composed blockwise from subsampled randomized Hadamard transforms (SRHTs). The block SRHT is expected to outperform well-known dimension reduction maps, including SRHT and Gaussian matrices on distributed architectures. We prove that a block SRHT with enough rows is an oblivious subspace embedding, i.e., an approximate isometry for an arbitrary low-dimensional subspace with high probability. Our estimate of the required number of rows is similar to that of the standard SRHT. This suggests that the two transforms should provide the same accuracy of approximation in the algorithms. The block SRHT can be readily incorporated into randomized methods for computing a low-rank approximation of a large-scale matrix, such as the Nyström method. For completeness, we revisit this method with a discussion of its implementation on distributed architectures.

1. Introduction

Randomization has become a powerful tool for tackling massive problems in numerical algebra and data science (Mahoney et al., 2011; Woodruff et al., 2014; Vershynin, 2018; Martinsson & Tropp, 2020). Modern randomized methods can, in particular, provide solutions to problems of dimensions beyond the reach of deterministic methods, and allow effective use of computational resources. Recent significant development has made them very reliable, and not just used as a last resort, as it was not so long ago. Along with increased efficiency, they can now provide strong accuracy guarantees with a user-specified failure probability that can be chosen extremely low, say 10^{-10}, without much impact on computational costs.

This article is concerned with randomized methods that are based on a dimension reduction, called sketching (Woodruff et al., 2014), with oblivious ℓ_2-subspace embeddings (OSEs) defined below.

Definition 1.1. Let $0 \leq \varepsilon < 1$ and $0 < \delta < 1$. A random matrix $\boldsymbol{\Omega} \in \mathbb{R}^{l \times n}$ is said to be a (ε, δ, d) OSE, if for any fixed d-dimensional subspace $V \subseteq \mathbb{R}^n$,

$$\forall \mathbf{x} \in V, \quad |\|\mathbf{x}\|_2^2 - \|\boldsymbol{\Omega}\mathbf{x}\|_2^2| \leq \varepsilon \|\mathbf{x}\|_2^2 \tag{1}$$

holds with probability at least $1 - \delta$.

It is a consequence of the Johnson-Lindenstrauss lemma (Johnson & Lindenstrauss, 1984) that there exist (ε, δ, d) OSEs of sizes $l = \mathcal{O}(\varepsilon^{-2}(d + \log \frac{1}{\delta})$. The fact that n does not appear in the right-hand-side and the logarithmic dependence on the probability of failure δ shows the potential of a dimension reduction with such embeddings. There are several distributions that are known to satisfy the OSE property with the optimal or close to optimal l. The Gaussian, Rademacher distributions, sub-sampled randomized Hadamard transform (SRHT), sub-sampled randomized Fourier transform, and CountSketch matrix are ones of the most popular distributions. The random sketching matrix in the algorithm should be chosen depending on the computational architecture to yield the most benefit. For instance, the SRHT is a structured matrix that can be efficiently applied to a vector in a sequential computational environment, while the application of a Rademacher matrix is efficient in a highly parallel environment. In this paper we propose a novel OSE, called block SRHT, which should be superior to all currently existing ones, including the recent (Teng et al., 2020; Charalambides et al., 2022), on a distributed computational architecture, with not too many processors.

The OSEs are used in a variety of randomized methods for machine learning, scientific computing, and signal processing. Perhaps one of the most representative applications is

[1]Sorbonne Université, Inria, CNRS, Université de Paris, Laboratoire Jacques-Louis Lions, Paris, France. [2]Work performed while at Sorbonne Université, Inria, CNRS, Université de Paris, Laboratoire Jacques-Louis Lions, Paris, France. [3]Paul Scherrer Institute, Laboratory for Simulation and Modelling, 5232 PSI Villigen, and École Polytechnique Fédérale de Lausanne (EPFL), Institute of Mathematics, 1015 Lausanne, Switzerland. Correspondence to: Oleg Balabanov <olegbalabanov@gmail.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

the linear regression problem. Suppose that we seek a vector $\mathbf{x} \in \mathbb{R}^d$ that minimizes $\|\mathbf{Ax} - \mathbf{b}\|_2$, where $\mathbf{A} \in \mathbb{R}^{n\times d}$ is a large-scale dense matrix, $\mathbf{b} \in \mathbb{R}^n$ is a large-scale vector, and $d \ll n$. It follows that the solution to this problem can be approximated by a minimizer of $\|\mathbf{\Omega}(\mathbf{Ax} - \mathbf{b})\|_2$ requiring considerably lower computational cost. The accuracy of such an approximation is guaranteed, given that $\mathbf{\Omega}$ is $(\varepsilon, \delta, d + 1)$ OSE. Besides the linear regression problem, the sketching technique with OSEs has been successfully applied to the nearest neighbors problem (Ailon & Chazelle, 2006), approximation of products of matrices (Sarlos, 2006), computation of low-rank approximations of matrices (Halko et al., 2011) and tensor decompositions (Sun et al., 2020), dictionary learning (Anaraki & Hughes, 2013), solution of parametric equations (Balabanov & Nouy, 2019), and solution of linear systems and eigenvalue problems (Balabanov & Grigori, 2022; Nakatsukasa & Tropp, 2021).

In this paper the potential of the block SRHT is realized on the low-rank approximation problem. Such problems are ubiquitous, for instance, in the principal component analysis of large data sets and kernel ridge regression. A randomized low-rank approximation for machine learning tasks was addressed in e.g. (Bach, 2013; Alaoui & Mahoney, 2015; Derezinski et al., 2020). In (Zhang et al., 2013; Calandriello et al., 2016; Rudi et al., 2017; Meanti et al., 2020; Yin et al., 2021) a particular focus was given to make the methods suited to modern architectures. In details, given a large matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with rapidly decaying spectrum, we seek a matrix $\mathbf{B}_k$ preferably in an SVD form, of rank at most $k \ll \min(m, n)$, that approximates well $\mathbf{A}$. The matrix $\mathbf{B}_k$ can be obtained by first restricting its range to a subspace that captures the most of the action of $\mathbf{A}$, and then minimizing the chosen error measure, say the spectral error $\|\mathbf{A} - \mathbf{B}_k\|_2$. As shown in (Halko et al., 2011), the most of $\mathbf{A}$'s action can be well captured by the range of $\mathbf{A\Omega}^{\mathrm{T}}$, which constitutes the core of state-of-the-art RSVD algorithm. Over the past years more sophisticated randomized low-rank approximation methods have been developed such as the Nyström method for spd matrices (Gittens & Mahoney, 2013; Tropp et al., 2017a), and the single-view approximations for general matrices (Kannan et al., 2014; Upadhyay, 2016; Tropp et al., 2019). In this work, we chose the Nyström method as a representative.

The paper is organized as follows. The rest of Section 1 discusses contributions and relation to prior work. Section 2 introduces a block SRHT matrix and discusses its properties. In Section 3 we present Nyström algorithm for computing a low-rank approximation based on oblivious embeddings. Section 4 contains some computational aspects and experimental results. The proof of the main theoretical result is given in Section 5. Section 6 concludes this work.

Contributions

The proposed block SRHT matrix has the potential to combine the benefits of both structured OSEs, such as SRHT, and unstructured OSEs, such as Gaussian matrices, in terms of complexity and performance. Recent works in this direction include (Teng et al., 2020; Charalambides et al., 2022). The OSE from (Charalambides et al., 2022) generalizes the regular SRHT by sampling blocks instead of rows, which can improve efficiency in asynchronous systems. However, this approach requires communication-expensive computation of a global Hadamard transform and a global permutation. The OSE from (Teng et al., 2020) has a high relation to our proposed block SRHT matrix. It also uses local Hadamard transforms to reduce communication costs. However, it involves a global permutation on the right and samples rows locally without aggregating the contributions. Because of these differences, our block SRHT should be more efficient on distributed architectures and have greater ability for dimensionality reduction. In fact, we demonstrate that the block SRHT matrix of size $l = \mathcal{O}(\varepsilon^{-2}(d + \log \frac{n}{\delta}) \log \frac{d}{\delta})$ satisfies the (ε, δ, d) OSE property. In contrast, the required size for the OSE in (Teng et al., 2020) can be up to p times larger, where p denotes the number of processors (see Theorem 4 in (Teng et al., 2020)). Notably, our bound for l is close to that for the regular SRHT from the literature (Tropp, 2011; Boutsidis & Gittens, 2013). This result does not simply follow from the analysis in (Tropp, 2011; Boutsidis & Gittens, 2013), and particularly requires incorporation of a useful technical trick that, to the best of our knowledge, was not employed before in the randomized numerical linear algebra community.

We also describe important aspects of implementation of the Nyström approximation on distributed architectures, using the suitability of block SRHT for these architectures, and provide a rigorous characterization of the accuracy of Nyström method based solely on OSEs property.

2. Block SRHT

For n being a power of two, an SRHT matrix can be defined as follows:

$$\mathbf{\Omega} = \sqrt{\tfrac{n}{l}}\mathbf{RHD}, \tag{2}$$

where $\mathbf{R}$ is a $l \times n$ uniform, with or without replacement, random sampling matrix, $\mathbf{H}$ is a $n \times n$ Walsh-Hadamard matrix rescaled by $\frac{1}{\sqrt{n}}$, and $\mathbf{D}$ is a diagonal matrix with i.i.d. Rademacher random variables ± 1 on the diagonal. The properties of SRHT were thoroughly described in (Tropp, 2011) with a follow up analysis in (Boutsidis & Gittens, 2013). SRHT matrices are commonly used in randomized algorithms as they can be applied to vectors using only $n \log_2 n$ flops, while general unstructured matrices require $2nl$ flops. At the same time, they satisfy the

(ε, δ, d) OSE property, if (Balabanov & Nouy, 2019)

$$l \geq 3\varepsilon^{-2}(\sqrt{d} + \sqrt{8\log\tfrac{6n}{\delta}})^2 \tfrac{3d}{\delta}, \tag{3}$$

which is only by a logarithmic factor in δ and n larger than the optimal bound. For a general n, a partial SRHT can be used that is defined as the first n columns of an SRHT matrix. Using a partial SRHT is equivalent to padding the input data with zeros to make its dimension a power of two.

Unfortunately, products with SRHT matrices are not well suited to distributed computing limiting the benefits of SRHT on modern architectures (see e.g. (Yang et al., 2015)). This happens majorly due to computing products with $\mathbf{H}$ in tensor form

$$\mathbf{H} = \tfrac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \ldots \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

requiring cumbersome reduction operator such as a sequence of arrays of butterflies, rather than a simple addition, which we have with Gaussian matrices. This article attempts to alleviate this problem by constructing $\mathbf{\Omega}$ blockwise as follows

$$\mathbf{\Omega} = [\mathbf{\Omega}^{(1)} \quad \mathbf{\Omega}^{(2)} \quad \ldots \quad \mathbf{\Omega}^{(p)}], \tag{4}$$

where $\mathbf{\Omega}^{(i)} = \sqrt{\frac{r}{l}}\widetilde{\mathbf{D}}^{(i)}\mathbf{R}\mathbf{H}\mathbf{D}^{(i)}$ are $l \times r$ SRHT matrices related to a unique sampling matrix $\mathbf{R}$ and different (independent from each other) diagonal matrices $\mathbf{D}^{(i)}$ with i.i.d. Rademacher entries ± 1, multiplied from the left by another diagonal matrices $\widetilde{\mathbf{D}}^{(i)}$ with Rademacher entries, $r = \frac{n}{p}$, $1 \leq i \leq p$. As in the standard SRHT, the condition that r is a power of two can be achieved by zero padding of the input data. The advantage of $\mathbf{\Omega}$ defined by (4) is that it can be multiplied by an $n \times d$ matrix $\mathbf{V}$ distributed between p processors with rowwise partitioning, as

$$\mathbf{\Omega}\mathbf{V} = \sum_{1 \leq i \leq p} \mathbf{\Omega}^{(i)}\mathbf{V}^{(i)}, \tag{5}$$

where $\mathbf{V}^{(i)}$ are the corresponding local blocks of rows of $\mathbf{V}$. In this way, to obtain $\mathbf{\Omega}\mathbf{V}$ one can compute the local contributions $\mathbf{\Omega}^{(i)}\mathbf{V}^{(i)}$ on each processor and then sum-reduce them to the master processor. This makes block SRHT matrices have the same application cost in terms of communication as Gaussian matrices. Thus, they should yield much better scalability of computations than standard SRHT (Yang et al., 2015). The sum-reduce operation requires exchanging $\mathcal{O}(\log p)$ messages and $\mathcal{O}(dl \log p)$ per-processor communication volume that can be by a factor $\mathcal{O}(\frac{r}{l})$ less than the volume of communication used by standard SRHT (if $l \leq r$). At the same time, block SRHT require less flops per processor than Gaussian matrices. To be more specific, the application cost of block SRHT using (5) is only $\mathcal{O}(rd \log r + dl \log p)$ flops per processor, while Gaussian matrices require $\mathcal{O}(rdl + dl \log p)$ flops per processor. It is deduced that block SRHT matrices are both well-suited to distributed computing and efficient in terms of flops. They are expected to outperform all existing oblivious embedding when the local dimension r and the sampling dimension l are large enough.

The procedure for application of the block SRHT can be easily extended to the case when $\mathbf{V}$ is distributed with a 2D partitioning. Namely, to multiply $\mathbf{\Omega}$ by $n \times n$ matrix $\mathbf{V}$ distributed over a grid of $p \times p$ processors, we first compute the local contributions $\mathbf{X}^{(i,j)} = \mathbf{\Omega}^{(i)}\mathbf{V}^{(i,j)}$ on each processor, and then sum-reduce the contributions from the j-th column of blocks to the processor $(1, j)$, $1 \leq j \leq p$. Note that in this case the resulting matrix $\mathbf{Y}^{\mathrm{T}} = \mathbf{\Omega}\mathbf{A}$ is distributed with rowwise partitioning over processors $(1,1), (1,2), \ldots, (1,p)$. This provides the ability to efficiently compute the sketch $\mathbf{\Omega}\mathbf{Y}$, or to orthogonalize $\mathbf{Y}$ with a routine suited for distributed computing, with no need to reorganize $\mathbf{Y}$. This can be particularly handy in the low-rank approximation algorithm from Section 3.

We will assume that $\mathbf{R}$ in (4) samples rows uniformly at random and *with replacement*. Interestingly, in this case the block SRHT can be viewed as a generalization of the SRHT with replacement and the Rademacher embedding, as it reduces to these maps when $p = 1$ and $p = n$, respectively. Sampling with replacement can be important, for instance, when $r = \frac{n}{p}$ is smaller than the dimension of the embedded subspace.

Theorem 2.1 is the main result of the article. It implies the compatibility of the block SRHT with all randomized methods that rely on OSEs, including the methods in Section 3. The estimate of l in Theorem 2.1 is similar to (3) for the standard SRHT matrix, and in particular depends only logarithmically on n and δ.

Theorem 2.1 (Main Theorem)**.** *Let $0 < \varepsilon < 1$ and $0 < \delta < 1$. Let $\mathbf{\Omega} \in \mathbb{R}^{l \times n}$ be defined by (4). If,*

$$n \geq l \geq 3.7\varepsilon^{-2}(\sqrt{d} + 4\sqrt{\log\tfrac{n}{\delta} + 6.3})^2 \log\tfrac{5d}{\delta},$$

then $\mathbf{\Omega}$ is an (ε, δ, d) OSE.

For better presentation the proof of Theorem 2.1 is deferred to the end of the article (see Section 5).

3. Application to the Nyström method

This section addresses the problem of efficient computation of a rank-k approximation of a large positive semi-definite matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ with rapidly decaying spectrum, with the Nyström method. A particular focus is given to the scenario where $\mathbf{A}$ is (uniformly) distributed over a 2D grid of processors.

3.1. Nyström method

It is a well-known fact that the best rank-k approximation of $\mathbf{A}$ in terms of the spectral, trace and Frobenius error is given by $[\![\mathbf{A}]\!]_k := \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{U}_k^{\mathrm{T}}$, where $\mathbf{\Sigma}_k$ is a diagonal matrix of k dominant singular values of $\mathbf{A}$, and $\mathbf{U}_k$ contains the associated singular vectors. In other words, we have

$$[\![\mathbf{A}]\!]_k = \arg\min_{\mathrm{rank}(\mathbf{B})=k} \|\mathbf{A}-\mathbf{B}\|_\xi, \tag{6}$$

where $\xi = 2, *$ or F.

Obtaining the best rank-k approximation can be computationally expensive and often becomes the bottleneck of an algorithm. In such case, one has to turn to alternative methods for computing a low-rank approximation, such as the Nyström method described below.

We first notice that a low-rank approximation of $\mathbf{A}$ can be interpreted as reduction of the range of $\mathbf{A}$ to a low-dimensional subspace Q capturing the most of $\mathbf{A}$'s action. The SVD approximation $[\![\mathbf{A}]\!]_k$ corresponds to taking Q as $\mathrm{range}(\mathbf{U}_k)$. A more efficient way is to take $Q = \mathrm{range}(\mathbf{A}\mathbf{\Omega}^{\mathrm{T}})$, where $\mathbf{\Omega} \in \mathbb{R}^{l\times n}$ is an OSE (Halko et al., 2011; Woodruff et al., 2014). Furthermore, it can be computationally beneficial to change the norm $\|\cdot\|_\xi$ in (6) to its sketched estimate $\|\mathbf{\Omega}\cdot\mathbf{\Omega}^{\mathrm{T}}\|_\xi$. The accuracy of such an estimation can be guaranteed thanks to the fact that $\mathbf{\Omega}$ is an OSE. This leads to Nyström approximation $[\![\mathbf{A}]\!]^{(\mathrm{Nyst})}$ given below

$$[\![\mathbf{A}]\!]^{(\mathrm{Nyst})} := \arg\min_{\mathrm{range}(\mathbf{B})\subseteq Q} \|\mathbf{\Omega}(\mathbf{A}-\mathbf{B})\mathbf{\Omega}^{\mathrm{T}}\|_\xi, \tag{7}$$

or in a more usual form (Drineas et al., 2005; Gittens & Mahoney, 2013; Chiu & Demanet, 2013; Alaoui & Mahoney, 2015; Tropp et al., 2017a):

$$[\![\mathbf{A}]\!]^{(\mathrm{Nyst})} = (\mathbf{\Omega}\mathbf{A})^{\mathrm{T}}(\mathbf{\Omega}\mathbf{A}\mathbf{\Omega}^{\mathrm{T}})^{\dagger}(\mathbf{\Omega}\mathbf{A}),$$

where $(\mathbf{\Omega}\mathbf{A}\mathbf{\Omega}^{\mathrm{T}})^{\dagger}$ denotes the pseudo-inverse of $\mathbf{\Omega}\mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$. Then a rank-$k$ approximation of $\mathbf{A}$ can be obtained by an SVD of $[\![\mathbf{A}]\!]^{(\mathrm{Nyst})}$, which leads to the approximation $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})} := [\![[\![\mathbf{A}]\!]^{(\mathrm{Nyst})}]\!]_k$. This way of obtaining a rank-k approximation from $[\![\mathbf{A}]\!]^{(\mathrm{Nyst})}$ is referred to as the modified fixed-rank Nyström via QR (Tropp et al., 2017a; Pourkamali-Anaraki et al., 2018; Pourkamali-Anaraki & Becker, 2019).

Algorithm 1 describes a way for computing $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}$ suited for distributed computing under 2D partitioning of $\mathbf{A}$. The matrices $\mathbf{Y}$ and $\mathbf{\Omega}\mathbf{Y}$ can be computed with the procedure from Section 2 using the block structure of $\mathbf{\Omega}$. The QR factorization $\mathbf{Z} = \mathbf{Q}\mathbf{R}$ in step 4 can be computed with TSQR (Demmel et al., 2012) or other methods having low communication cost. Note that in step 6, instead of computing $\widehat{\mathbf{U}}_k$ as $(\mathbf{Y}\tilde{\mathbf{V}}_k)\tilde{\mathbf{\Sigma}}_k^{-1}$ we could use $\widehat{\mathbf{U}}_k = \mathbf{Q}\tilde{\mathbf{U}}_k$, which would provide more numerical stability but entail a larger computational cost.

Algorithm 1 needs only one pass over the matrix $\mathbf{A}$, and does not involve any high-dimensional operations on $\mathbf{A}$ except the computation of the sketch $\mathbf{Y} = \mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$, which implies its superiority over the standard SVD as well as randomized SVD (Halko et al., 2011; Tropp et al., 2017a). In fact, the dominant computational cost of Algorithm 1 is associated with computing $\mathbf{Y}$ and $\mathbf{\Omega}\mathbf{Y}$ in steps 1 and 2, when r is sufficiently large, the efficiency of which is ensured by the block SRHT.

Algorithm 1 Randomized Nyström approximation

Require: matrix $\mathbf{A}$, OSE $\mathbf{\Omega}$, target rank k.
1: Compute $\mathbf{Y} = \mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$.
2: Obtain a Cholesky factor $\mathbf{C}$ of $\mathbf{\Omega}\mathbf{Y}$.
3: Compute $\mathbf{Z} = \mathbf{Y}\mathbf{C}^{-1}$ with backward substitution.
4: Obtain the R factor $\mathbf{R}$ of $\mathbf{Z}$ with TSQR or similar.
5: Use SVD to compute the best rank-k approximation $\tilde{\mathbf{U}}_k\tilde{\mathbf{\Sigma}}_k\tilde{\mathbf{V}}_k^{\mathrm{T}}$ of $\mathbf{R}$.
6: Compute $\widehat{\mathbf{U}}_k = (\mathbf{Y}\tilde{\mathbf{V}}_k)\tilde{\mathbf{\Sigma}}_k^{-1}$.
7: Output factorization $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})} = \widehat{\mathbf{U}}_k\tilde{\mathbf{\Sigma}}_k^2\widehat{\mathbf{U}}_k^{\mathrm{T}}$.

Remark 3.1. The matrix $\mathbf{\Omega}\mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$ can be rank-deficient, for instance, if $\mathbf{A}$ or $\mathbf{\Omega}$ have lower rank than l, which will cause a problem for obtaining a Cholesky factorization in step 2. In this case, a remedy can be to compute an SVD instead of the Cholesky factorization, and take $\mathbf{C}$ as a square root of $\mathbf{\Omega}\mathbf{Y}$ in SVD form, that then can be used for the pseudo-inversion in step 3. Another possibility is to make $\mathbf{A}$ full-rank by using shifting as in (Li et al., 2017; Tropp et al., 2017a).

The accuracy of the Nyström approximation can be guaranteed with high probability if $\mathbf{\Omega}$ is an OSE of sufficiently large dimension. Such a guarantee is derived in Appendix A.

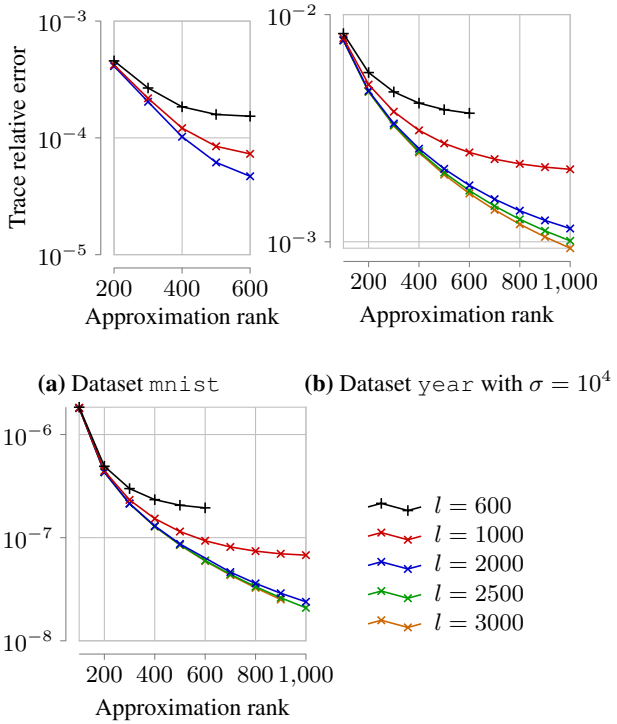
4. Numerical experiments

The validation of block SRHT is done through comparison with Gaussian embeddings. In the plots, BSRHT refers to block SRHT. The comparison with standard SRHT is impertinent since SRHT matrices are not that well scalable as Gaussian matrices and have no better accuracy (Yang et al., 2015).

4.1. Nyström approximation

This experiment was executed with Julia programming language version 1.7.2 along with the Distributed.jl and DistributedArrays.jl packages for parallelism. We used 2 nodes Intel Skylake 2.7GHz (AVX512) having 48 available cores and 180 MB of RAM each. In this experi-

ment we used only 32 cores on each node. The code is available at https://github.com/matthiasbe/block_srht. As input data we took the MNIST or YearPredictionMSD datasets (Lecun et al., 1998; Bertin-Mahieux et al., 2011). The radial basis function $e^{-\|x_i - x_j\|^2/\sigma^2}$ was used to build a dense positive definite matrix $\mathbf{A}$ of size $n \times n$ from n rows of the input data. The parameter σ was chosen as 100 for the MNIST dataset and 10^4 as well as 10^5 for the YearPredictionMSD dataset. The dimension n was taken as 65536. The matrix $\mathbf{A}$ has been uniformly distributed on a square grid of 8×8 processors. In all the experiments, the local matrices $\mathbf{\Omega}^{(i)}$ on each processor were generated with a seeded random number generator with a low communication cost. Figure 1 depicts the



(a) Dataset mnist **(b)** Dataset year with $\sigma = 10^4$

(c) Dataset year with $\sigma = 10^5$

Figure 1: Trace error $\|\mathbf{A} - [\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}\|_*/\|\mathbf{A}\|_*$ using BSRHT.

convergence of the error of the low-rank approximation obtained with Algorithm 1 taking $\mathbf{\Omega}$ as a block SRHT. The results for Gaussian $\mathbf{\Omega}$ are practically identical and therefore are not displayed. In this numerical experiment, the error is measured with the trace norm. Different sketching sizes l were tested. For each pair of parameters (l, k) 20 different approximations were computed for each type of $\mathbf{\Omega}$, in order to have the 95% confidence interval. Nevertheless this interval is not displayed as it is too small to be visible. Figure 2 gives runtime characterization. In particular we depict the runtime spent on computing $\mathbf{Y} = \mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$ and $\mathbf{\Omega}\mathbf{Y}$ in steps 1 and 2 of Algorithm 1. These operations will dominate the overall computational cost, when the block size is large enough. Nevertheless the reader should be aware that TSQR and the SVD of $\mathbf{R}$ (step 4 and 5) are also important, especially when the sampling size is close to the block size. The parameter k is not involved in steps 1 and 3 hence not mentioned in Figure 2.

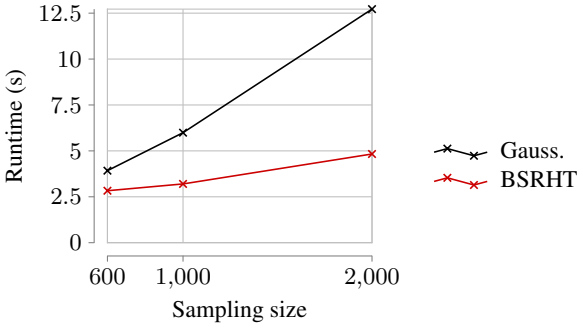


Figure 2: Runtimes of computing $\mathbf{Y} = \mathbf{A}\mathbf{\Omega}^{\mathrm{T}}$ and $\mathbf{\Omega}\mathbf{Y}$ in Algorithm 1 for different sampling sizes.

According to Figure 2, the runtime of the Gaussian sampling is up to 2.5 times higher and grows faster with l than the runtime taken by block SRHT. Note that for block SRHT the local computation cost is independent of l, hence the slope comes only from the reductions in steps 1 and 2. On the other hand, the Gaussian sampling involves local computations with linear dependency in l, in addition to these reductions.

4.2. Cost of application to tall-and-skinny matrix

Next we investigate the performance of block SRHT on larger scale. For this we consider a product of $\mathbf{\Omega}$ with a tall-and-skinny matrix $\mathbf{V}$, for instance in the context of solving an overdetermined least-squares problem. The same computing environment is used as in the previous experiment, involving now up to 32 nodes and using C99/MPI instead of Julia. The code was compiled using IntelMPI C compiler version 20.0.2 and sequential MKL 20.0.2 with option ILP64. The library FFTW3 used has Intel-specific routines. There is therefore up to 1536 cores available. In this way we generated a random matrix $\mathbf{V}$ with $d = 200$ columns and a varying number n of rows. This matrix was distributed among a varying number p of processors with block rowwise partitioning. Then $\mathbf{V}$ was multiplied by either a Gaussian or block SRHT matrix $\mathbf{\Omega}$ with $l = 2000$ rows using (5). In all experiments, the local $\mathbf{\Omega}^{(i)}$ matrices on each processor were generated with a seeded random number generator with negligible communication cost. Figure 3 presents a strong scalability test for $n = 10^7$. We see that the block SRHT provides an overall speedup by a factor of more than 2.5 over the Gaussian matrices,

while demonstrating as good scalability when $p \leq 384$. For larger p, however, the reduction operation starts to dominate, which reduces the gain in efficiency. We observe a variability in the MPI_Allreduce operation on larger number of processors for both Gaussian and block SRHT algorithms. However the compute times for both algorithms scale well when increasing the number of processors up to $p = 1536$. Figure 4 shows a strong scalability test for a higher dimension $n = 10^8$. Again we see a great scalability of block SRHT for $p \leq 384$. For Gaussian matrices, on the other hand, we revealed issues with reaching the memory limit needed to store $\mathbf{\Omega}^{(i)}$ which made its application on $p \leq 384$ processors infeasible. In principle, this problem can be overcome by generating $\mathbf{\Omega}^{(i)}$ blockwise and applying the blocks to $\mathbf{V}^{(i)}$ "on the fly". This however entails a dramatic increase in runtime and therefore is omitted in comparison. On the other hand, for block SRHT we do not have any memory problems[1]. To quantify the advantage of block SRHT in such context, in Figure 5 we provide the memory consumption of the Gaussian and block SRHT matrix. We see that in this sense the reduction in computational cost is indeed drastic. Finally, Figure 6 provides a weak scalability test using $n = rp$ where $r = 10^5$. We again see a reduction in runtime of about 2.5 and good scalability for block SRHT up to using $p = 1536$ processors, similar as in the strong scalability test.

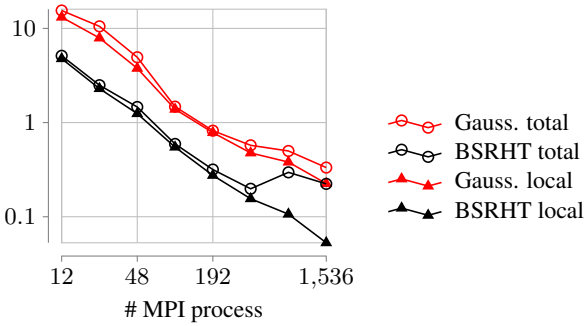


Figure 3: Strong scalability runtimes associated with computing $\mathbf{\Omega V}$ with $n = 10^7$ and $l = 2000$, versus p. "Gauss. total" and "BSRHT total" correspond to the overall runtimes, whereas "Gauss. local" and "BSRHT local" stand for the max per-processor runtimes taken by local multiplications.

5. Proof of the main theorem

Before providing the proof for Theorem 2.1, let us first motivate the chosen proof path. Le $\mathbf{V}$ be a fixed $n \times d$ matrix with orthonormal columns, partitioned using block rowwise partitioning with p blocks $\mathbf{V}^{(i)}$ of size $r \times d$. The statement of the theorem can then be proven by showing that the singular values of $\mathbf{\Omega V}$ belong to the interval

[1]To reduce the memory consumption, local matrices $\mathbf{V}^{(i)}$ are multiplied by $\mathbf{\Omega}^{(i)}$ in blocks of 20 columns.

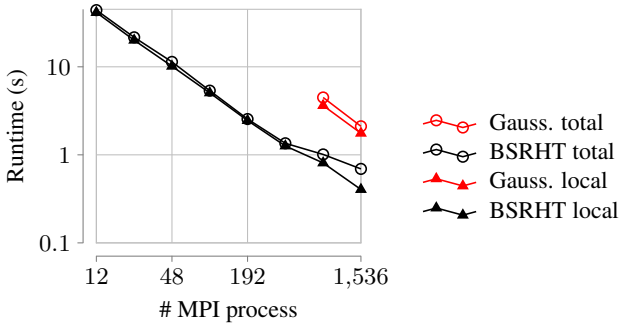


Figure 4: Strong scalability runtimes associated with computing $\mathbf{\Omega V}$ with $n = 10^8$ and $l = 2000$, versus p. "Gauss. total" and "BSRHT total" correspond to the overall runtimes, whereas "Gauss. local" and "BSRHT local" stand for the max per-processor runtimes taken by local multiplications.

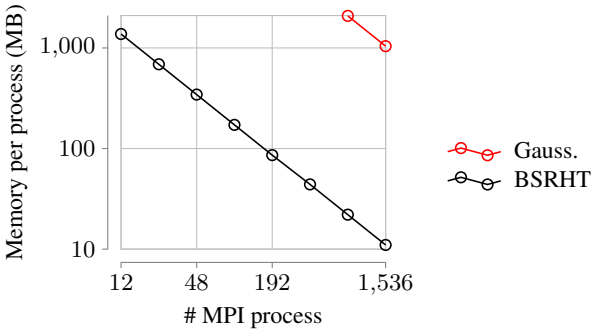


Figure 5: Max per-processor memory needed for computing $\mathbf{\Omega V}$ with $n = 10^8$ and $l = 2000$, versus p.

$[\sqrt{1-\varepsilon}, \sqrt{1+\varepsilon}]$ with probability at least $1 - \delta$.

Assume for a moment, that $\mathbf{R}$ in (4) is a uniform sampling matrix *without replacement*. Notice that the random sampling of rows without replacement and then flipping their signs is equivalent to first flipping the signs and then sampling. By using this consideration, the expression (5) can be developed further as $\mathbf{\Omega V} =$

$$\sqrt{\frac{r}{l}} \sum_{i=1}^{p} \left(\widetilde{\mathbf{D}}^{(i)} \mathbf{R} \mathbf{H} \mathbf{D}^{(i)} \mathbf{V}^{(i)} \right) = \sqrt{\frac{r}{l}} \sum_{i=1}^{p} \left(\mathbf{R} \widehat{\mathbf{D}}^{(i)} \mathbf{H} \mathbf{D}^{(i)} \mathbf{V}^{(i)} \right)$$
$$= \sqrt{\frac{r}{l}} \mathbf{R} \sum_{i=1}^{p} \left(\widehat{\mathbf{D}}^{(i)} \mathbf{H} \mathbf{D}^{(i)} \mathbf{V}^{(i)} \right) = \sqrt{\frac{r}{l}} \mathbf{R} \mathbf{W} \mathbf{V}, \tag{8}$$

where $\widehat{\mathbf{D}}^{(i)}$ are $r \times r$ diagonal matrices with Rademacher random variables ± 1 on the diagonal, and $\mathbf{W} = \left[\widehat{\mathbf{D}}^{(1)} \mathbf{H} \mathbf{D}^{(1)}, \widehat{\mathbf{D}}^{(2)} \mathbf{H} \mathbf{D}^{(2)}, \ldots, \widehat{\mathbf{D}}^{(l)} \mathbf{H} \mathbf{D}^{(l)} \right]$.

Looking at (5), one can detect many similarities of $\mathbf{\Omega}$ with standard SRHT matrix. Consequently, in order to argue that $\mathbf{\Omega V}$ is approximately orthonormal, the first thing to try should be to follow the steps from (Tropp, 2011) in the

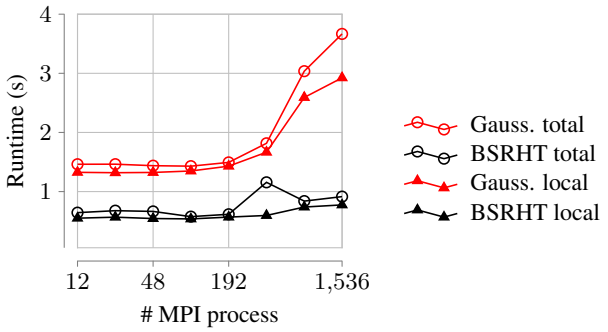


Figure 6: Weak scalability runtimes associated with computing $\mathbf{\Omega V}$ with $n = 10^5 \times p$ and $l = 2000$, versus p. "Gauss. total" and "BSRHT total" correspond to the overall runtimes, whereas "Gauss. local" and "BSRHT local" stand for the max per-processor runtimes taken by local multiplications.

analysis of the original SRHT. In this case the proof recipe would be as follows. First, it could be shown that the matrix $\mathbf{W}$ with high probability homogenizes the rows of $\mathbf{V}$. This result then would allow the Matrix Chernoff concentration inequality from (Tropp, 2011) to be applied to show that $\mathbf{WV}$ and $\sqrt{\frac{r}{l}}\mathbf{RWV}$ have approximately equal minimal and maximal singular values. With these results, it would remain to show that with high probability $\mathbf{WV}$ is approximately orthonormal. This, however, can be cumbersome or even impossible in some situations. Think, for example, of the situation when $r < d$. Therefore, we will assume that $\mathbf{R}$ is a uniform sampling matrix *with replacement* and use the following trick. For better presentation define parameters $\varepsilon^* = \frac{15}{16}\varepsilon$ and $\delta^* = \frac{\delta}{5}$.

Recall that the sampling matrix $\mathbf{R}$ restricts a vector $\mathbf{x} = (x_1, \ldots, x_r)$ to l coordinates, i.e., we have

$$\mathbf{Rx} = (x_{i_1}, \ldots, x_{i_l}), \text{ with } 1 \le i_1, \ldots, i_l \le r.$$

The (multi-)set of indices $\{i_1, \ldots, i_l\}$ is a uniform random sample of $\{1, \ldots, r\}$ with replacement. Notice that such sampling of indices is equivalent to the sampling uniformly at random with replacement from $\{1, \ldots, 1, 2, \ldots 2, \ldots, r, \ldots, r\}$ containing $K = \lceil 10^4 \frac{n^2}{r\delta^*} \rceil$ copies of each index. This observation implies that the sampling matrix $\mathbf{R}$ satisfies the identity $\mathbf{RH} = \widehat{\mathbf{R}}[\mathbf{H}\ \mathbf{H} \ldots \mathbf{H}]^{\mathrm{T}} = \widehat{\mathbf{R}}\widehat{\mathbf{H}}$, where $\widehat{\mathbf{R}}$ is uniform sampling, with replacement, matrix of size $l \times rK$, and $\widehat{\mathbf{H}}$ is a block matrix with K blocks of rows, each being equal to $\mathbf{H}$. For a vector $\mathbf{x} = (x_1, \ldots, x_{rK})$, matrix $\widehat{\mathbf{R}}$ satisfies

$$\widehat{\mathbf{R}}\mathbf{x} = (x_{i_1}, \ldots, x_{i_l}), \text{ with } 1 \le i_1, \ldots, i_l \le rK, \quad (9)$$

where the indices $\{i_1, \ldots, i_l\}$ are drawn uniformly at random *with replacement* from $\{1, \ldots, rK\}$. Let $\mathcal{S}$ denote the event when $i_1, \ldots, i_l$ in (9) are all disjoint indices.

Lemma 5.1. *$\mathcal{S}$ occurs with probability at least $1 - \delta^*$.*

Proof. There are in total $\frac{(rK)^l}{l!}$ ways to select l elements from a (rK)-element set and $\binom{rK}{l} = \frac{rK(rK-1)\ldots(rK-l+1)}{l!}$ ways to select l disjoint elements. Consequently, we have

$$\mathbb{P}(S) = \prod_{i=1}^{l}(1 - \frac{i-1}{rK}) \ge (1 - \frac{l}{rK})^l \ge 1 - \frac{l^2}{rK} \ge 1 - \delta^*.$$

□

The goal will be to bound the singular values of $\mathbf{\Omega V}$ under the condition $\mathcal{S}$. The overall probability of success, then will follow by the union bound argument. Next is assumed that $\mathcal{S}$ is occurring. Notice that, in this case, matrix $\widehat{\mathbf{R}}$ is equivalent to the matrix that samples the entries uniformly at random and *without replacement*. Then, using the same arguments as in (8), we have the following expression for the product $\mathbf{\Omega V}$:

$$\mathbf{\Omega V} = \sqrt{\frac{r}{l}}\sum_{i=1}^{p}\left(\widetilde{\mathbf{D}}^{(i)}\widehat{\mathbf{R}}\widehat{\mathbf{H}}\mathbf{D}^{(i)}\mathbf{V}^{(i)}\right) = \sqrt{\frac{r}{l}}\widehat{\mathbf{R}}\widehat{\mathbf{W}}\mathbf{V}, \quad (10)$$

where $\widehat{\mathbf{W}}$ is a block matrix composed of $K \times p$ blocks, with the (j,i)-th block being $\widehat{\mathbf{D}}^{(i,j)}\mathbf{HD}^{(i)}$, where $\widehat{\mathbf{D}}^{(i,j)}$ are diagonal matrices with entries i.i.d. Rademacher random variables ± 1. Unlike $\mathbf{WV}$, the matrix $\widehat{\mathbf{W}}\mathbf{V}$ (rescaled by $1/\sqrt{K}$) for sufficiently large K can be proven to be approximately orthonormal with high probability. We are ready to establish the proof of Theorem 2.1.

Notice that the condition in Theorem 2.1 implies that

$$n \ge l \ge 3.2\varepsilon^{*-2}(\sqrt{d} + \sqrt{8\log(rK/\delta^*)})^2\log(d/\delta^*). \quad (11)$$

In Proposition 5.2 is shown that, given $\mathcal{S}$, the matrix $\widehat{\mathbf{W}}\mathbf{V}$ has rows with equilibrated norms.

Proposition 5.2. *Given $\mathcal{S}$. The rows $\varphi^{(j)}$ of $\widehat{\mathbf{W}}\mathbf{V}$ satisfy*

$$\mathbb{P}\left(\max_{j=1,\ldots,rK}\|\varphi^{(j)}\|_2 \le \sqrt{\frac{d}{r}} + \sqrt{\frac{8\log(rK/\delta^*)}{r}}\right) \ge 1 - \delta^*.$$

Proof. Notice that, each row of $\widehat{\mathbf{W}}$ has entries that are i.i.d Rademacher random variables rescaled by $1/\sqrt{r}$. Consequently, we have

$$\varphi^{(j)} = {\xi^{(j)}}^{\mathrm{T}}\mathbf{V}/\sqrt{r},$$

where $\xi^{(j)}$ is a Rademacher vector. Define convex function $f(\mathbf{x}) = \|\mathbf{x}^{\mathrm{T}}\mathbf{V}/\sqrt{r}\|_2$. Observe that $f(\mathbf{x})$ satisfies the Lipschitz bound:

$$\forall \mathbf{x}, \mathbf{y},\ |f(\mathbf{x}) - f(\mathbf{y})| \le \|\mathbf{x} - \mathbf{y}\|_2\|\mathbf{V}/\sqrt{r}\|_2 = \|\mathbf{x} - \mathbf{y}\|_2/\sqrt{r}.$$

This allows to apply the Rademacher tail bound: $\forall t \ge 0$,

$$\mathbb{P}\left(f(\xi^{(j)}) \ge \mathbb{E}f(\xi^{(j)}) + t/\sqrt{r}\right) \le \exp(-t^2/8). \quad (12)$$

Observe that $\mathbb{E}(f(\xi^{(j)})) \leq (\mathbb{E}(f(\xi^{(j)})^2))^{\frac{1}{2}} = \|\mathbf{V}/\sqrt{r}\|_{\mathrm{F}} \leq \sqrt{d/r}$. The statement of the lemma follows by combining this relation with (12) with $t = \sqrt{8 \log \frac{1}{\delta}}$ and using the union bound argument. □

In Proposition 5.3 is proven that $\frac{1}{\sqrt{K}}\widehat{\mathbf{W}}\mathbf{V}$ with high probability has singular values close to 1.

Proposition 5.3. *Given $\mathcal{S}$. The singular values of $\frac{1}{\sqrt{K}}\widehat{\mathbf{W}}\mathbf{V}$ with probability at least $1 - \delta^*$ lie inside the interval $[\sqrt{1-\varepsilon^*/30}, \sqrt{1+\varepsilon^*/30}]$.*

Proof. Define $\tau = \varepsilon^*/30$. Notice that

$$K \geq 10^4 l \geq 7.87\tau^{-2}(6.9d + \log(r/\delta^*)).$$

We have, for any $\mathbf{x} \in \mathbb{R}^d$,

$$\|\widehat{\mathbf{W}}\mathbf{V}\mathbf{x}\|_2^2 = \sum_{j=1}^{K} \| \sum_{i=1}^{r} \widehat{\mathbf{D}}^{(i,j)}\mathbf{H}\mathbf{D}^{(i)}\mathbf{V}^{(i)}\mathbf{x}\|_2^2. \tag{13}$$

Denote by $\mathbf{d}^{(k,j)}$ a vector with i-th entry equal to the (k,k)-th entry of matrix $\widehat{\mathbf{D}}^{(i,j)}$, $1 \leq k \leq r$. Denote by $\mathbf{Z}^{(k)}$ the matrix with i-th row equal to the k-th row of matrix $\mathbf{H}\mathbf{D}^{(i)}\mathbf{V}^{(i)}$, $1 \leq k \leq r$. Notice the following relation:

$$\| \sum_{i=1}^{r} \widehat{\mathbf{D}}^{(i,j)}\mathbf{H}\mathbf{D}^{(i)}\mathbf{V}^{(i)}\mathbf{x}\|_2^2 = \sum_{k=1}^{r} \langle \mathbf{d}^{(k,j)}, \mathbf{Z}^{(k)}\mathbf{x}\rangle^2, \tag{14}$$

with $1 \leq j \leq K$, and

$$\sum_{k=1}^{r} \|\mathbf{Z}^{(k)}\mathbf{x}\|_2^2 = \|\mathbf{V}\mathbf{x}\|_2^2. \tag{15}$$

We have $\frac{1}{K}\sum_{j=1}^{K} \langle \mathbf{d}^{(k,j)}, \mathbf{Z}^{(k)}\mathbf{x}\rangle^2 = \|\mathbf{\Theta}\mathbf{Z}^{(k)}\mathbf{x}\|_2^2$, where $\mathbf{\Theta}$ is a $K \times l$ rescaled Rademacher matrix. By Proposition 3.7 from (Balabanov & Nouy, 2019), $\mathbf{\Theta}$ is an $(\tau, \delta^*/r, d)$ OSE, which implies that

$$\forall \mathbf{x} \in \mathbb{R}^d, \ \frac{1}{K}\sum_{j=1}^{K} \langle \mathbf{d}^{(k,j)}, \mathbf{Z}^{(k)}\mathbf{x}\rangle^2 = (1 \pm \tau)\|\mathbf{Z}^{(k)}\mathbf{x}\|_2^2,$$

holds with probability at least $1 - \delta^*/r$. By the summation over k and the union bound argument we conclude that $\forall \mathbf{x} \in \mathbb{R}^d$,

$$\frac{1}{K}\sum_{j=1}^{K}\sum_{k=1}^{r} \langle \mathbf{d}^{(k,j)}, \mathbf{Z}^{(k)}\mathbf{x}\rangle^2 = (1 \pm \tau)\sum_{k=1}^{r} \|\mathbf{Z}^{(k)}\mathbf{x}\|_2^2, \tag{16}$$

holds with probability at least $1 - \delta^*$. By straightforward substitution of the expressions (14) and (15) into (16), and using (13), we conclude that with probability at least $1-\delta^*$,

$$\forall \mathbf{x} \in \mathbb{R}^d, \ \frac{1}{K}\|\widehat{\mathbf{W}}\mathbf{V}\mathbf{x}\|_2^2 = (1 \pm \tau)\|\mathbf{V}\mathbf{x}\|_2^2,$$

which is equivalent to the statement of the proposition. □

Proposition 5.4 presents a corollary of the Matrix Chernoff inequality from (Tropp, 2011), used to show that $\mathbf{M} = \frac{1}{\sqrt{K}}\widehat{\mathbf{W}}\mathbf{V}$ and $\sqrt{\frac{rK}{l}}\widehat{\mathbf{R}}\mathbf{M} = \mathbf{\Omega}\mathbf{V}$ have approximately equal maximal and minimal singular values.

Proposition 5.4 (Corollary of Theorem 2.2 in (Tropp, 2011))**.** *Let $\mathbf{M}$ be some $rK \times d$ matrix. Let $0 < \varepsilon^* < 1$ and $0 < \delta^* < 1$. Let $\mathbf{m}^{(j)}$ denote the rows of $\mathbf{M}$ and let $M := rK \max_{j=1,\dots,rK} \|\mathbf{m}^{(j)}\|_2^2$ and $N \geq \sigma_{min}(\mathbf{M})^{-2}$. Draw at random a sampling matrix $\widehat{\mathbf{R}}$ in (9) with*

$$l \geq 2(\varepsilon^{*2} - \varepsilon^{*3}/3)^{-1} MN \log(d/\delta^*).$$

Given $\mathcal{S}$, then with probability at least $1 - 2\delta^*$,

$$\begin{aligned}\sqrt{1-\varepsilon^*}\sigma_{min}(\mathbf{M}) &\leq \sigma_{min}(\sqrt{\frac{rK}{l}}\widehat{\mathbf{R}}\mathbf{M}) \\ &\leq \sigma_{max}(\sqrt{\frac{rK}{l}}\widehat{\mathbf{R}}\mathbf{M}) \leq \sqrt{1+\varepsilon^*}\sigma_{max}(\mathbf{M}).\end{aligned} \tag{17}$$

Proof. For any symmetric matrix $\mathbf{X}$, let $\lambda_{min}(\mathbf{X})$ and $\lambda_{max}(\mathbf{X})$ denote the minimal and the maximal eigenvalues of $\mathbf{X}$. To prove Proposition 5.4 we will use the matrix Chernoff tail bounds from (Tropp, 2011) presented in Theorem 5.5.

Define $X := \{\mathbf{m}^{(j)}(\mathbf{m}^{(j)})^{\mathrm{T}}\}_{j=1}^{n}$. Consider the matrix

$$\mathbf{X} := (\widehat{\mathbf{R}}\mathbf{M})^{\mathrm{T}}\widehat{\mathbf{R}}\mathbf{M} = \sum_{j \in T} \mathbf{m}^{(j)}(\mathbf{m}^{(j)})^{\mathrm{T}},$$

where T is a set, with $\#T = l$, of elements of $\{1, 2, \dots, rK\}$ drawn uniformly and without replacement. The matrix $\mathbf{X}$ can be written as $\mathbf{X} = \sum_{i=1}^{l} \mathbf{X}_i$, where $\{\mathbf{X}_i\}_{i=1}^{l}$ is a uniformly drawn, without replacement, random subset of X. We have $\mathbb{E}(\mathbf{X}_1) = \frac{1}{rK}\mathbf{M}^{\mathrm{T}}\mathbf{M}$. Furthermore,

$$\lambda_{max}(\mathbf{m}^{(j)}(\mathbf{m}^{(j)})^{\mathrm{T}}) = \|\mathbf{m}^{(j)}\|^2 \leq \frac{M}{rK}, \ 1 \leq j \leq rK.$$

By applying Theorem 5.5 and some algebraic operations, we obtain

$$\begin{aligned}\mathbb{P}(\lambda_{min}(\mathbf{X}) &\leq (1-\varepsilon^*)\lambda_{min}(\mathbf{M}^{\mathrm{T}}\mathbf{M})\tfrac{l}{rK}) \\ &\leq d\left(\tfrac{e^{-\varepsilon^*}}{(1-\varepsilon^*)^{1-\varepsilon^*}}\right)^{\lambda_{min}(\mathbf{M}^{\mathrm{T}}\mathbf{M})l/M} \\ &\leq d\, e^{-(\varepsilon^{*2}/2 - \varepsilon^{*3}/6)(MN)^{-1}l} \leq \delta,\end{aligned}$$

$$\begin{aligned}\mathbb{P}(\lambda_{max}(\mathbf{X}) &\geq (1+\varepsilon^*)\lambda_{max}(\mathbf{M}^{\mathrm{T}}\mathbf{M})\tfrac{l}{rK}) \\ &\leq d\left(\tfrac{e^{\varepsilon^*}}{(1+\varepsilon^*)^{1+\varepsilon^*}}\right)^{\lambda_{max}(\mathbf{M}^{\mathrm{T}}\mathbf{M})l/M} \\ &\leq d\, e^{-(\varepsilon^{*2}/2 - \varepsilon^{*3}/6)(MN)^{-1}l} \leq \delta.\end{aligned}$$

The statement of the lemma follows by a union bound argument. □

Theorem 5.5 (Matrix Chernoff tail bounds from (Tropp, 2011))**.** *Consider a finite set X of symmetric positive semi-definite matrices of size $d \times d$. Define the constant $L := \max_{\mathbf{X}_j \in X} \lambda_{max}(\mathbf{X}_j)$. Let $\{\mathbf{X}_i\}_{i=1}^{l}$ be a uniformly sampled, without replacement, random subset of X and $\mathbf{X} := \sum_{i=1}^{l} \mathbf{X}_i$. Then*

$$\mathbb{P}\left(\lambda_{min}(\mathbf{X}) \leq (1-\varepsilon)\mu_{min}\right) \leq d\left(\frac{e^{-\varepsilon}}{(1-\varepsilon)^{1-\varepsilon}}\right)^{\mu_{min}/L}$$

$$\mathbb{P}\left(\lambda_{max}(\mathbf{X}) \geq (1+\varepsilon)\mu_{max}\right) \leq d\left(\frac{e^{\varepsilon}}{(1+\varepsilon)^{1+\varepsilon}}\right)^{\mu_{max}/L}$$

where $\mu_{min} = l\lambda_{min}(\mathbb{E}\mathbf{X}_1)$ and $\mu_{max} = l\lambda_{max}(\mathbb{E}\mathbf{X}_1)$.

By plugging Proposition 5.2 and the result of Proposition 5.3 into Proposition 5.4 and taking $\mathbf{M} = \frac{1}{\sqrt{K}}\widehat{\mathbf{W}}\mathbf{V}$, $M = (\sqrt{d} + \sqrt{8\log rK/\delta^*})^2$, $N = 1.07$, along with the union bound argument, we deduce that,

$$\sqrt{1-\varepsilon^*}\sqrt{1-\varepsilon^*/30} \leq \sigma_{min}(\mathbf{\Omega V}) \leq \sigma_{max}(\mathbf{\Omega V}) \leq \sqrt{1+\varepsilon^*}\sqrt{1+\varepsilon^*/30}$$

holds with probability at least $1 - 4\delta^*$ under the condition $\mathcal{S}$. Finally by few algebraic operations, we conclude that, given $\mathcal{S}$, the singular values of $\mathbf{\Omega V}$ belong to $[\sqrt{1-\varepsilon}, \sqrt{1+\varepsilon}]$ with probability at least $4\delta/5$. The proof of the main theorem is finished by reminding that $\mathcal{S}$ occurs with probability at least $1 - \delta^*$, the union bound argument and few additional algebraic operations. □

6. Conclusion

The proposed block SRHT can combine the advantages of structured and unstructured sketching matrices, such as low application complexity and suitability for distributed computing. It should outperform all known embeddings in a distributed architecture with not too large number of processors. At the same time it yields the same approximation guarantees as standard SRHT.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 810367).

References

Ailon, N. and Chazelle, B. Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pp. 557–563, 2006.

Alaoui, A. and Mahoney, M. W. Fast randomized kernel ridge regression with statistical guarantees. *Advances in neural information processing systems*, 28, 2015.

Anaraki, F. P. and Hughes, S. M. Compressive k-svd. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5469–5473. IEEE, 2013.

Bach, F. Sharp analysis of low-rank kernel matrix approximations. In *Conference on Learning Theory*, pp. 185–209. PMLR, 2013.

Balabanov, O. and Grigori, L. Randomized Gram–Schmidt process with application to GMRES. *SIAM Journal on Scientific Computing*, 44(3):A1450–A1474, 2022.

Balabanov, O. and Nouy, A. Randomized linear algebra for model reduction. Part I: Galerkin methods and error estimation. *Advances in Computational Mathematics*, 45 (5-6):2969–3019, 2019.

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

Boutsidis, C. and Gittens, A. Improved matrix algorithms via the subsampled randomized Hadamard transform. *SIAM Journal on Matrix Analysis and Applications*, 34 (3):1301–1340, 2013.

Calandriello, D., Lazaric, A., and Valko, M. Analysis of Nyström method with sequential ridge leverage score sampling. In *Uncertainty in Artificial Intelligence*, 2016.

Charalambides, N., Mahdavifar, H., Pilanci, M., and Hero, A. O. Orthonormal sketches for secure coded regression. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pp. 826–831. IEEE, 2022.

Chiu, J. and Demanet, L. Sublinear randomized algorithms for skeleton decompositions. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1361–1383, 2013.

Demmel, J., Grigori, L., Hoemmen, M., and Langou, J. Communication-optimal parallel and sequential qr and lu factorizations. *SIAM Journal on Scientific Computing*, 34(1):A206–A239, 2012.

Derezinski, M., Khanna, R., and Mahoney, M. W. Improved guarantees and a multiple-descent curve for column subset selection and the nystrom method. *Advances in Neural Information Processing Systems*, 33: 4953–4964, 2020.

Drineas, P., Mahoney, M. W., and Cristianini, N. On the Nyström method for approximating a Gram Matrix for improved kernel-based learning. *journal of machine learning research*, 6(12), 2005.

Gittens, A. The spectral norm error of the naive nystrom extension. *arXiv preprint arXiv:1110.5305*, 2011.

Gittens, A. and Mahoney, M. Revisiting the nystrom method for improved large-scale machine learning. In *International Conference on Machine Learning*, pp. 567–575. PMLR, 2013.

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

Johnson, W. B. and Lindenstrauss, J. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26:28, 1984.

Kannan, R., Vempala, S., and Woodruff, D. Principal component analysis and higher correlations for distributed data. In *Conference on Learning Theory*, pp. 1040–1057. PMLR, 2014.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Li, H., Linderman, G. C., Szlam, A., Stanton, K. P., Kluger, Y., and Tygert, M. Algorithm 971: An implementation of a randomized algorithm for principal component analysis. *ACM Transactions on Mathematical Software (TOMS)*, 43(3):1–14, 2017.

Mahoney, M. W. et al. Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224, 2011.

Martinsson, P.-G. and Tropp, J. A. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

Meanti, G., Carratino, L., Rosasco, L., and Rudi, A. Kernel methods through the roof: handling billions of points efficiently. *Advances in Neural Information Processing Systems*, 33:14410–14422, 2020.

Nakatsukasa, Y. and Tropp, J. A. Fast & accurate randomized algorithms for linear systems and eigenvalue problems. *arXiv preprint arXiv:2111.00113*, 2021.

Pourkamali-Anaraki, F. and Becker, S. Improved fixed-rank Nyström approximation via qr decomposition: Practical and theoretical aspects. *Neurocomputing*, 363: 261–272, 2019.

Pourkamali-Anaraki, F., Becker, S., and Wakin, M. Randomized clustered nystrom for large-scale kernel machines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Rudi, A., Carratino, L., and Rosasco, L. Falkon: An optimal large scale kernel method. *Advances in neural information processing systems*, 30, 2017.

Sarlos, T. Improved approximation algorithms for large matrices via random projections. In *2006 47th annual IEEE symposium on foundations of computer science (FOCS'06)*, pp. 143–152. IEEE, 2006.

Sun, Y., Guo, Y., Luo, C., Tropp, J., and Udell, M. Low-rank Tucker approximation of a tensor from streaming data. *SIAM Journal on Mathematics of Data Science*, 2 (4):1123–1150, 2020.

Teng, D., Zhang, X., Cheng, L., and Chu, D. Least squares approximation via sparse subsampled randomized Hadamard transform. *IEEE Transactions on Big Data*, 8(2):446–457, 2020.

Tropp, J. A. Improved analysis of the subsampled randomized Hadamard transform. *Advances in Adaptive Data Analysis*, 3(01n02):115–126, 2011.

Tropp, J. A., Yurtsever, A., Udell, M., and Cevher, V. Fixed-rank approximation of a positive-semidefinite matrix from streaming data. *Advances in Neural Information Processing Systems*, 30, 2017a.

Tropp, J. A., Yurtsever, A., Udell, M., and Cevher, V. Practical sketching algorithms for low-rank matrix approximation. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1454–1485, 2017b.

Tropp, J. A., Yurtsever, A., Udell, M., and Cevher, V. Streaming low-rank matrix approximation with an application to scientific simulation. *SIAM Journal on Scientific Computing*, 41(4):A2430–A2463, 2019.

Upadhyay, J. Fast and space-optimal low-rank factorization in the streaming model with application in differential privacy. *arXiv preprint arXiv:1604.01429*, 2016.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Woodruff, D. P. et al. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.

Yang, J., Meng, X., and Mahoney, M. W. Implementing randomized matrix algorithms in parallel and distributed environments. *Proceedings of the IEEE*, 104(1):58–92, 2015.

Yin, R., Wang, W., and Meng, D. Distributed nyström kernel learning with communications. In *International Conference on Machine Learning*, pp. 12019–12028. PMLR, 2021.

Zhang, Y., Duchi, J., and Wainwright, M. Divide and conquer kernel ridge regression. In *Conference on learning theory*, pp. 592–617. PMLR, 2013.

Appendix

A. Characterization of the accuracy of Nyström approximation

We here provide a characterization of the accuracy of Nyström approximation from Section 3.

Notice the following identity (Gittens, 2011):

$$\mathbf{A} - [\![\mathbf{A}]\!]^{(\mathrm{Nyst})} = (\mathbf{A}^{\frac{1}{2}} - \mathbf{\Pi}_{\mathbf{X}}\mathbf{A}^{\frac{1}{2}})^{\mathrm{T}}(\mathbf{A}^{\frac{1}{2}} - \mathbf{\Pi}_{\mathbf{X}}\mathbf{A}^{\frac{1}{2}}),$$

where $\mathbf{\Pi}_{\mathbf{X}}$ denotes the orthogonal projector onto the range of $\mathbf{X} = \mathbf{A}^{\frac{1}{2}}\mathbf{\Omega}^{\mathrm{T}}$. Thus, to show that $[\![\mathbf{A}]\!]^{(\mathrm{Nyst})}$ approximates well $\mathbf{A}$, it suffices to show that $\mathbf{\Pi}_{\mathbf{X}}$ captures well the action of $\mathbf{A}^{\frac{1}{2}}$. In particular, by assuming that for some $d \geq k$ and $\varepsilon^* \leq \frac{1}{2}$ it holds that

$$\|\mathbf{A}^{\frac{1}{2}} - \mathbf{\Pi}_{\mathbf{X}}\mathbf{A}^{\frac{1}{2}}\|_{\mathrm{F}}^2 \leq (1+\varepsilon^*)\|\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d\|_{\mathrm{F}}^2, \tag{18}$$

we can obtain

$$\|\mathbf{A} - [\![\mathbf{A}]\!]^{(\mathrm{Nyst})}\|_* = \|\mathbf{A}^{\frac{1}{2}} - \mathbf{\Pi}_{\mathbf{X}}\mathbf{A}^{\frac{1}{2}}\|_{\mathrm{F}}^2 \leq (1+\varepsilon^*)\|\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d\|_{\mathrm{F}}^2 = (1+\varepsilon^*)\|\mathbf{A} - [\![\mathbf{A}]\!]_d\|_*. \tag{19}$$

It is then noticed that (18) also implies the accuracy of the truncated approximation $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}$ due to the following consequence of the triangle inequality (see for instance Proposition A.6 in (Tropp et al., 2017b)):

$$\|\mathbf{A} - [\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}\|_\xi \leq \|\mathbf{A} - [\![\mathbf{A}]\!]_k\|_\xi + 2\|\mathbf{A} - [\![\mathbf{A}]\!]^{(\mathrm{Nyst})}\|_\xi, \tag{20}$$

where $\xi = 2$ or $*$, so that we have by (19),

$$\|\mathbf{A} - [\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}\|_\xi \leq \|\mathbf{A} - [\![\mathbf{A}]\!]_k\|_\xi + 3\|\mathbf{A} - [\![\mathbf{A}]\!]_d\|_*. \tag{21}$$

This result guarantees, under the condition (18), the quasi-optimality of $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}$ with respect to the trace norm. Furthermore, if $\mathbf{A}$ has a fast enough singular value decay and d is large enough, so that the tail after d-th singular value of $\mathbf{A}$, i.e. $\sum_{i=d+1}^n \sigma_i = \|\mathbf{A} - [\![\mathbf{A}]\!]_d\|_*$ is small compared to the $k+1$-th singular value $\sigma_{k+1} = \|\mathbf{A} - [\![\mathbf{A}]\!]_k\|_2$ then $[\![\mathbf{A}]\!]_k^{(\mathrm{Nyst})}$ is almost as accurate as $[\![\mathbf{A}]\!]_k$ with respect to both the trace norm and the spectral norm.

It remains to obtain the conditions on $\mathbf{\Omega}$ such that (18) holds with high probability. This can be done for instance with the results from (Woodruff et al., 2014). Take $\varepsilon = \frac{4}{9}\varepsilon^*$. It follows from Lemma 45 in (Woodruff et al., 2014) and its proof that (18) holds with probability at least $1-\delta$ if $\mathbf{\Omega}$ is an $(\frac{1}{3}, \delta, d)$ OSE, and

$$\|\mathbf{U}_d^{\mathrm{T}}\mathbf{\Omega}^{\mathrm{T}}\mathbf{\Omega}(\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d)\|_{\mathrm{F}}^2 \leq \varepsilon\|\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d\|_{\mathrm{F}}^2.$$

In turn the latter condition is satisfied with probability at least $1-\delta$ if $\mathbf{\Omega}$ is an $(\sqrt{\frac{\varepsilon}{d}}, \frac{\delta}{N}, 1)$ OSE, where $N = 2nd + n + d$, as shown below. The OSE property of $\mathbf{\Omega}$ and the union bound argument guarantee that for given N fixed vectors $\mathbf{z}_i$, we have

$$(1 - \sqrt{\tfrac{\varepsilon}{d}})\|\mathbf{z}_i\|_2^2 \leq \|\mathbf{\Omega}\mathbf{z}_i\|_2^2 \leq (1 - \sqrt{\tfrac{\varepsilon}{d}})\|\mathbf{z}_i\|_2^2, \text{ for } 1 \leq i \leq N \tag{22}$$

with probability at least $1-\delta$. Take set $\{\mathbf{z}_i\}$ composed of the normalized columns of $\mathbf{U}_d$ denoted by $\mathbf{x}_i$, the normalized columns of $\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d$ denoted by $\mathbf{y}_i$, and all the pairs $\mathbf{x}_i + \mathbf{y}_j$ and $\mathbf{x}_i - \mathbf{y}_j$. Then the relation (22), the parallelogram identity and the fact that $\mathbf{x}_i^{\mathrm{T}}\mathbf{y}_j = 0$, imply that

$$|\mathbf{x}_i^{\mathrm{T}}\mathbf{\Omega}^{\mathrm{T}}\mathbf{\Omega}\mathbf{y}_j| \leq \sqrt{\tfrac{\varepsilon}{d}}\|\mathbf{x}_i\|_2\|\mathbf{y}_j\|_2, \text{ for } 1 \leq i \leq d,\ 1 \leq j \leq n.$$

Consequently, we have

$$\|\mathbf{U}_d^{\mathrm{T}}\mathbf{\Omega}^{\mathrm{T}}\mathbf{\Omega}(\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d)\|_{\mathrm{F}}^2 = \sum_{i=1}^d\sum_{j=1}^n |\mathbf{x}_i^{\mathrm{T}}\mathbf{\Omega}^{\mathrm{T}}\mathbf{\Omega}\mathbf{y}_j|^2 \leq \tfrac{\varepsilon}{d}\sum_{i=1}^d\sum_{j=1}^n \|\mathbf{x}_i\|_2^2\|\mathbf{y}_j\|_2^2 = \tfrac{\varepsilon}{d}\|\mathbf{U}_d\|_{\mathrm{F}}^2\|\mathbf{A}^{\frac{1}{2}} - [\![\mathbf{A}^{\frac{1}{2}}]\!]_d\|_{\mathrm{F}}^2 \tag{23}$$

with probability at least $1-\delta$. The proof is finished by noting that $\|\mathbf{U}_d\|_{\mathrm{F}}^2 = d$.

It is concluded that (18) and as a consequence (21) are satisfied with probability at least $1-2\delta$ if $\mathbf{\Omega}$ is an $(\frac{1}{3}, \delta, d)$ OSE and $(\sqrt{\frac{\varepsilon}{d}}, \frac{\delta}{N}, 1)$ OSE. In turn, according to Theorem 2.1 and (3), this condition is satisfied by the block as well as the standard SRHT with $l = \mathcal{O}(d \log^2 \frac{n}{\delta})$ rows (taking $\varepsilon^* = \frac{1}{2}$, $\varepsilon = \frac{2}{9}$). Whereas for Gaussian matrices the required number of rows to satisfy the aforementioned OSEs properties is somewhat lower: $l = \mathcal{O}(d \log \frac{n}{\delta})$. Although it has to be said that SRHT matrices in practice give similar results as Gaussian matrices (Halko et al., 2011). As can be seen from our experiments, this should also be the case for block SRHT. Moreover, we note that the condition $l = \mathcal{O}(d \log \frac{n}{\delta})$ for Gaussian matrices is still pessimistic. This overestimation is an artifact due to the use of a general analysis based solely on the OSE property. In reality, a Gaussian $\mathbf{\Omega}$ should satisfy (18) with high probability if it has size $l = \mathcal{O}(d)$ with a small constant (say 2 or 4) (Halko et al., 2011; Tropp et al., 2017a).