18.3368 PROJECT: LARGEST SINGULAR VALUES OF BI-DIAGONAL MATRICES IN JULIA *

EVELYNE RINGOOT†

Abstract. Approximate singular value decomposition methods continue to gain importance as full SVD calculations become too computationally and energetically expensive for extremely large data sets. We explore the accuracy of approximating the SVD of a full matrix by the SVD of its upper left corner for bidiagonal matrices and link our insights to recent developments in randomized SVD algorithms.

Key words. Singular Value Decomposition, Random Singular Value Decomposition, Bidiagonal Matrix, Tridiagonal Matrix, Random Matrices, Julia language

1. Introduction.

As the amount of available data in the world has grown exponentially over the past decade, [19] so have the computational requirements on matrices. Extremely large data is for example found in data mining[17, 32]. Nowadays, this data has the size of gigabytes or terabytes [16] while typical consumer GPUs contain around 8 GB [34] and supercomputer GPUs around 80 GB [1].[35] While CPUs typically have larger memory, they lack the parallelization possibilities of GPUs that allow for fast calculations.[7] As this trend of increasing data availability persists, the question arises whether to keep increasing the hardware capacity, at the expense of immense power consumption, [4, 18, 36, 25, 13, 47] or rather invest in more computationally efficient methods to treat such extremely large data sets as has been argued by [37]. This report is one step towards the latter.

The Singular Value Decomposition is of particular interest for large data due to its wide range of applications, the most well-known one being machine learning.[27] However, it is also widely applied in image processing, [8, 22, 39] quantum information theory, [31, 28] and general data feature analysis [9]. More broadly, the SVD is of importance in any field that uses principal component analysis, matrix rank estimation, matrix inverse, or least squares calculation.[24, 29] For many of these applications, the largest singular value or the N-th largest singular value is the most valuable information (with N being significantly smaller than the matrix size).[23, 1, 26] The intent of this report is to verify the accuracy of estimating the N-th largest singular value as the N-th largest singular value of a truncated version of the matrix for certain types of matrices, in particular for the bidiagonal matrices, the lower bidiagonal form of a Gaussian matrix.[12]. It has already been found that there is no evidence to the contrary for tridiagonal matrices,[20] the tridiagonal reduction of the symmetric part of a random matrix. [12] In the first part of this report in section 2, we will assess the accuracy obtained by calculating the SVD of truncated bidiagonal matrices.[2] We will run Monte-Carlo simulations for smaller matrices, attempt to find a pattern, and verify whether this pattern holds for larger matrices to obtain out-of-sample accuracy. We will also verify the distribution of the largest singular values in section 3.

More generally, this type of procedure has been previously applied in randomized SVD algorithms, [45, 17, 2, 22, 33, 27, 15] where the largest SVD values of any matrix have been approximated by the SVD values of a subset of its rows, columns, or both.

*Submitted October 31, 2022

†Massachusetts Institute of Technology, Cambridge MA, USA (eringoot@mit.edu).

[1] For supercomputing applications, multiple such GPUs are typically connected together.

[2] Not to be confused with the truncated SVD, where a lower-rank approximation is made considering the full matrix.

For physical problems, this is equivalent to considering the most important axes or components of a system and neglecting the minor elements or axes of such a problem, a common strategy in engineering problems. Mathematically, we approximate a near-sparse matrix with most weight in a block by a sparse matrix. The known results for tridiagonal matrices suggest that the accuracy for specific types of matrices might be significantly greater than for general matrices, and consequently that a faster algorithm might be possible. There has been extensive research into the accuracy and algorithmic speed of randomized truncated SVD algorithms on general matrices. [3, 40, 11, 42, 21, 44]. The objective of this paper is not to devise such new algorithms for specific types of matrices, but rather to point out the interest of the question and conduct a computational exploration to guide future theoretical research. For this purpose, a brief literature review on randomized SVD algorithms will be conducted and the idea behind these algorithms will be tested on a real-world matrix in section 4.

2. Accuracy for bidiagonal matrices.

2.1. Background and methods. [12] provides the necessary background for this section of the report, and defines tridiagonal matrices as follows. Consider $A_{(i,j)} = G(0,1)$ a matrix of Gaussians. $A_{(i,j)}$ can then be reduced to the bidiagonal matrix $B(n,m)_{(i,j)}$ of size $(n+1)\times n$ with $n > m$ through Householder transformations.

$$B(n,m) = \begin{pmatrix} \chi_n & & & \\ \chi_m & \chi_{n-1} & & \\ & \ddots & \ddots & \\ & & \chi_2 & \chi_{n-m+1} \\ & & & \chi_1 \end{pmatrix}$$

We wish to validate that the k-th singular value of $B(n,m)$ can be well aproximated by the k-th singular value of the upper left block of the matrix, which we will call $B(n,m)\{l,l\}$ for the upper block of the first l rows and l columns.

$$B(n,m)\{l,l\} = \begin{pmatrix} \chi_n & & & \\ \chi_m & \chi_{n-1} & & \\ & \ddots & \ddots & \\ & & \chi_l & \chi_{n-m+1+l} \end{pmatrix}$$

For this purpose, we run a series of Monte-Carlo simulations for various values of k, m, n, and $p = l/n$. We then estimate the error in function of these parameters and validate it out-of-sample. We also review the distribution of singular values over the different trials of the Monte-Carlo simulation: as $B(n,m)\{l,l\}$ is a part of $B(n,m)$, the singular values will be a lower bound.

2.2. Results. For $k = 1$ and $m = n$, figure 4 shows the average error over Monte-Carlo trials. The portion p is displayed as a fraction of m. For matrix size $m = n = 1$ the bidiagonal reduction is actually of size $(2,1)$ and $p = 1$ thus refers to taking the upper left $(1,1)$, resulting in an error higher than zero. As soon as this effect is mitigated, around $m = n = 10$, and the discarding of the last row is proportionally less significant, a dark purple area of error below $1e-6$ designates the area with almost perfect accuracy. For values of n higher than 100, numerical errors come into play and the accuracy is lower than in $10 < n < 100$. In the areas of slightly higher errors, for error values above 0.01%, the logarithmic-logarithmic relationship is mostly linear: at higher n we need a lower portion p of the bidiagonal reduction to accurately estimate the largest singular value. Using an error of 0.01% as the cut-off value, we can approximate the required portion p of the matrix in function of n by a

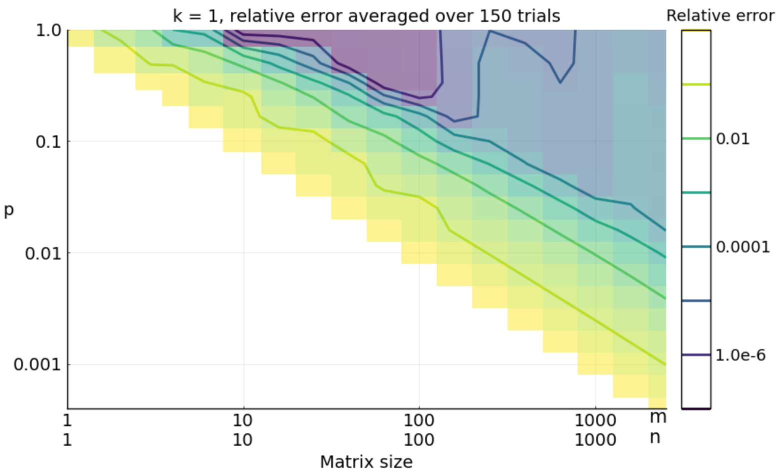

Fig. 1: Relative error of approximation of the first singular value of the bidiagonal reduction of a random matrix of size $n = m$ by using a portion p of the rows and columns, averaged over 150 trials of Monte-Carlo simulations.

logistic regression and obtain that:

$$p \approx 6.4n^{(-4/5)} \tag{2.1}$$

Out-of-sample accuracy was verified on a $m = n = 10^4$ matrix over 20 trials and confirmed.

To understand the influence of changing the values of k, of the m/n ratio and of the average over maximum error over different trials, we conduct a parameter sensitivity study in Figure 2 and Figure 3. We can draw the following conclusions:

- *Low sensitivity to k.* On figure 2, we observe that changing the value of k barely has any effect on the result for the higher error levels. The values of k we examined are significantly smaller values of m and n that are examined. As such, we would expect the accuracy to go down slightly with increasing k as the lower elements do influence the lower singular values more than the highest, but since we are keeping k a lot higher than m and n this effect is negligible. The logarithmic - logarithmic relationship we observed earlier thus applies to larger values of k as well as long as k stays well below m.
- *Accuracy slightly decreases with increasing ratio n/m.* As this ratio goes up, the relative difference between the diagonal and subdiagonal increases. We observe in figure 2 that the relative error for very large m increases around the area where the error is smaller than 10^{-4}, while there is barely any effect on the linear relationships for smaller errors. The larger difference between the two diagonals possibly explains the numerical inaccuracies observed.
- *No longer linear relationship between n and p for error line for maximum error value over different trials.* Everything discussed so far considered the average value over Monte-Carlo trials in figure 2. We will now compare this with the maximum values of the error over the trials from figure 3. We observe

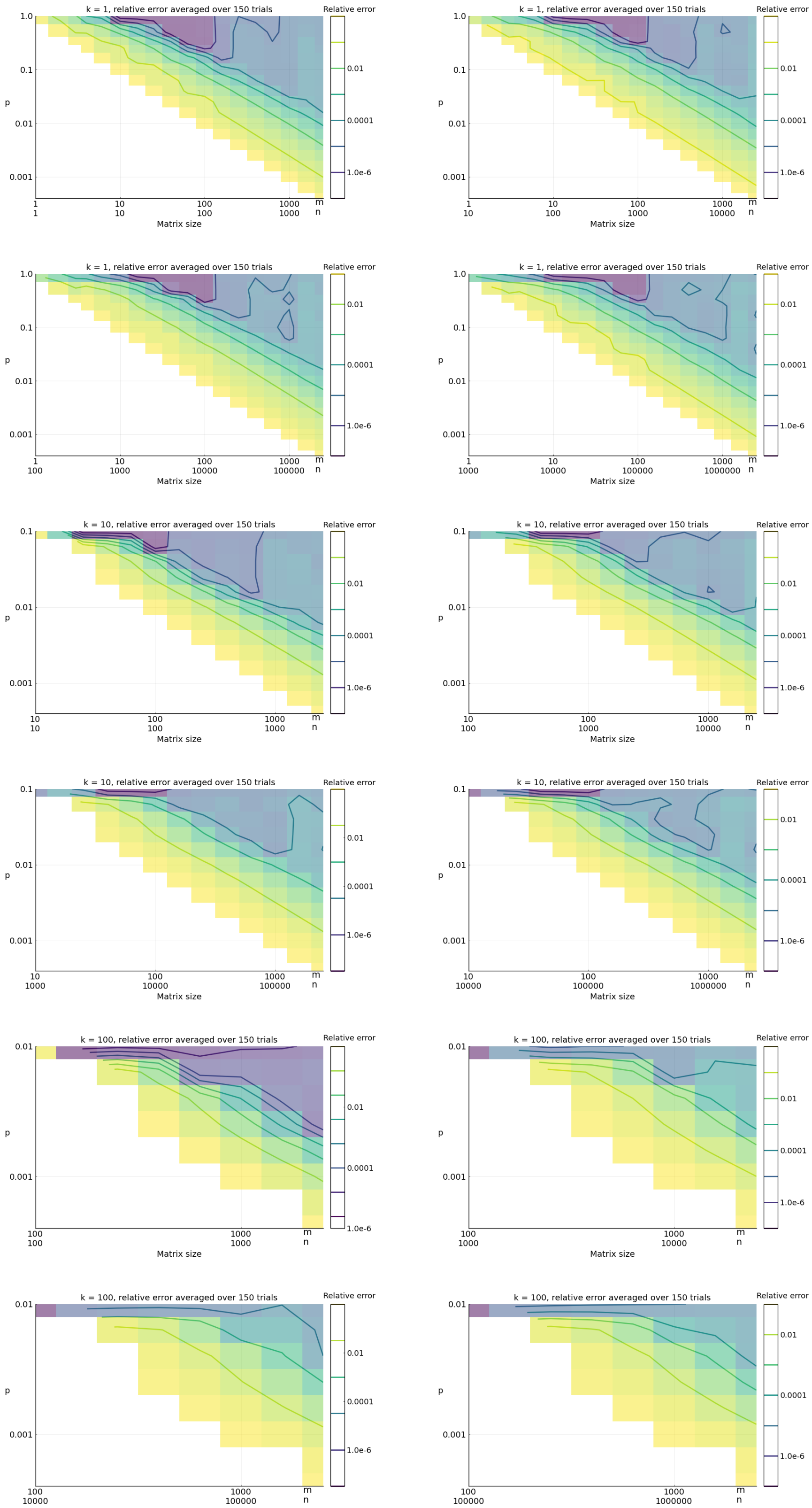

Fig. 2: Relative error of approximation of k-th singular value of the bidiagonal reduction of a random matrix of size $n \times m$ by using a portion p of the rows and columns, averaged over 150 trials of Monte-Carlo simulations.

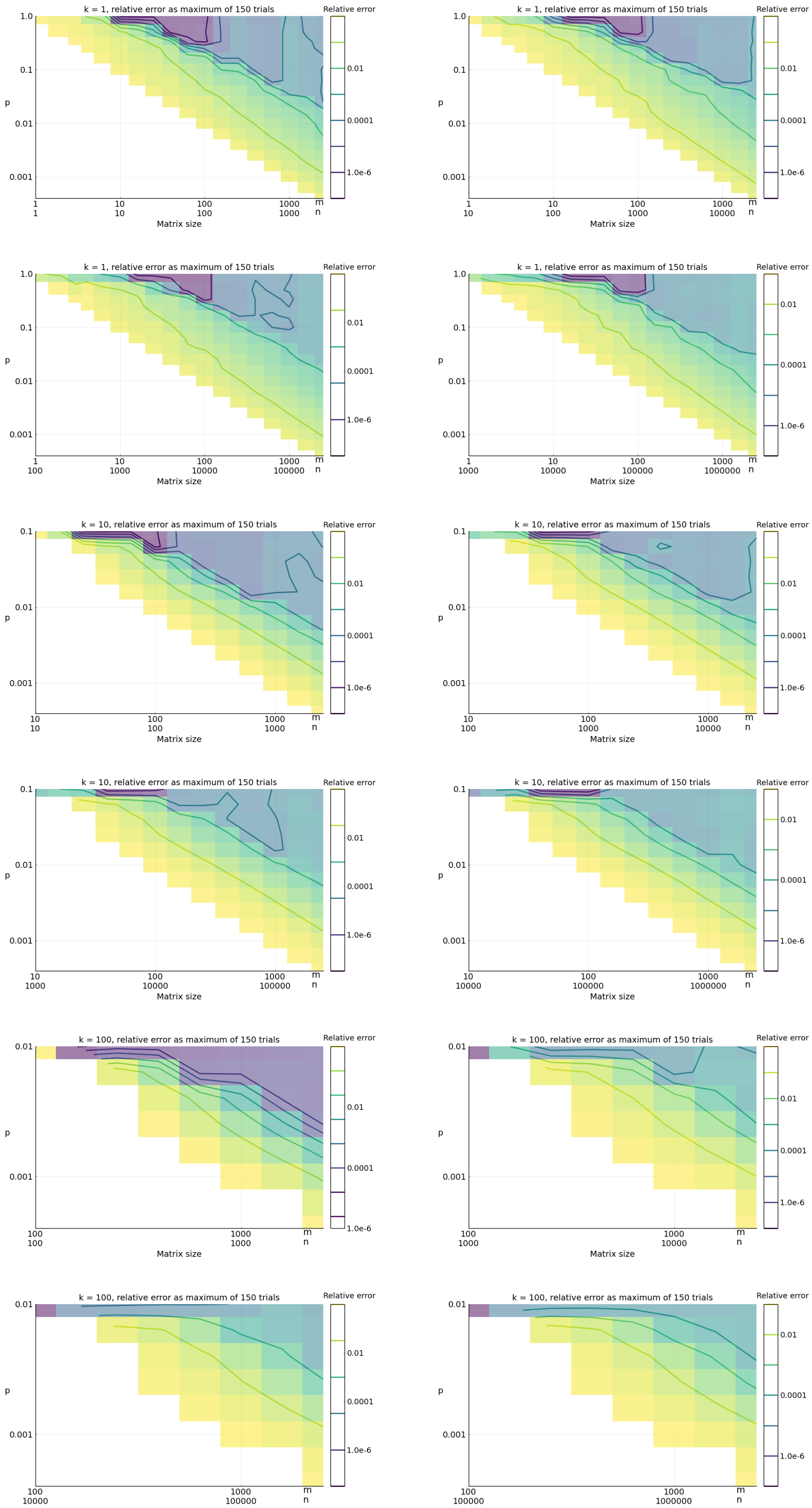

Fig. 3: Relative error of approximation of k-th singular value of the bidiagonal reduction of a random matrix of size $n \times m$ by using a portion p of the rows and columns, as the maximum of 150 trials of Monte-Carlo simulations.

the linear relationships from figure 2 are not as clearly linear anymore on figure 3. The singular value estimate obtained by considering a truncated part of the matrix is always a lower bound. The average over many Monte-Carlo trials will end up close to the exact values. However, the upper bound is typically a less good approximation for the exact values than the lower bound for this matrix truncation scenario, as such the dependence of the relative error on n and p is more complicated here. This estimation in relationship 2.1 method is thus inappropriate to use to obtain a 'certain' estimate for one specific matrix. Rather, it will estimate values well where multiple matrices need to be considered and averaged.

3. Limit distribution of largest singular values. The Tracy-Widom distribution describes the distribution of the largest eigenvalues of random symmetric matrices. Interestingly, it also describes a range of other distributions, including that of the square of the largest singular values of the bidiagonal matrices, or the Laguerre ensemble. The scaling is described by the following soft-edge limit [12]:

$$E^{soft}(0; J) = \lim_{n\to\infty} = E^{(n)}_{LUE}(0; 4n + 2(2n)^{(}1/3)J; \alpha) \tag{3.1}$$

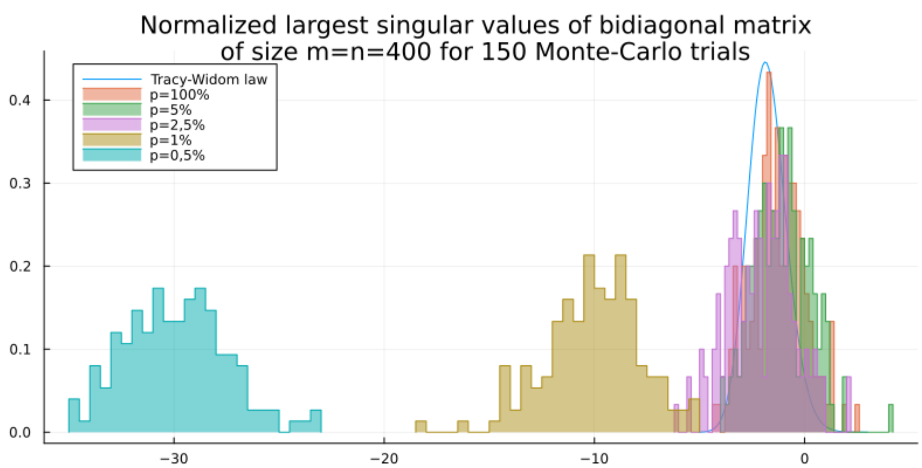

Fig. 4: Normalized largest singular value distribution of bidiagonal reductions of Gaussian matrices in function of the portion of rows and columns considered p. We observe that for $p = 5\%$ which is in correspondence with equation 2.1 there is very good correspondence between the Tracy-Widom law and the singular value distribution. As p increases, there is an offset and it is clear the singular values calculated from the truncated matrix version are no longer approximately accurate.

This is of particular interest to us, as it allows us to investigate whether the equation 2.1 holds for very large n, which would otherwise be impossible to compute and verify. In Figure 4 this distribution is shown for various truncated versions of the bidiagonal matrix. According to formula 2.1, we would need $p = 5\%$. We can indeed confirm that the distribution for $p = 5\%$ is very similar to the distribution for $p = 100\%$. As p decreases further, the distribution changes: this indeed confirms that lower values of p no longer result in accurate estimates for the largest svd values.

4. Impact on random matrix algorithms. The previous section provides an example of a type of matrix in which the singular values can be well-approximated by the singular values of the upper left corner of it. This is extendable to general matrices, and is exactly the principle on which randomized singular value decomposition is based. For low-rank approximations of matrices, where one wishes to obtain only the k first singular values, a certain subset of rows and columns is selected with a defined probability. The question of how to achieve the computationally least expensive and most accurate sampling technique has been the topic of extensive research.

- *Based on Frobenius norm or spectral norm.* Subsets of rows and columns are selected based on their Frobenius norm or spectral norm: the rows and columns with larger values have more probability of being selected. [15, 6, 5]
- *Adaptive sampling.* After a first subset of rows and columns is selected, subsequent ones are also determined by their distance to the selected first ones. [10]
- *Volume-based sampling.* The Frobenius norm of combinations of rows or combinations of columns is considered to determine the probability of selection. [10]
- *Using random projections.* Gaussian-type or Hadamard-type projections of the matrix of a lower rank are used to select a subspace of the matrix. [41, 30, 46]

Earlier work also includes theoretical formulas for the bounds on the error of the approximate SVD.[41] In recent years, interest in random algorithms has taken up as the methods lend themselves well to GPU-based programming, due to their reduced memory requirements and independent subcomputations. [14] Very recently, the main commercial GPU package CUDA has also included a randomized singular value decomposition algorithm. [43]

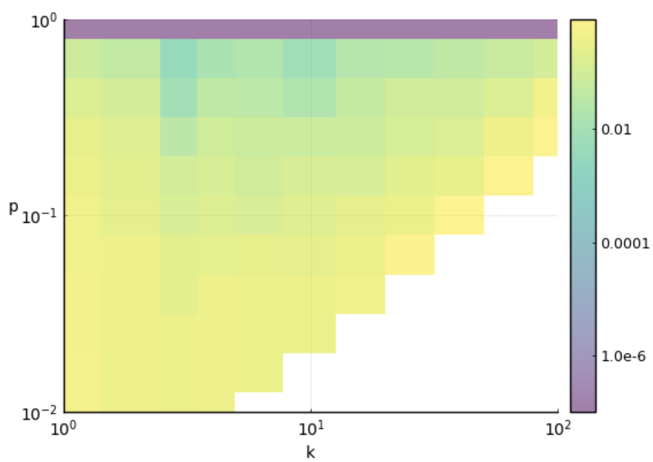

Fig. 5: Relative error of approximation of k-th singular value of economic data from the 1970s [38] of size $n = m \approx 500$ by using a portion p of the rows and columns, those with the highest norm. We observe here acceptable accuracy as well, even with low values of p.

4.1. Demonstration. To understand the accuracy of these random SVD methods in practice, we will investigate one real-world matrix example, containing US economic data from the 1970s [38]. We will not implement a full RSVD algorithm but rather extend the methodology we have applied to bidiagonal matrices to general matrices. We select the columns and rows then based on the Frobenius norm, as was the case in some of the older RSVD algorithms, but neglect the random part and simply select those columns and rows with the largest norm. The results of this test are displayed in figure 5. We observe that an accuracy of 1 - 10% can easily be obtained even at low values of p, which demonstrates it is reasonable to approximate singular values of large matrices as those of one of the subspaces. In addition, we have neglected here the random part of the selection. This random part links back to the bidiagonals reductions we discussed earlier. As random other columns will be selected, and we average over a set of 'trials', we can expect to obtain higher accuracy due to the random component.

5. Conclusion. We have explored the accuracy of approximating the k- th largest singular value of the bidiagonal reduction of a random matrix by a subset of its rows and columns. We have developed and validated an approximate relationship that determines how much of the rows and columns is required to have a relative error lower than $0,01\%$, validated the relationship out-of-sample, and investigated its sensitivity to the value of k and the ratio m/n. We have also shown the limit distribution of the square of the largest singular values approaches the Tracy-Widom law. Finally, we have linked this selection of rows and columns and low-rank approximation to randomized algorithms, and explored these algorithms with a real matrix example.

Acknowledgments. The contribution of Dr. Sung Woo Jeong and professor Alan Edelman was crucial to the fulfillment of this project. The author was supported by a Fellowship of the Belgian American Educational Foundation.

REFERENCES

[1] S. Ahmadi-Asl, S. Abukhovich, M. G. Asante-Mensah, A. Cichocki, A. H. Phan, T. Tanaka, and I. Oseledets, *Randomized algorithms for computation of tucker decomposition and higher order svd (hosvd)*, IEEE Access, 9 (2021), pp. 28684–28706, https://doi.org/10.1109/ACCESS.2021.3058103.

[2] K. Batselier, W. Yu, L. Daniel, and N. Wong, *Computing low-rank approximations of large-scale matrices with the tensor network randomized svd*, SIAM Journal on Matrix Analysis and Applications, 39 (2018), pp. 1221–1244, https://doi.org/10.1137/17M1140480.

[3] N. Benjamin Erichson, S. L. Brunton, and J. Nathan Kutz, *Compressed singular value decomposition for image and video processing*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops, Oct 2017.

[4] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano, and L. Benini, *Predictive modeling for job power consumption in hpc systems*, in High Performance Computing, J. M. Kunkel, P. Balaji, and J. Dongarra, eds., Cham, 2016, Springer International Publishing, pp. 181–199.

[5] C. Boutsidis, P. Drineas, and M. Magdon-Ismail, *Near-optimal column-based matrix reconstruction*, SIAM Journal on Computing, 43 (2014), pp. 687–717, https://doi.org/10.1137/12086755X.

[6] C. Boutsidis, M. W. Mahoney, and P. Drineas, *An Improved Approximation Algorithm for the Column Subset Selection Problem*, pp. 968–977, https://doi.org/10.1137/1.9781611973068.105.

[7] B. CaulField, *What's the difference between a cpu and a gpu?* online, 2009, https://blogs.nvidia.com/blog/2009/12/16/whats-the-difference-between-a-cpu-and-a-gpu/ (accessed 2022-10-30).

[8] Z. Chen, *Singular value decomposition and its applications in image processing*, in Proceedings of the 2018 1st International Conference on Mathematics and Statistics, ICoMS '18, New York, NY, USA, 2018, Association for Computing Machinery, p. 16–22, https://doi.org/10.1145/3274250.3274261.

[9] K. Coughlin and K. K. Tung, *EMPIRICAL MODE DECOMPOSITION AND CLIMATE VARIABILITY*, pp. 149–165, https://doi.org/10.1142/9789812703347_0007.

[10] A. Deshpande, L. Rademacher, S. S. Vempala, and G. Wang, *Matrix approximation and projective clustering via volume sampling*, Theory of Computing, 2 (2006), pp. 225–247, https://doi.org/10.4086/toc.2006.v002a012, https://theoryofcomputing.org/articles/v002a012.

[11] P. D. E. Drinea and P. Huggins, *A randomized singular value decomposition algorithm for image processing applications*, Proc. 8th Panhellenic Conf. Informatics, (2001), pp. 278–288.

[12] A. Edelman, *Random matrix theory.* Cambridge, MA, USA, draft version per personal communciation, September 2022.

[13] W.-c. Feng and K. Cameron, *The green500 list: Encouraging sustainable supercomputing*, Computer, 40 (2007), pp. 50–55, https://doi.org/10.1109/MC.2007.445.

[14] B. Foster, S. Mahadevan, and R. Wang, *A gpu-based approximate svd algorithm*, in Parallel Processing and Applied Mathematics, R. Wyrzykowski, J. Dongarra, K. Karczewski, and J. Waśniewski, eds., Berlin, Heidelberg, 2012, Springer Berlin Heidelberg, pp. 569–578.

[15] A. Frieze, R. Kannan, and S. Vempala, *Fast monte-carlo algorithms for finding low-rank approximations*, J. ACM, 51 (2004), p. 1025–1041, https://doi.org/10.1145/1039488.1039494.

[16] Google, *The size and quality of a data set.* Google Developers Foundational courses in Machine Learning, July 2022, https://developers.google.com/machine-learning/data-prep/construct/collect/data-size-quality (accessed 2022-10-30).

[17] N. Halko, P. G. Martinsson, and J. A. Tropp, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Review, 53 (2011), pp. 217–288, https://doi.org/10.1137/090771806.

[18] M. Hennecke, W. Frings, W. Homberg, A. Zitz, M. Knobloch, and H. Böttiger, *Measuring power consumption on IBM blue gene/p*, 27, pp. 329–336, https://doi.org/10.1007/s00450-011-0192-y.

[19] S. IDC, Seagate, *Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2020, with forecasts from 2021 to 2025.* Statista chart, Sept. 2022, https://www.statista.com/statistics/871513/worldwide-data-created/ (accessed 2022-10-19).

[20] S. W. Jeong. personal communication, Oct. 2022.

[21] H. Ji, W. Yu, and Y. Li, *A rank revealing randomized singular value decomposition (r3svd) algorithm for low-rank matrix approximations*, 2016, https://doi.org/10.48550/ARXIV.1605.08134.

[22] L. Y. Ji H, *Gpu acceleralated randomized singular value decomposition and its application in image compression*, 2014.

[23] Z. Jia and Y. Yang, *Modified truncated randomized singular value decomposition (mtrsvd) algorithms for large scale discrete ill-posed problems with general-form regularization*, Inverse Problems, 34 (2018), p. 055013, https://doi.org/10.1088/1361-6420/aab92d.

[24] D. Kalman, *A singularly valuable decomposition: The svd of a matrix*, The College Mathematics Journal, 27 (1996), pp. 2–23, https://doi.org/10.1080/07468342.1996.11973744.

[25] Y. Kodama, T. Odajima, E. Arima, and M. Sato, *Evaluation of power management control on the supercomputer fugaku*, in 2020 IEEE International Conference on Cluster Computing (CLUSTER), 2020, pp. 484–493, https://doi.org/10.1109/CLUSTER49012.2020.00069.

[26] N. K. Kumar and J. Schneider, *Literature survey on low rank approximation of matrices*, Linear and Multilinear Algebra, 65 (2017), pp. 2212–2244, https://doi.org/10.1080/03081087.2016.1267104.

[27] M. W. Mahoney, *Randomized algorithms for matrices and data*, Foundations and Trends in Machine Learning, 3 (2011), pp. 123–224, https://doi.org/10.1561/2200000035.

[28] C. D. Martin and M. A. Porter, *The extraordinary svd*, The American Mathematical Monthly, 119 (2012), pp. 838–851, https://doi.org/10.4169/amer.math.monthly.119.10.838.

[29] C. D. Martin and M. A. Porter, *The extraordinary svd*, The American Mathematical Monthly, 119 (2012), pp. 838–851, https://doi.org/10.4169/amer.math.monthly.119.10.838.

[30] V. Menon, Q. Du, and J. E. Fowler, *Fast svd with random hadamard projection for hyperspectral dimensionality reduction*, IEEE Geoscience and Remote Sensing Letters, 13 (2016), pp. 1275–1279, https://doi.org/10.1109/LGRS.2016.2581172.

[31] J. A. MISZCZAK, *Singular value decomposition and matrix reorderings in quantum in-*

formation theory, International Journal of Modern Physics C, 22 (2011), pp. 897–918, https://doi.org/10.1142/S0129183111016683.

[32] C. Musco and C. Musco, *Randomized block krylov methods for stronger and faster approximate singular value decomposition*, in Advances in Neural Information Processing Systems, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds., vol. 28, Curran Associates, Inc.

[33] C. Musco and C. Musco, *Randomized block krylov methods for stronger and faster approximate singular value decomposition*, in Advances in Neural Information Processing Systems, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds., vol. 28, Curran Associates, Inc.

[34] Nvidia, *Compare geforce graphics cards*. online, 2022, https://www.nvidia.com/en-us/geforce/graphics-cards/compare/ (accessed 2022-10-30).

[35] Nvidia, *Nvidia a100 tensor core gpu*. online, 2022, https://www.nvidia.com/en-us/data-center/a100/ (accessed 2022-10-30).

[36] S. Pakin, C. Storlie, M. Lang, R. E. Fields III, E. E. Romero Jr., C. Idler, S. Michalak, H. Greenberg, J. Loncaric, R. Rheinheimer, G. Grider, and J. Wendelberger, *Power usage of production supercomputers and production workloads*, Concurrency and Computation: Practice and Experience, 28 (2016), pp. 274–290, https://doi.org/https://doi.org/10.1002/cpe.3191.

[37] T. Palmer, *Modelling: Build imprecise supercomputers*, 526, pp. 32–33, https://doi.org/10.1038/526032a, https://doi.org/10.1038/526032a.

[38] K. R. B. M. R. Boisvert, R. Pozo and R. Lipman, *Mbeacxc: Economic models us economic transactions in 1972 – commodities x commodities*, https://math.nist.gov/MatrixMarket/data/Harwell-Boeing/econiea/mbeacxc.html#set (accessed 2022/15/12).

[39] R. A. Sadek, *Svd based image processing applications: State of the art, contributions and research challenges*, International Journal of Advanced Computer Science and Applications, 3 (2012).

[40] A. K. Saibaba, J. Hart, and B. van Bloemen Waanders, *Randomized algorithms for generalized singular value decomposition with application to sensitivity analysis*, Numerical Linear Algebra with Applications, 28 (2021), p. e2364, https://doi.org/https://doi.org/10.1002/nla.2364.

[41] T. Sarlos, *Improved approximation algorithms for large matrices via random projections*, in 2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06), 2006, pp. 143–152, https://doi.org/10.1109/FOCS.2006.37.

[42] P. Song, J. D. Trzasko, A. Manduca, B. Qiang, R. Kadirvel, D. F. Kallmes, and S. Chen, *Accelerated singular value-based ultrasound blood flow clutter filtering with randomized singular value decomposition and randomized spatial downsampling*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 64 (2017), pp. 706–716, https://doi.org/10.1109/TUFFC.2017.2665342.

[43] Struski, P. Morkisz, P. Spurek, S. R. Bernabeu, and T. Trzciński, *Efficient gpu implementation of randomized svd and its applications*, 2021, https://doi.org/10.48550/ARXIV.2110.03423.

[44] S. Vatankhah, S. Liu, R. A. Renaut, X. Hu, and J. Baniamerian, *Improving the use of the randomized singular value decomposition for the inversion of gravity and magnetic data*, GEOPHYSICS, 85 (2020), pp. G93–G107, https://doi.org/10.1190/geo2019-0603.1.

[45] S. Voronin and P.-G. Martinsson, *Rsvdpack: An implementation of randomized algorithms for computing the singular value, interpolative, and cur decompositions of matrices on multi-core and gpu architectures*, 2015, https://doi.org/10.48550/ARXIV.1502.05366.

[46] F. Woolfe, E. Liberty, V. Rokhlin, and M. Tygert, *A fast randomized algorithm for the approximation of matrices*, Applied and Computational Harmonic Analysis, 25 (2008), pp. 335–366, https://doi.org/https://doi.org/10.1016/j.acha.2007.12.002.

[47] X. Xie, X. Fang, S. Hu, and D. Wu, *Evolution of supercomputers*, 4, pp. 428–436, https://doi.org/10.1007/s11704-010-0118-z.