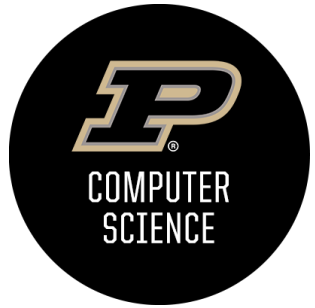
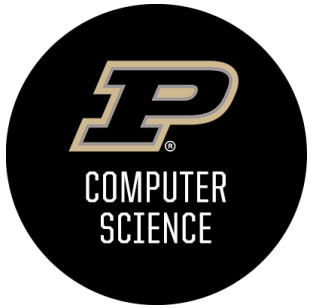
(RAND)NLA: (RANDOMIZED) NUMERICAL LINEAR ALGEBRA

Petros Drineas



OVERVIEW

- A historical viewpoint
- **RandNLA**: **Rand**omized **N**umerical **L**inear **A**lgebra
- From **RandNLA** to **Stochastic Rounding**



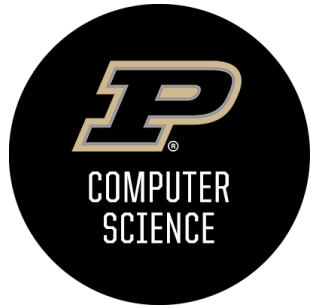
NUMERICAL ANALYSIS

Let's start with Numerical Analysis, a.k.a, Scientific Computing

My favorite definition of Numerical Analysis:

The study of approximate solutions to mathematical problems, taking into account the extent of possible errors.

American Heritage Dictionary, 1992



NUMERICAL ANALYSIS

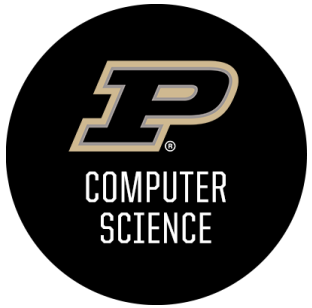
Let's start with Numerical Analysis, a.k.a, Scientific Computing

My favorite definition of Numerical Analysis:

The study of approximate solutions to mathematical problems, taking into account the extent of possible errors.

American Heritage Dictionary, 1992

Numerical analysis was motivated by the development of the earliest computers, to solve problems in ballistics, PDE's, early data analysis, etc.

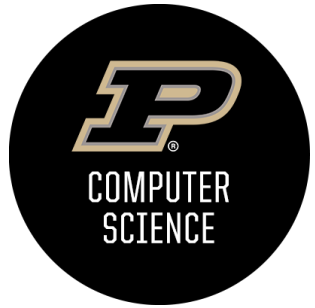


NUMERICAL LINEAR ALGEBRA

From Numerical Analysis to Numerical Linear Algebra (NLA)

Numerical Linear Algebra is Numerical Analysis focused on algorithms involving matrices.

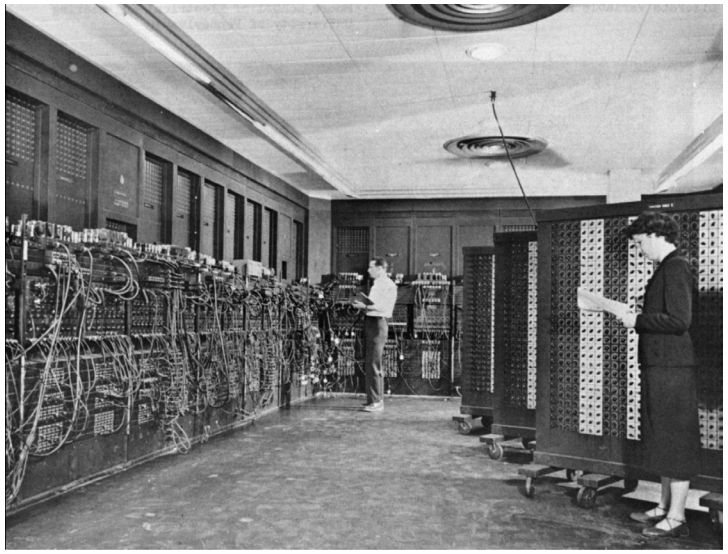
Examples: matrix multiplication, solving systems of linear equations, regression, eigenvalue and eigenvector problems, matrix decompositions, etc.



EARLY DAYS OF COMPUTING

First computers: ENIAC (decimal, 1945), followed by EDVAC (binary)

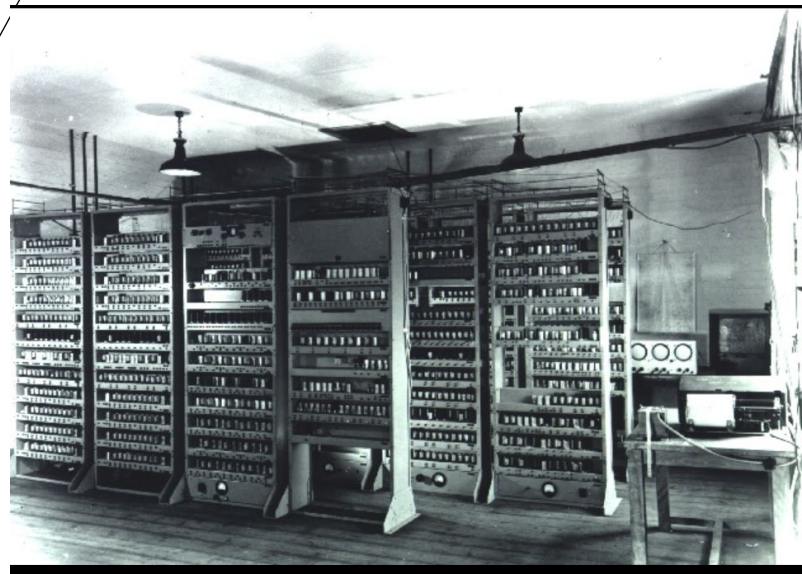
- First programmable, electronic, general-purpose digital computer
- Able to solve "a large class of **numerical problems**" through reprogramming
- Designed by **J. Mauchly** and **J. P. Eckert** to calculate artillery firing tables for the US Army Ballistic Research Laboratory
- Its first program was a study of the feasibility of the thermonuclear weapon
- **Specs:** 27 tons; 2m tall, 1m deep, 30m long; occupied 28 m^2
- **Performance:** 500 FLOPS!



EARLY DAYS OF COMPUTING

First computers: EDSAC (1949)

- Designed by **M. Wilkes**; inspired by **J. Von Neumann**'s seminal *"First Draft of a Report on the EDVAC"*
- Developed and implemented the concept of a stored-program computer, where both the program and data are stored in the same memory: *Von Neumann architecture*
- Flexibility and automation in computation.
- **Specs:** 2 tons; occupied 20 m^2
- **Performance:** 700-900 FLOPS!

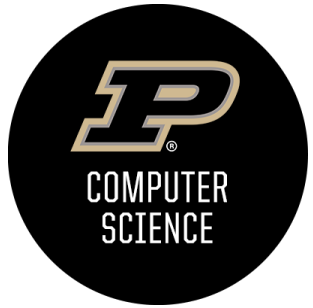


AN IRON LAW OF COMPUTING

Iron Law

All numbers used in a computer shall have a fixed number of digits. Therefore, the output of (almost) all primitive operations executed in a computer are wrong.

- **Major concern:** These *roundoff* errors accumulate and could be catastrophic[1].



AN IRON LAW OF COMPUTING

Iron Law

All numbers used in a computer shall have a fixed number of digits. Therefore, the output of (almost) all primitive operations executed in a computer are wrong.

- **Major concern:** These *roundoff* errors accumulate and could be catastrophic[1].

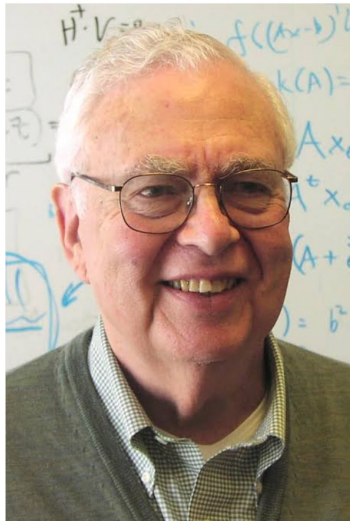
COMPUTER SCIENCE

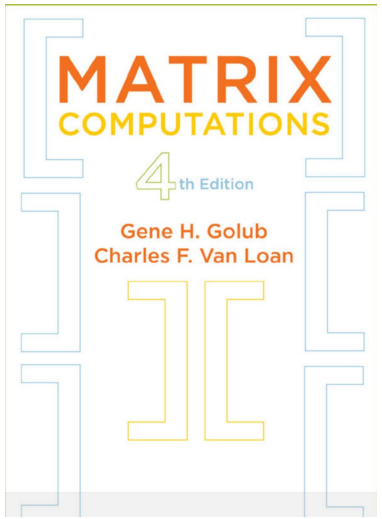
[1] **More precisely:** Roundoff errors accumulate and could be catastrophic for **ill-conditioned** problems!

THE SILENT KILLER: ROUND-OFF ERRORS

- Thus, there is an inherent error in any computer's most basic arithmetic operations.
- Round-off errors are quietly and quickly accumulated in every computation, and should not be overlooked.

Gene H. Golub
(1932-2007)



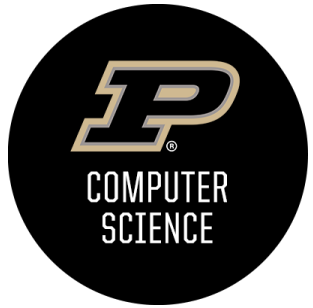


COMPUTER SCIENCE

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!



DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

Alan Turing
(1912-1954)

Worked on condition numbers of matrices for Gaussian Elimination, LU factorization, etc.

A detailed theory of condition numbers: John Rice, 1966.

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

John V Neumann
(1903-1957)

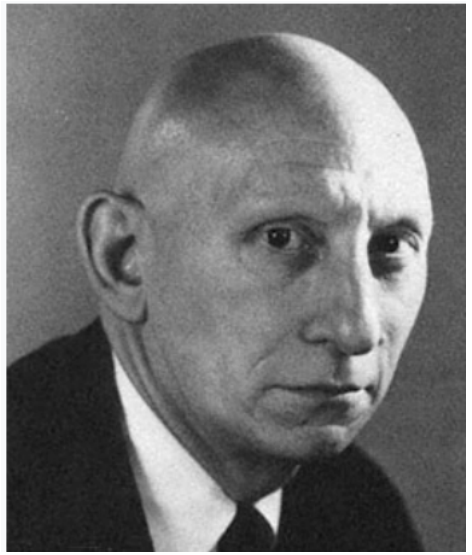
His work with Goldstine in 1947 on the "Numerical Inversion of Matrices of High Order" is widely considered the first NLA paper

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

Alan Perlis
(1922-1990)



Developed the *Purdue Compiler*, one of the first algebraic language compilers

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

James H. Wilkinson (1919-1986)

Popularized backward error analysis, the "backbone" of NLA

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because virtually all "early" Computer Science luminaries worked on addressing them.

Recipients of the ACM Turing award

Year	Recipient(s)	Photo	Rationale	Affiliated institute(s)
1966	Alan Perlis		"For his influence in the area of advanced computer programming techniques and compiler construction"[16][17]	Carnegie Mellon University
1967	Maurice Wilkes		"Wilkes is best known as the builder and designer of the EDSAC, the second computer with an internally stored program. Built in 1949, the EDSAC used a mercury delay line memory. He is also known as the author, with David Wheeler and Stanley Gill, of a volume on 'Preparation of Programs for Electronic Digital Computers' in 1951, in which program libraries were effectively introduced."[18][19]	University of Cambridge
1968	Richard Hamming		"For his work on numerical methods, automatic coding systems, and error-detecting and error-correcting codes"[20][21]	Bell Labs
1969	Marvin Minsky		"For his central role in creating, shaping, promoting, and advancing the field of artificial intelligence"[22][23]	Massachusetts Institute of Technology
1970	James H. Wilkinson		"For his research in numerical analysis to facilitate the use of the high-speed digital computer, having received special recognition for his work in computations in linear algebra and 'backward' error analysis"[24][25]	National Physical Laboratory

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

William M. Kahan (1933-)

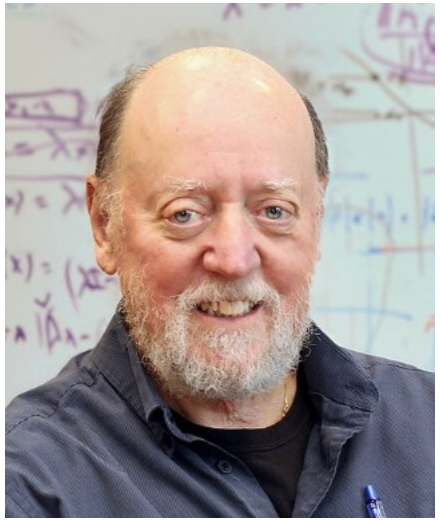
Developed the IEEE Floating Point Standard to simplify the task of writing high-quality, reliable numerical software

DO WE TRUST COMPUTERS TO "COMPUTE" GIVEN ROUNDOFF ERRORS?

Yes, because Computer Science luminaries worked on addressing them.

Many breakthroughs in numerical analysis and numerical linear algebra!

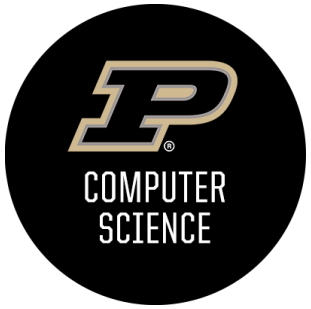
Jack J. Dongarra (1950-)

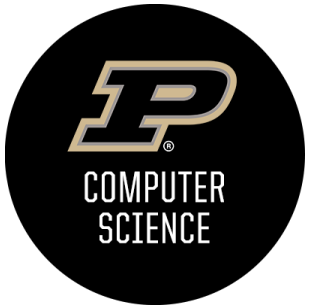


Developed the widely used LINPACK and LAPACK libraries with *J. Demmel*, *C. Moler*, etc.

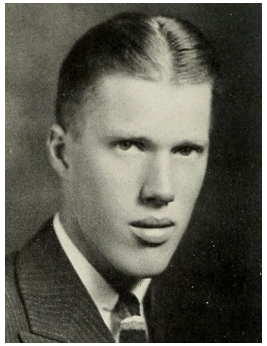
NUMERICAL ANALYSTS ALSO FOUNDED THE FIRST CS DEPARTMENTS IN THE US

Sam Conte (1917-2002): **Founded Purdue CS in 1962**





George Forsythe (1917-1972): **Founded Stanford CS in 1965**

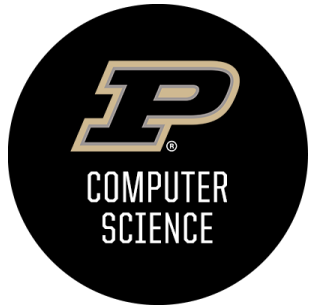


Stanford ENGINEERING | Computer Science

NLA & DS, ML, AI

NLA: An algorithmic foundation of DS, ML, and AI

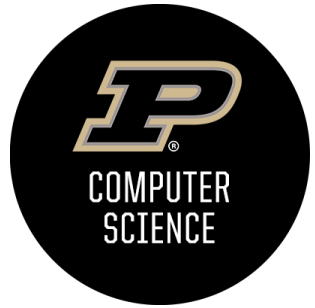
- **Language of Data:** Vectors, matrices, tensors represent structured and unstructured data.
- **Core Concepts:** Dot products, norms, projections underpin similarity, clustering, and classification.
- **Matrices in Practice:** Found in regression, graphs, embeddings, and recommendation systems.
- **Tensors in Deep Learning:** Represent inputs, weights, activations in multi-layer architectures.



NLA & DS, ML, AI

NLA: An algorithmic foundation of DS, ML, and AI

- **Language of Data:** Vectors, matrices, tensors represent structured and unstructured data.
- **Core Concepts:** Dot products, norms, projections underpin similarity, clustering, and classification.
- **Matrices in Practice:** Found in regression, graphs, embeddings, and recommendation systems.
- **Tensors in Deep Learning:** Represent inputs, weights, activations in multi-layer architectures.
- **Key Takeaway:** Without NLA, understanding, implementing, and evaluating modern data-driven AI is effectively impossible.



NLA & DS, ML, AI

NLA: Driving Efficiency, Stability, & Scalability

Efficient Algorithms: Solve large systems of linear equations, SVD, QR, LU with speed and accuracy.

Stability: Precision control and conditioning ensure model robustness.

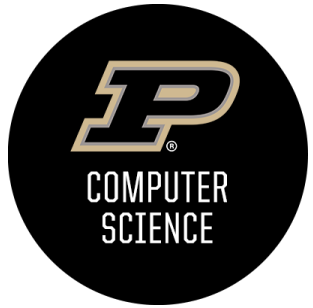
Big Data Scale: NLA enables scalable AI via matrix kernels on GPUs (e.g., cuBLAS, BLAS).

ML & AI Applications:

- *PCA* → Eigen-decomposition or Singular Value Decomposition
- *SVMs & Logistic Regression* → Matrix-vector operations
- *Neural Networks* → Matrix multiplications and convolutions

Oversimplifying (from a Lex Friedman podcast):

"Modern AI is just clever matrix multiplication"



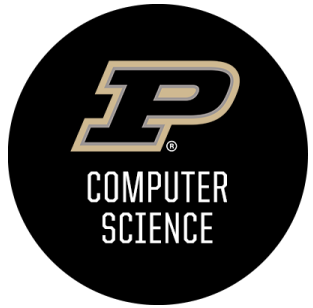
THE CHALLENGE: SCALE & COMPLEXITY

Big Data Bottlenecks: Traditional NLA has high computational complexity (e.g., $O(n^3)$ for SVD).

High Dimensionality: Text, images, and genomic data create storage and compute challenges.

Exactness is Expensive: Need for faster, approximate solutions to support real-world deployment.

Motivation for Randomized Methods: Can we trade a little accuracy for massive speed gains?



THE CHALLENGE: SCALE & COMPLEXITY

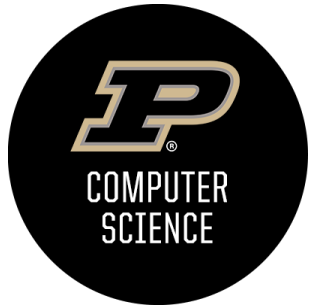
Big Data Bottlenecks: Traditional NLA has high computational complexity (e.g., $O(n^3)$ for SVD).

High Dimensionality: Text, images, and genomic data create storage and compute challenges.

Exactness is Expensive: Need for faster, approximate solutions to support real-world deployment.

Motivation for Randomized Methods: Can we trade a little accuracy for massive speed gains?

Enter the world of RandNLA: Randomized Numerical Linear Algebra!



RANDNLA

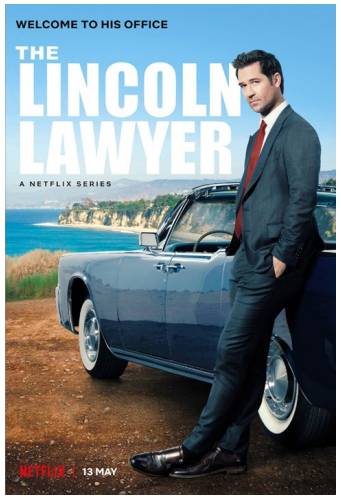
RandNLA: Randomized **N**umerical **L**inear **A**lgebra

Use randomization and sampling to design provably accurate and practically efficient matrix algorithms for “linear algebraic” problems that are:

- Massive (limited number of passes over the input matrix)
- NP-hard (k-means, sparse regression or sparse PCA, etc.)

NETFLIX

SEASON 1
EPISODE 3



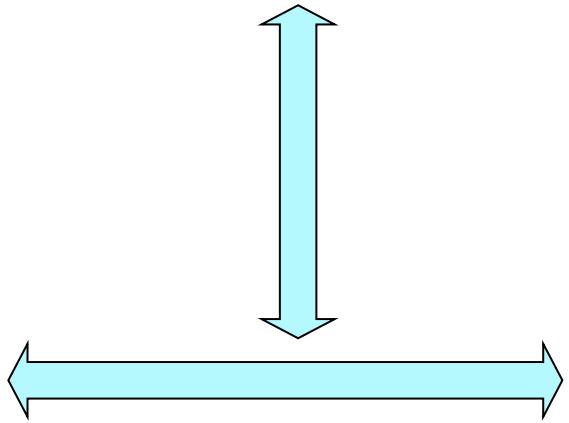
Interplay between disciplines

"Randomization is arguably the most exciting and innovative idea to have hit linear algebra in a long time."

(Avron et al. SISC 2010)

Applications

Data Science, Machine Learning, Bioinformatics, etc.



Theoretical Computer Science

Randomized & approximation algorithms

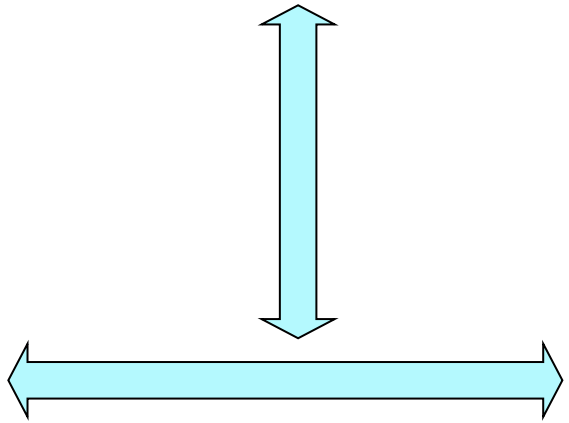
Applied Math

Numerical Analysis & Probability

Interplay between disciplines

"Randomization is arguably the most exciting and innovative idea to have hit linear algebra in a long time."

(Avron et al. SISC 2010)



Applications

Data Science, Machine Learning, Bioinformatics, etc.

Theoretical Computer Science

Randomized & approximation algorithms

Applied Math

Numerical Analysis & Probability

Many RandNLA events, including two Gene Golub SIAM Summer Schools (2015 and 2024), a semester long program at ICERM in Spring 2026, etc.

RandNLA: Randomized Numerical Linear Algebra

Matrices provide a natural structure with which to **model data**.

- $A \in \mathbb{R}^{m \times n}$ can encode information about m objects, each of which is described by n *features*; etc.
- A positive definite $A \in \mathbb{R}^{n \times n}$ can encode the correlations/similarities between all pairs of n objects; etc.

Motivated by data problems, recent years have witnessed **many exciting developments** in the theory and practice of matrix algorithms.

- Particularly remarkable is the use of *randomization*.
- Typically, it is assumed to be a property of the input data due (*e.g.*, to noise in the data generation mechanisms).
- Here, it is used as an algorithmic or computational resource.

RandNLA: Randomized Numerical Linear Algebra

An *interdisciplinary research area that exploits randomization as a computational resource to develop improved algorithms for large-scale linear algebra problems.*

- Foundational perspective: roots in theoretical computer science (TCS); deep connections with convex analysis, probability theory, and metric embedding theory, etc.; and strong connections with scientific computing, signal processing, and numerical linear algebra (NLA).
- Implementational perspective: well-engineered RandNLA algorithms beat highly-optimized software libraries for problems such as very over-determined least-squares and scale well to parallel/distributed environments.
- Data analysis perspective: strong connections with machine learning and statistics and many "non-methodological" applications of data analysis.

Growing interest in providing an *algorithmic and statistical foundation for modern large-scale data analysis.*

Randomized numerical linear algebra (RandNLA)

Lots of reviews of the past from multiple different perspectives.

- Tutorials, light on prerequisites
  - "RandNLA: randomized numerical linear algebra," by Drineas and Mahoney [DM16]
  - "Lectures on randomized numerical linear algebra," by Drineas and Mahoney [DM18]
- Broad and proof-heavy resources
  - "Sketching as a tool for numerical linear algebra," by Woodruff [Woo14]
  - "An introduction to matrix concentration inequalities," by Tropp [Tro15]
  - "Lecture notes on randomized linear algebra," by Mahoney [Mah16]
- Perspectives on theory, light on proofs
  - "Randomized algorithms for matrices and data," by Mahoney [Mah11]
  - "Determinantal point processes in randomized numerical linear algebra," by Dereziński and Mahoney [DM21]
- Deep investigations of specific topics
  - "Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions," by Halko, Martinsson, and Tropp [HMT11]
  - "Randomized algorithms in numerical linear algebra," by Kannan and Vempala [KV17]
  - "Randomized methods for matrix computations," by Martinsson [Mar18]
  - "Randomized numerical linear algebra: Foundations and Algorithms," by Martinsson and Tropp [MT20]

Basic Principles of "Classical" RandNLA [DM16]

Basic RandNLA method: given an input matrix:

- Construct a "sketch" (a smaller or sparser matrix that represents the essential information in the original matrix) by random sampling.
- Use that sketch as a surrogate to compute quantities of interest.

Basic design principles[1] underlying RandNLA:

- Randomly sample (in a careful data-dependent manner) a small number of elements to create a much sparser sketch of the original matrix.
- Randomly sample (in a careful data-dependent manner) a small number of columns and/or rows to create a much smaller sketch of the original matrix.
- Preprocess an input matrix with a random-projection-type matrix and then do uniform sampling of rows/columns/elements in order to create a sketch.

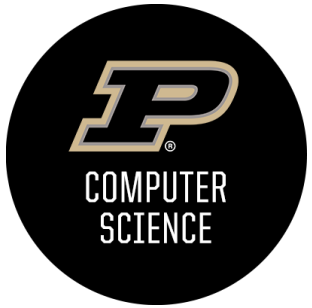
[1] First two principles deal with identifying nonuniformity structure. Third principle deals with preconditioning input (*i.e.*, uniformizing nonuniformity structure) s.t. uniform random sampling performs well.

HOW IMPORTANT IS NLA?

Top Ten Algorithms in Science (Jack Dongarra, 2000)

1. Metropolis Algorithm for Monte Carlo
2. Simplex Method for Linear Programming
3. Krylov Subspace Iteration Methods
4. The Decompositional Approach to Matrix Computations
5. The Fortran Optimizing Compiler
6. QR Algorithm for Computing Eigenvalues
7. Quicksort Algorithm for Sorting
8. Fast Fourier Transform
9. Integer Relation Detection
10. Fast Multipole Method

- Red: Algorithms within the exclusive domain of NLA research.
- Blue: Algorithms strongly (though not exclusively) connected to NLA research.



Small Singular Values *Increase* after Rounding

[for sufficiently tall-and-thin matrices; using stochastic rounding; with high probability; etc.]

Petros Drineas (Purdue CS)

Joint work with I. Ipsen (NCSU) & C. Boutsikas, G. Dexter, L. Ma (Purdue)

PURDUE UNIVERSITY

Research Topic

Rounding and the smallest singular value of a matrix

Given a matrix $\mathbf{A} \in \mathbb{R}^{n \times d}$ (exact representation), what happens to its smallest singular value after rounding $\mathbf{A}$ to $\tilde{\mathbf{A}} \in \mathcal{F}^{n \times d}$?

- Here $\mathcal{F}$ could be, for example, the set of all *double*, *single*, or *half* precision numbers.

Rounding as a perturbation

A straight-forward approach

- Model rounding error as a perturbation $\mathbf{E}$
- Formally, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{E}$
- Use perturbation theory to get bounds

What does Weyl's inequality reveal about the small singular values?

- If the largest singular value of $\mathbf{E}$ ("noise" due to rounding) is larger than the smallest singular value of $\mathbf{A}$, not much...

$$\underbrace{\sigma_{\min}(\mathbf{A}) - \|\mathbf{E}\|_2}_{\text{trivial if } \leq 0} \leq \sigma_{\min}(\underbrace{\mathbf{A} + \mathbf{E}}_{\tilde{\mathbf{A}}})$$

Prior knowledge

Large singular values remain unharmed, but small singular values tend to increase.

See, for example, [Stewart & Sun, 1990, pg. 266]

> *"...small singular values tend to increase" [under small perturbations]*

and [Rump, 2009, pg. 261]

> *"...even an approximate inverse of an arbitrarily ill-conditioned matrix does, in general, contain useful information. This is due to a kind of regularization by rounding to working precision."*

A higher order expansion

[Boutsikas, Drineas, and Ipsen SIMAX '24]

(Building upon [G. W. Stewart LAA '84])

- Partition the $n \times d$ matrices $\mathbf{A}$ and $\mathbf{E}$
- ($\mathbf{\Sigma}_1$ is $(d-1) \times (d-1)$)

$$\mathbf{A} = \mathbf{U} \begin{pmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \sigma_d \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{V}^T, \quad \mathbf{E} = \mathbf{U} \begin{pmatrix} \mathbf{E}_{11} & \mathbf{e}_{12} \\ \mathbf{e}_{21}^T & e_{22} \\ \mathbf{E}_{31} & \mathbf{e}_{32} \end{pmatrix} \mathbf{V}^T$$

A higher order expansion

[Boutsikas, Drineas, and Ipsen SIMAX '24]

- Assume
  1. Large singular values are large: $\sigma_{d-1} > 4\|\mathbf{E}\|_2$
  2. A single small singular value: $\sigma_d < \|\mathbf{E}\|_2$
- We prove[1]

$$\sigma_d(\mathbf{A}+\mathbf{E})^2 \geq (\sigma_d + e_{22})^2 + \|\mathbf{e}_{32}\|_2^2 - r_3 - r_4$$

- r_3, r_4 contains terms of $\mathcal{O}(\|\mathbf{E}\|_2^3)$ or higher

$$r_3 = 2\underbrace{\mathbf{e}_{12}^T (\mathbf{\Sigma}_1 + \mathbf{E}_{11})^{-T} \begin{pmatrix} \mathbf{e}_{21} & \mathbf{E}_{31}^T \end{pmatrix} \begin{pmatrix} e_{22} + \sigma_d \\ \mathbf{e}_{32} \end{pmatrix}}_{\mathbf{w}}$$

$$r_4 = \|\mathbf{w}\|_2^2 + 4\frac{\|\mathbf{E}\|_2^2 \|(\mathbf{\Sigma}_1 + \mathbf{E}_{11})^{-1}(\mathbf{e}_{12} + \mathbf{w})\|_2^2}{1 - 4\|\mathbf{E}\|_2^2 \|\|(\mathbf{\Sigma}_1 + \mathbf{E}_{11})^{-1}\|_2^2}$$

[1] We also prove a generalized version of this result for clusters of small singular values.

Pros & Cons

Pros

- True lower bound (beyond second order)
- Assumes a small gap between σ_{d-1}, σ_d
- Numerical experiments confirm our theory

Cons

- The higher order terms are challenging to interpret

Pros & Cons

Pros

- True lower bound (beyond second order)
- Assumes a small gap between σ_{d-1}, σ_d
- Numerical experiments confirm our theory

Cons

- The higher order terms are challenging to interpret

Let analyze *Stochastic Rounding (SR)*.

Normalized FP numbers

FP Model

- Given a basis β and a precision p

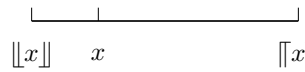
$$x = (-1)^s \cdot m \cdot \beta^{e-p}$$

- s is the sign bit
- e is the exponent
- The significand m is an integer in

$$\beta^{p-1} \leq m \leq \beta^p$$

Properties

- Let $\mathcal{F}$ be the set of normalized FP numbers and let $x \in \mathbb{R} - \mathcal{F}$
- The two FP numbers enclosing x are denoted by $\lfloor\!\lfloor x \rfloor\!\rfloor$, $\lceil\!\lceil x \rceil\!\rceil$



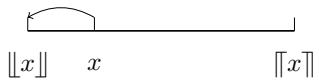
- The following inequality holds:

$$\max\{x - \lfloor\!\lfloor x \rfloor\!\rfloor, \lceil\!\lceil x \rceil\!\rceil - x\} \leq \beta^{1-p}|x|$$

Deterministic vs Stochastic Rounding (SR)

Deterministic

- Round-to-Nearest (RN)

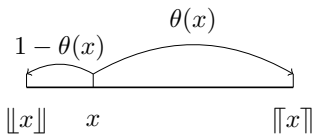


- For RN

$$\max\{x - \lfloor x \rfloor, \lceil x \rceil - x\} \leq 1/2 \beta^{1-p} |x|$$

Stochastic

- $\theta(x) = \dfrac{x - \lfloor x \rfloor}{\lceil x \rceil - \lfloor x \rfloor}$
- SR - *nearness*:

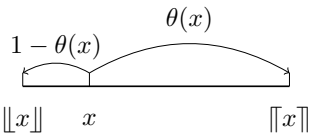


- Property: $\mathbb{E}\left[\mathtt{SR}(x)\right] = x$

Stochastic Rounding (SR)

Stochastic Rounding

- $\theta(x) = \dfrac{x - \lfloor x \rfloor}{\lceil x \rceil - \lfloor x \rfloor}$
- SR - *nearness*:



- Property: $\mathbb{E}\left[\mathtt{SR}(x)\right] = x$

History:

- Can be traced back to Forsythe 1950
- Also von Neumann & Goldstine 1947
- Recent resurgence: increasing interest for low-precision FP arithmetic for ML and DNNs [Gupta *et al.* 2015]
- Many patents held by (GPU) chip designers
- Review: Croci *et al.* 2022; Drineas & Ipsen 2024

SR: A simple example

Why SR?

- Let $\mathcal{F} = \{0, 1\}$ and consider the rank one matrix

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \\ \vdots & \vdots \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \xrightarrow{\mathtt{RN}(\mathbf{A})} \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ \vdots & \vdots \\ 1 & 1 \end{pmatrix}$$

- Any deterministic rounding will result to a rounded matrix $\tilde{\mathbf{A}}$ that is also rank one.

SR: A simple example

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

This is not the case for SR

- Let $\mathcal{F} = \{0, 1\}$ and consider the rank one matrix

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \\ \vdots & \vdots \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \xrightarrow{\mathtt{SR}(\mathbf{A})} \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 1 & 1 \end{pmatrix} = \tilde{\mathbf{A}}$$

- We can prove that for such $n \times 2$ matrices (with probability at least 0.997)

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \gtrapprox {}^{1}\!/_{2}\sqrt{n}$$

Our bound

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

For simplicity, assume $\mathbf{A} \in [-1, 1]^{n \times d}$ and let $\tilde{\mathbf{A}}$ be the stochastically rounded $\mathbf{A}$.

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \geq$$

Model

- $\mathbf{A} \in^{n \times d}$ with $n \gg d$
- SR to FP numbers
- $\mathbf{E} = \widetilde{\mathbf{A}} - \mathbf{A}$
- $\mathbb{E}[\mathbf{E}] = \mathbf{0}$

Ingredients

- β is the basis of our FP arithmetic
- p is the working precision

Our bound

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

For simplicity, assume $\mathbf{A} \in [-1,1]^{n\times d}$ and let $\tilde{\mathbf{A}}$ be the stochastically rounded $\mathbf{A}$.

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \geq \beta^{1-p}\sqrt{n}\,(\sqrt{\nu}-\varepsilon_{n,d})$$

Model

- $\mathbf{A} \in^{n\times d}$ with $n \gg d$
- SR to FP numbers
- $\mathbf{E} = \widetilde{\mathbf{A}} - \mathbf{A}$
- $\mathbb{E}[\mathbf{E}] = \mathbf{0}$

Ingredients

- β is the basis of our FP arithmetic
- p is the working precision
- ν measures the amount of available randomness during the rounding process
- $\varepsilon_{n,d}$ captures *lower-order* terms

Our bound: ν

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

For simplicity, assume $\mathbf{A} \in [-1, 1]^{n \times d}$ and let $\tilde{\mathbf{A}}$ be the stochastically rounded $\mathbf{A}$.

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \geq \beta^{1-p}\sqrt{n}\left(\sqrt{\nu} - \varepsilon_{n,d}\right)$$

Model

- $\mathbf{A} \in^{n \times d}$ with $n \gg d$
- SR to FP numbers
- $\mathbf{E} = \widetilde{\mathbf{A}} - \mathbf{A}$
- $\mathbb{E}[\mathbf{E}] = \mathbf{0}$

Understanding ν

- Consider a matrix with, say, two identical columns whose entries are FPs: $\sigma_{\min}(\mathbf{A}) = 0$.
- SR will not modify those columns: $\sigma_{\min}(\tilde{\mathbf{A}}) = 0$.
- Intuitively: no randomness for SR to exploit.
- This lack of randomness is captured by ν, which, in this case, is equal to zero.

Our bound: ν

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

For simplicity, assume $\mathbf{A} \in [-1, 1]^{n \times d}$ and let $\widetilde{\mathbf{A}}$ be the stochastically rounded $\mathbf{A}$.

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \geq \beta^{1-p}\sqrt{n}\left(\sqrt{\nu} - \varepsilon_{n,d}\right)$$

Model

- $\mathbf{A} \in^{n \times d}$ with $n \gg d$
- SR to FP numbers
- $\mathbf{E} = \widetilde{\mathbf{A}} - \mathbf{A}$
- $\mathbb{E}[\mathbf{E}] = \mathbf{0}$

Understanding ν

- Formally[a]: $\nu \propto \min_{\text{all columns } j} \sum_{i=1}^{n} \mathrm{Var}\left(\mathbf{E}_{ij}\right)$
- $0 \leq \nu \leq 1$

[a] Skipping a normalization factor

Interpreting our bound

For simplicity, assume $\mathbf{A} \in [-1, 1]^{n \times d}$ and let $\tilde{\mathbf{A}}$ be the stochastically rounded $\mathbf{A}$.

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \geq \beta^{1-p}\sqrt{n}\left(\sqrt{\nu} - \varepsilon_{n,d}\right)$$

- As n grows, $\sigma_{\min}\left(\widetilde{\mathbf{A}}\right)$ increases
- β^{1-p} captures the parameters of FP arithmetic
- ν captures the amount of available *stochasticity* in $\mathtt{SR}(\mathbf{A})$
- $\varepsilon_{n,d}$ depends only on n, d:
  - → If n is $\omega(d^4)$, then $\lim_{n\to\infty} \varepsilon_{n,d} = 0$.
  - → (hiding $\log$ factors)

Our main result: A perturbation theory bound

Main Theorem

Let $\mathbf{A}$ and $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{E}$ be real $n \times d$ matrices. Here $\mathbf{E}$ models a zero-mean random perturbation matrix with minimal (normalized) column variance ν and $\max_{i,j} |\mathbf{E}_{ij}| \leq \rho$.

If $n \geq 836$, then with probability at least $1 - 1/n^c - 2d^2/n^2$,

$$\sigma_{\min}(\widetilde{\mathbf{A}}) \geq \rho\sqrt{n}(\sqrt{\nu} - \varepsilon_{n,d}),$$

where

$$\varepsilon_{n,d} \equiv \sqrt{\frac{d}{n}} + 2d^2\sqrt{\frac{\log n}{n}} + \frac{C(\log n)^{2/3}}{n^{1/30}} \cdot \left(\frac{d}{n}\right)^{\frac{1}{54}},$$

and c and C are absolute constants.

Our bound is tight

[Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]

Tightness of our bound

Let $\mathbf{A}$ and $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{E}$ be real $n \times d$ matrices. Here $\mathbf{E}$ models a zero-mean random perturbation matrix with minimal (normalized) column variance ν and $\max_{i,j} |\mathbf{E}_{ij}| \leq \rho$.

Our main bound is that, with high probability,

$$\sigma_{\min}(\widetilde{\mathbf{A}}) \gtrapprox \rho\sqrt{n\nu}.$$

We exhibit $n \times d$ matrices $\mathbf{A}$ for which SR returns the matrix $\widetilde{\mathbf{A}}$ such that

$$\sigma_{\min}\left(\widetilde{\mathbf{A}}\right) \leq \left(1 + \sqrt{1/(d-1)}\right) \cdot \rho\sqrt{n\nu}.$$

Proof outline

Steps:

1. We introduce the orthogonal projector $\mathbf{P_A}$ onto the left column space of $\mathbf{A}$. This allows us to focus on $\mathbf{P_A E}$.
2. Weyl's inequality yields a lower bound on the smallest singular value of $(\mathbf{I} - \mathbf{P_A})\mathbf{E}$ by lower bounding the smallest singular value of $\mathbf{E}$ and upper bounding the largest singular value of $\mathbf{P_A E}$.
3. Application of a Random Matrix Theory bound from [Dumitriu & Zhu '24] shows that the smallest singular value of $\mathbf{E}$ is sufficiently large.
4. The largest singular value of the projection $\mathbf{P_A E}$ is small, because $\mathbf{P_A}$ projects $\mathbf{E}$ on the low-dimensional subspace of dimension d.
5. Standard measure concentration bounds show that $\mathbf{E}$ *does not concentrate* in any low-dimensional subspace.
6. Finally, we combine the bounds for the smallest singular value of $\mathbf{E}$ and the largest singular value of $\mathbf{P_A E}$.

Main theorem: Proof sketch (I)

Orthogonal Projectors

- $\mathbf{P}_{\mathbf{A}} \in \mathbb{R}^{n \times n} \quad \rightarrow \quad$ onto column space of $\mathbf{A}$
- $\mathbf{P}_{\mathbf{A},\perp} = \mathbf{I}_n - \mathbf{P}_{\mathbf{A}} \quad \rightarrow \quad$ onto left nullspace of $\mathbf{A}$

Product on singular values & Weyl's

- $\sigma_{\min}(\mathbf{A} + \mathbf{E}) \geq \sigma_{\min}(\mathbf{P}_{\mathbf{A},\perp}\mathbf{E}) \geq \sigma_{\min}(\mathbf{E} - \mathbf{P}_{\mathbf{A}}\mathbf{E}) \geq \sigma_{\min}(\mathbf{E}) - \|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$
- [Theorem 3.3.16, Horn & Johnson '91] [Weyl's inequality]

Lower bound for $\sigma_{\min}(\mathbf{E})$ and upper bound for $\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$

Main theorem: Proof sketch (II)

Lower bound on $\sigma_{\min}(\mathbf{E})$

- Random Matrix Theory
- Theorem 2.10, Dumitriu & Zhu '24[a]
- $\sigma_{\min}(\mathbf{E}) \geq \rho\sqrt{\nu \cdot n} - \rho\sqrt{d} - \frac{C \cdot \rho \cdot (\log n)^{2/3}}{n^{1/30}} \cdot \left(\frac{d}{n}\right)^{1/54} \cdot \sqrt{n}$

[a] To be precise, we used Theorem 2.10 from arXiv v2 Oct '22.

Upper bound on $\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$

- Concentration inequalities
- $\mathbb{P}\left[\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2 \geq 2d^2 \cdot \rho\sqrt{\log n}\right] \leq \frac{2d^2}{n^2}$

Recall

$$\sigma_{\min}(\mathbf{A}+\mathbf{E}) \geq \sigma_{\min}(\mathbf{E}) - \|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$$

Experimental setting

Objective

- Investigate $\sigma_{\min}(\widetilde{\mathbf{A}})$

Factors

- aspect ratio n/d
- smallest singular value $\sigma_{\min}(\mathbf{A})$
- minimum column variance ν

Design

- $n = 10^4$ and $d = 10;\ 100;\ 1000$
- $\mathcal{F} = \{\text{Fixed point, Floating point}\}$
- $\mathbf{A}_{ij} \sim \mathcal{N}(0, 1)$ or $\texttt{LogNorm}(0, 3)$
- Modify $\mathbf{A}$ to control $\sigma(\mathbf{A})$
- Run $\texttt{SR}(\mathbf{A})$ 100 times $\rightarrow \sigma_{\min}(\widetilde{\mathbf{A}})$
- Compute $s_{\min} = \min\{\sigma_{\min}(\widetilde{\mathbf{A}})\}$

Output

- % of $\left(\sigma_{\min}(\widetilde{\mathbf{A}}) < c\rho\sqrt{n\nu}\right)$ violations
- Relative error of $s_{\min}$, $\rho\sqrt{n\nu}$

Fixed-point arithmetic: Singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & $\mathcal{N}(0,1)$

Fixed point arithmetic with $\beta = 10$

- All elements of $\mathtt{SR}(\mathbf{A}) \in \mathcal{F}^{\{p\}}$, $1 \le p \le 3$

$$\mathcal{F}^{\{p\}} = \{\pm {}^{m}/_{10^p}, \quad \text{for all integers } m = \underbrace{0, 1, 2, \ldots, 10^p - 1}_{\le p \text{ digits}}\} \cup \{\pm 1\}$$

- $\lceil\!\lceil \mathbf{A}_{ij} \rceil\!\rceil - \lfloor\!\lfloor \mathbf{A}_{ij} \rfloor\!\rfloor \le 10^{-p}$

Matrices

- $n = 10^4$ and $d = 10;\ 100;\ 1000$
- $\mathbf{A}_{ij} \sim \mathcal{N}(0,1)$
- Modify $\mathbf{A}$ to set $\sigma_{\min}(\mathbf{A}) = 0$

Fixed-point arithmetic: Singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & $\mathcal{N}(0,1)$

d	$\% \left(\sigma_{\min}(\widetilde{\mathbf{A}}) < c \cdot \rho\sqrt{n\nu}\right)$						$1 - s_{\min}/\rho\sqrt{n\nu}$		
	p = 1		**p = 2**		**p = 3**		**p = 1**	**p = 2**	**p = 3**
	$c = 1$	$c = .9$	$c = 1$	$c = .9$	$c = 1$	$c = .9$			
10	26%	0%	37%	0%	30%	0%	.01	.01	.01
100	48%	0%	46%	0%	51%	0%	.02	.01	.02
1000	100%	0%	100%	0%	100%	0%	.06	.06	.06

Fixed-point arithmetic: Non-singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & $\texttt{LogNorm}(0, 3)$

Fixed point arithmetic with $\beta = 10$

- All elements of $\texttt{SR}(\mathbf{A}) \in \mathcal{F}^{\{p\}}$, $1 \le p \le 3$

$$\mathcal{F}^{\{p\}} = \{\pm m/10^p, \quad \text{for all integers } m = \underbrace{0, 1, 2, \ldots, 10^p - 1}_{\le p \text{ digits}}\} \cup \{\pm 1\}$$

- $\llbracket \mathbf{A}_{ij} \rrbracket - \lfloor\!\lfloor \mathbf{A}_{ij} \rfloor\!\rfloor \le 10^{-p}$

Matrices

- $n = 10^4$ and $d = 10;\ 100;\ 1000$
- $\mathbf{A}_{ij} \sim \texttt{LogNorm}(0, 3)$
- Modify $\mathbf{A}$ to set $\sigma_{\min}(\mathbf{A}) = 10^{-2}$

Fixed-point arithmetic: Non-singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & $\texttt{LogNorm}(0, 3)$

d	$\% \left(\sigma_{\min}(\widetilde{\mathbf{A}}) < c \cdot \rho\sqrt{n\nu} \right)$						$1 - s_{\min}/\rho\sqrt{n\nu}$		
	p = 1		**p = 2**		**p = 3**		**p = 1**	**p = 2**	**p = 3**
	$c = 1$	$c = .9$	$c = 1$	$c = .9$	$c = 1$	$c = .9$			
10	0%	0%	15%	0%	22%	0%	N/A	.01	.01
100	0%	0%	36%	0%	34%	0%	N/A	.01	.01
1000	100%	0%	100%	0%	100%	0%	.04	.05	.06

Floating-point arithmetic: Singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & controlled ν

Lower precision via `SR`

- Demote $\mathbf{A}$ in lower precision by emulating SR-nearness

Matrices

- $n = 10^4$ and $d = 10;\ 100;\ 1000$
- $\mathbf{A}_{ij} \sim \mathcal{N}(0, 1)$
- A^h "high" value of ν
- A^l "low" value of ν
- $\nu(A^h)/\nu(A^l) \approx 100$
- Modify $\mathbf{A}^h, \mathbf{A}^l$ to be singular (without affecting $\nu(A^h)/\nu(A^l)$)

Floating-point arithmetic: Singular $\mathbf{A} \in \mathbb{R}^{n \times d}$ & controlled ν

$\mathbf{d}$	$\% \left(\sigma_{\min}(\widetilde{\mathbf{A}}) < c\rho\sqrt{n\nu} \right)$				$1 - s_{\min}/\rho\sqrt{n\nu}$	
	$\widetilde{\mathbf{A}}^{\mathbf{h}}$		$\widetilde{\mathbf{A}}^{\mathbf{l}}$		$\widetilde{\mathbf{A}}^{\mathbf{h}}$	$\widetilde{\mathbf{A}}^{\mathbf{l}}$
	$c = 1$	$c = .8$	$c = 1$	$c = .8$		
$\mathbf{10}$	46%	0%	54%	0%	.02	.1
$\mathbf{1,000}$	100%	0%	75%	2%	.05	.2

An improved bound on $\sigma_{\min}(\widetilde{\mathbf{A}})$ [Ma, Yu, and Drineas, 2026]

Recall: $\sigma_d(\mathbf{A}+\mathbf{E}) \geq \sigma_d(\mathbf{E}) - \|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$

Upper bound on $\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2$

- Previous:
  $\mathbb{P}\left[\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2 \geq 2d^2\rho\sqrt{\log n}\right] \leq \frac{2d^2}{n^2}$
- New (via ε-net + Hoeffding):
  $\mathbb{P}\left[\|\mathbf{P}_{\mathbf{A}}\mathbf{E}\|_2 > c_2\rho\left(\sqrt{d+r}+u\right)\right] \leq 2e^{-c_2u^2}$

Lower bound on $\sigma_d(\mathbf{E})$

- Previous: Dumitriu & Zhu '24
- New: Brailovskaya & van Handel '24

Improved result

If $d \leq c_1 n$, then, with probability at least $1 - 1/\mathrm{poly}(n)$,

$$\sigma_d(\widetilde{\mathbf{A}}) \geq c_0\,\rho\sqrt{n}$$

- **Key improvement:** aspect ratio relaxes from $d = o\left(n^{1/4}\right)$ to $d \leq c_1 n$.
- I hid that n and ν must be sufficiently large.

Beyond $\sigma_{\min}$: What about a *cluster* of small singular values?

So far: Lower bounds on $\sigma_{\min}(\widetilde{\mathbf{A}})$ after stochastic rounding.

New question: Suppose $\mathbf{A}$ has a spectral gap at index k:

$$\sigma_1 \geq \cdots \geq \sigma_k \gg \sigma_{k+1} \geq \cdots \geq \sigma_d$$

Does SR increase the small singular values collectively?

Measure the increase:

$$\mathcal{T} := \mathbb{E}\left[\sum_{i=k+1}^{d} \sigma_i(\widetilde{\mathbf{A}})^2\right] - \sum_{i=k+1}^{d} \sigma_i(\mathbf{A})^2$$

Key observation: Since $\mathbb{E}[\mathbf{E}] = \mathbf{0}$, the total squared singular-value mass increases:

$$\mathbb{E}\left[\sum_{i=1}^{d} \sigma_i(\widetilde{\mathbf{A}})^2\right] = \sum_{i=1}^{d} \sigma_i(\mathbf{A})^2 + \mathbb{E}\|\mathbf{E}\|_F^2$$

So any rounding energy *not* absorbed by the top k goes to the small cluster.

Bounding the top-k change [Ma, Yu, and Drineas, 2026]

Proof: contour integral and resolvent expansion of $\mathbf{A}^\top\mathbf{A}$ (Tran, Vu, Vishnoi NeurIPS 2025)

- Gram matrices: $\mathbf{G} = \mathbf{A}^\top\mathbf{A}$, $\widetilde{\mathbf{G}} = \widetilde{\mathbf{A}}^\top\widetilde{\mathbf{A}}$, $\lambda_i = \sigma_i^2$ (and $\mathbf{\Delta} = \mathbf{G} - \widetilde{\mathbf{G}}$)
- Spectral gap: $g = \lambda_k - \lambda_{k+1}$; contour Γ encloses exactly $\lambda_1, \ldots, \lambda_k$
- Represent the top-k change via the resolvent $R_{\mathbf{G}}(z) = (z\mathbf{I} - \mathbf{G})^{-1}$:

$$\sum_{i=1}^{k} \sigma_i(\mathbf{A}+\mathbf{E})^2 - \sum_{i=1}^{k} \sigma_i(\mathbf{A})^2 = \frac{1}{2\pi i} \oint_\Gamma z \operatorname{tr}\big(R_{\widetilde{\mathbf{G}}}(z) - R_{\mathbf{G}}(z)\big)\, dz$$

- Neumann-series expansion for the resolvent:

$$\sum_{i=1}^{k} \sigma_i(\mathbf{A}+\mathbf{E})^2 - \sum_{i=1}^{k} \sigma_i(\mathbf{A})^2 = \sum_{j=1}^{\infty} \frac{1}{2\pi i} \oint_\Gamma z \operatorname{tr}\big((R_{\mathbf{G}}(z)\mathbf{\Delta})^j R_{\mathbf{G}}(z)\big)\, dz$$

- Each term bounded via: Cauchy residues, Hanson–Wright, ε-net arguments

Main result: SR regularizes a small cluster [Ma, Yu, and Drineas, 2026]

Setting: $\mathbf{A} \in \mathbb{R}^{n\times d}$, $\widetilde{\mathbf{A}} = \mathtt{SR}(\mathbf{A}) = \mathbf{A} + \mathbf{E}$, gap at index k: $\sigma_k \gg \sigma_{k+1}$

Column variances: $\nu_i := \sum_{m=1}^{n} \mathbb{E}[E_{mi}^2]$; $\nu_1^{\downarrow} \geq \cdots \geq \nu_d^{\downarrow}$: decreasing rearrangement

Simplified bound

If $\rho\sqrt{nd} \ll \sigma_k$ and $\sigma_1 = O(\sigma_k)$, then

$$\mathcal{T} \geq \sum_{j=k+1}^{d} \nu_j^{\downarrow} - c_0\left(kd\,\rho^2 + \frac{kn^2\rho^4}{\sigma_k^2} + \frac{d\,n^{3/2}\rho^3}{\sigma_k}\right)$$

Interpretation:

- Leading term $\sum_{j=k+1}^{d} \nu_j^{\downarrow}$: the stochasticity available for the small cluster
- Correction terms are lower order when ρ is small and the gap is large

SR in a downstream application: Least-squares regression

- Consider the least-squares (LS) problem

$$\mathbf{x}_{\texttt{opt}} = \arg\min_{\mathbf{x}\in\mathbb{R}^d} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$$

- Assume tall-and-thin $\mathbf{A} \in \mathbb{R}^{n\times d}$ with $n \gg d$

$$\mathbf{A} \leftarrow \sum_{i=1}^{d} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \quad \mathbf{x}_{\texttt{opt}} \leftarrow \sum_{i=1}^{d} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i$$

- What happens if $\mathbf{A}$ is stochastically rounded or Quantized to $\tilde{\mathbf{A}} = \texttt{SR}(\mathbf{A})$ and the resulting LS problem is solved?

$$\tilde{\mathbf{A}} = \texttt{SR}(\mathbf{A}) \leftarrow \sum_{i=1}^{d} \tilde{\sigma}_i \tilde{\mathbf{u}}_i \tilde{\mathbf{v}}_i^T, \quad \mathbf{x}_{\texttt{Q,opt}} \leftarrow \sum_{i=1}^{d} \frac{\tilde{\mathbf{u}_i}^T \mathbf{b}}{\tilde{\sigma}_i} \tilde{\mathbf{v}}_i$$

SR seems to behave like Truncated LS

- Consider the least-squares (LS) problem

$$\mathbf{x}_{\texttt{opt}} = \arg\min_{\mathbf{x}\in\mathbb{R}^d} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$$

- Truncated LS solution ($k < d$)

$$\mathbf{A} \leftarrow \sum_{i=1}^{d} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \quad \mathbf{x}_{\texttt{T,opt}} \leftarrow \sum_{i=1}^{k} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i$$

- Recall the solution to the LS problem when $\tilde{\mathbf{A}} = \texttt{SR}(\mathbf{A})$ is used instead of $\mathbf{A}$:

$$\mathbf{x}_{\texttt{Q,opt}} \leftarrow \sum_{i=1}^{d} \frac{\tilde{\mathbf{u}}_i^T \mathbf{b}}{\tilde{\sigma}_i} \tilde{\mathbf{v}}_i$$

Intuition

SR increases the bottom singular values of $\mathbf{A}$, and reduces their "influence" on the LS solution

A toy LS example: SR on A, keeping one decimal digit

- $\mathbf{A} = \begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}, \quad \dfrac{\sigma_2}{\sigma_3} = 100$
- True parameters $\rightarrow \mathbf{x}^\star = (1, 1, 0)^T$
- Noisy observations $\rightarrow \mathbf{b} = \underbrace{\begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}}_{\mathbf{x}^\star} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0.01 \end{pmatrix}}_{\texttt{noise}} = \begin{pmatrix} 20 \\ 2 \\ 0.01 \end{pmatrix}$

Standard regression

- $\hat{\mathbf{x}} = \mathbf{A}^\dagger \mathbf{b} = (1, 1, 0.5)^T$
- $\|\mathbf{x}^\star - \hat{\mathbf{x}}\|_2 = 0.5$

Truncated SVD ($k = 2$)

- $\mathbf{x}_{\mathrm{T},\ \mathtt{opt}} = \mathbf{A}_k^\dagger \mathbf{b} = (1, 1, 0)^T$
- $\|\mathbf{x}^\star - \mathbf{x}_{\mathrm{T},\ \mathtt{opt}}\|_2 = 0$

A toy LS example: SR on $\mathbf{A}$, keeping one decimal digit

- $\mathbf{A} = \begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}, \quad \frac{\sigma_2}{\sigma_3} = 100$
- True parameters $\rightarrow \mathbf{x}^\star = (1, 1, 0)^T$
- Noisy observations $\rightarrow \mathbf{b} = \underbrace{\begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}}_{\mathbf{x}^\star} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0.01 \end{pmatrix}}_{\texttt{noise}} = \begin{pmatrix} 20 \\ 2 \\ 0.01 \end{pmatrix}$

Standard regression

- $\hat{\mathbf{x}} = (1, 1, 0.5)^T$
- $\|\mathbf{x}^\star - \hat{\mathbf{x}}\|_2 = 0.5$

Truncated SVD ($k = 2$)

- $\mathbf{x}_{\mathtt{T},\ \mathtt{opt}} = (1, 1, 0)^T$
- $\|\mathbf{x}^\star - \mathbf{x}_{\mathtt{T},\ \mathtt{opt}}\|_2 = 0$

SR ($\tilde{\mathbf{A}}_{3,3} = 0$, with prob. 0.8)

- $\mathbf{x}_{\mathtt{Q},\ \mathtt{opt}} = (1, 1, 0)^T$
- $\|\mathbf{x}^\star - \mathbf{x}_{\mathtt{Q},\ \mathtt{opt}}\|_2 = 0$

A toy LS example: SR on $\mathbf{A}$, keeping one decimal digit

- $\mathbf{A} = \begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}, \quad \frac{\sigma_2}{\sigma_3} = 100$
- True parameters $\rightarrow \mathbf{x}^\star = (1, 1, 0)^T$
- Noisy observations $\rightarrow \mathbf{b} = \underbrace{\begin{pmatrix} 20 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}}_{\mathbf{x}^\star} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0.01 \end{pmatrix}}_{\texttt{noise}} = \begin{pmatrix} 20 \\ 2 \\ 0.01 \end{pmatrix}$

Standard regression

- $\hat{\mathbf{x}} = (1, 1, 0.5)^T$
- $\|\mathbf{x}^\star - \hat{\mathbf{x}}\|_2 = 0.5$

Truncated SVD ($k = 2$)

- $\mathbf{x}_{\mathtt{T},\ \mathtt{opt}} = (1, 1, 0)^T$
- $\|\mathbf{x}^\star - \mathbf{x}_{\mathtt{T},\ \mathtt{opt}}\|_2 = 0$

SR ($\tilde{\mathbf{A}}_{3,3} = 0.1$, w.p. 0.2)

- $\mathbf{x}_{\mathtt{Q},\ \mathtt{opt}} = (1, 1, 0.1)^T$
- $\|\mathbf{x}^\star - \mathbf{x}_{\mathtt{Q},\ \mathtt{opt}}\|_2 = 0.1$

What can we prove?

Notation: $\xi = \max_{i \in \{k+1,\cdots,d\},\, j \in \{1,\cdots,k\}} |\tilde{\mathbf{u}}_i^T \boldsymbol{\Delta}\, \mathbf{v}_j|, |\tilde{\mathbf{v}}_i^T \boldsymbol{\Delta}^T \mathbf{u}_j|,$
$\gamma = \sigma_{k+1}/\sigma_k, \eta = \|\boldsymbol{\Delta}\|_2/\sigma_k, \alpha = 1/1-\eta, \beta = 1/1-\eta-\gamma$

$$\frac{\|\mathbf{x}_{\mathtt{T},\mathtt{opt}} - \mathbf{x}_{\mathtt{Q},\mathtt{opt}}\|_2}{\|\mathbf{x}_{\mathtt{T},\mathtt{opt}}\|_2} \leq (\alpha+\beta)\,\eta + 2\beta\kappa(\mathbf{A}_k)\frac{\sqrt{(d-k)k}}{\tilde{\sigma}_d}\xi$$

When is the above relative error small?

- The gap between σ_k, σ_{k+1} is large (γ is small)
- $\mathbf{A}_k$ is well-conditioned and ξ is small
- σ_k is larger than the norm of the SR perturbation $\|\boldsymbol{\Delta}\|_2$
- $\tilde{\sigma}_d$ is not too small
- For simplicity, we assume that $\mathbf{b}$ lies in $\mathtt{range}(\mathbf{A}_k)$; can be relaxed

What can we prove?

Notation:

- $\mathbf{A} \in \mathbb{R}^{n \times d}$ and $\texttt{SR}(\mathbf{A}) = \mathbf{A} + \mathbf{\Delta}$
- $\eta = \dfrac{\|\mathbf{\Delta}\|_2}{\sigma_k}$
- ν measures the column-wise amount of "randomness" in $\texttt{SR}(\mathbf{A})$

Assumption: $\xi = O(\rho)$

$$\frac{\|\mathbf{x}_{\texttt{T},\texttt{opt}} - \mathbf{x}_{\texttt{Q},\texttt{opt}}\|_2}{\|\mathbf{x}_{\texttt{T},\texttt{opt}}\|_2} \leq c_0 \cdot \frac{\kappa(\mathbf{A}_k)}{\sqrt{\nu}} \cdot \left(\eta + \frac{\sqrt{(d-k)k}}{\sqrt{n}} \right)$$

Follows from the previous bound, after:

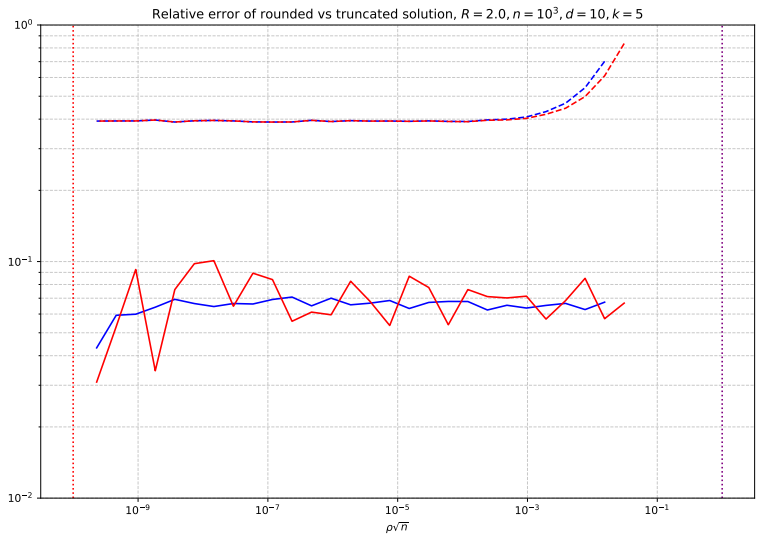
- Lower-bounding $\tilde{\sigma}_d$ [Dexter, Boutsikas, Ma, Ipsen, and Drineas SIMAX '25]
- and using an ϵ-net argument to bound $\|\mathbf{\Delta}\|_2$ [Vershynin '18]

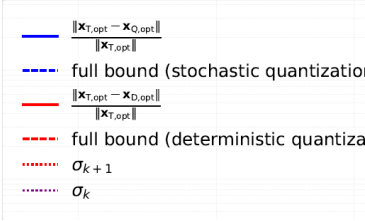
Experimental evaluation: Quantized regression

Setting

- $\mathbf{A}_{ij} \sim \texttt{Unif}[-1, 1]$
- Modify $\sigma(\mathbf{A})$ such as: $\sigma_{k+1} = \cdots = \sigma_d = 10^{-10}$
- Set R such as $\max_{i,j} |\mathbf{A}_{ij}| \leq R$
- Stochastic quantization $\rightarrow$ average over 50 repetitions
- Generate $\mathbf{b}$ such as
  - $\|\mathbf{b}_\perp\|_2 = 0$ ($\mathbf{b} \in \texttt{range}(\mathbf{A}_k)$)
  - $\|\mathbf{b}_\perp\|_2 = \underbrace{10^{-10}\|\mathbf{U}_k\mathbf{U}_k^T\mathbf{b}\|_2}_{\epsilon}$

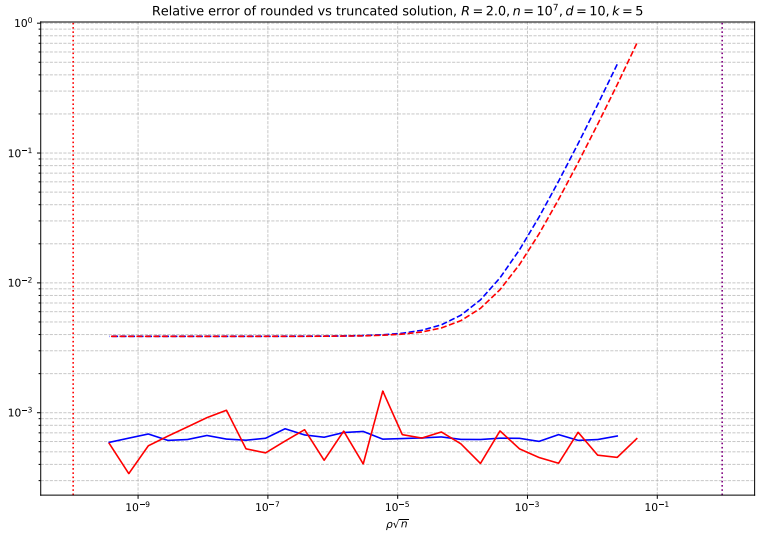
Experimental evaluation: Quantized regression ($\mathrm{b} \in \mathtt{range}(\mathrm{A}_k)$)

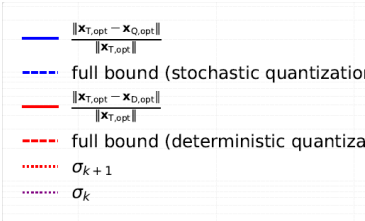




- $n/d = 10^2$
- $\|\mathbf{A}_{k,\perp}\mathbf{A}_{k,\perp}^{\dagger}\mathbf{b}\| = 0$

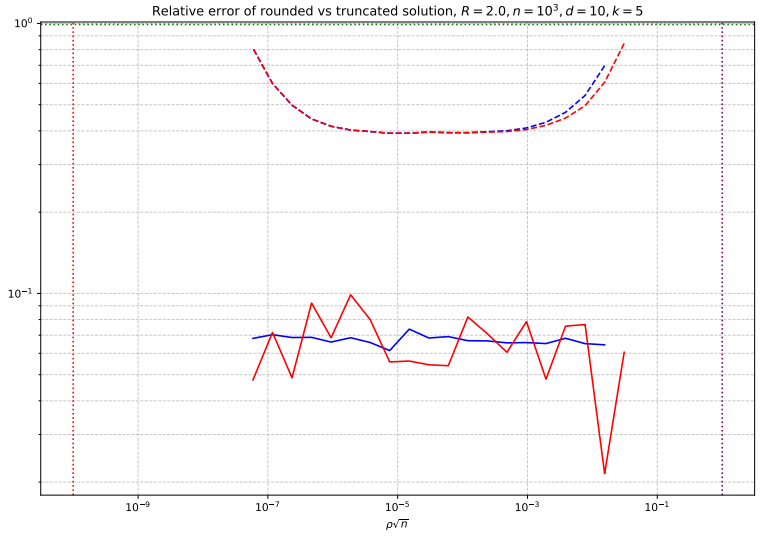
Experimental evaluation: Quantized regression ($\mathrm{b} \in \mathtt{range}(\mathrm{A}_k)$)

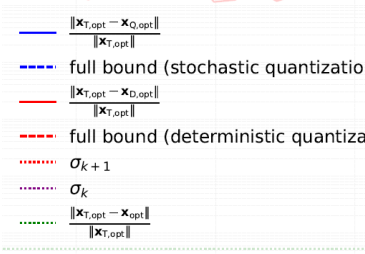




- $n/d = 10^6$
- $\|\mathbf{A}_{k,\perp}\mathbf{A}_{k,\perp}^{\dagger}\mathbf{b}\| = 0$

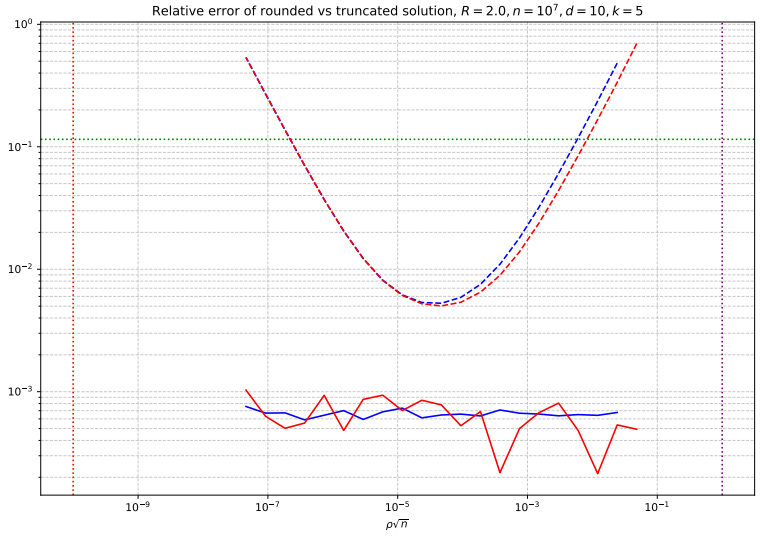
Experimental evaluation: Quantized regression ($\mathrm{b} \notin \mathtt{range}(\mathrm{A}_k)$)

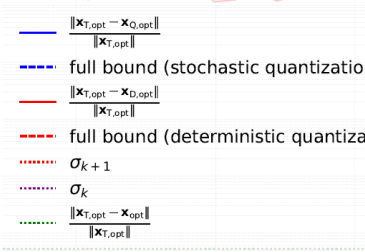




- $n/d = 10^2$
- $\|\mathbf{A}_{k,\perp}\mathbf{A}_{k,\perp}^{\dagger}\mathbf{b}\| \leq \epsilon$

Experimental evaluation: Quantized regression ($\mathrm{b} \notin \mathtt{range}(\mathrm{A}_k)$)





- $n/d = 10^6$
- $\|\mathbf{A}_{k,\perp}\mathbf{A}_{k,\perp}^{\dagger}\mathbf{b}\| \leq \epsilon$

Future work

Theory

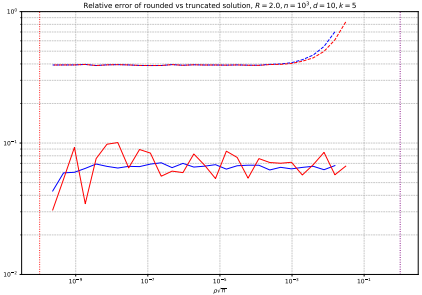
- New Random Matrix Theory bounds for matrices whose entries are independent, but *not* identically distributed random variables.
  1. Can we prove similar bounds for near-square matrices?
  2. Can we figure out the unknown constants?
  3. Can we get rid of the residual terms in our bounds?
- Effect of stochastic rounding in downstream applications[a].

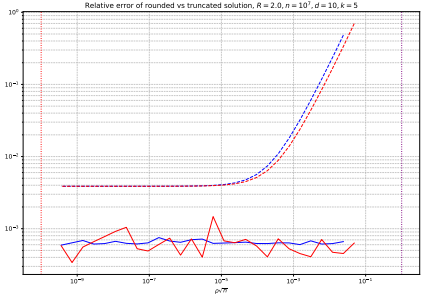
Experiments

- Experimental evaluation in GPUs/IPUs, especially ones that support stochastic rounding, e.g., GraphCore IPU.
- Effect of stochastic rounding in downstream applications[a].

[a] I.e., LS problems, but more importantly optimization and DNN training.

What I am really interested in...





- **In practice,** deterministic rounding should have a similar regularization effect
- But, **theoretically**, deterministic rounding can lead to adverse behavior
- **Some randomness** in rounding will be needed, especially for proofs
- What is the **least amount of randomness** that is necessary to get regularization effects in real applications?

References

C. Boutsikas, P. Drineas, and I. C. F. Ipsen, Small singular values can increase in lower precision, *SIAM Journal on Matrix Analysis and Applications*, 2024.

P. Drineas and I. C. F. Ipsen, Stochastic rounding 2.0, with a view towards complexity analysis, *SIAM News*, Nov 2024.

G. Dexter, C. Boutsikas, L. Ma, I. C. F. Ipsen, and P. Drineas, Stochastic rounding implicitly regularizes tall-and-thin matrices, *SIAM Journal on Matrix Analysis and Applications*, 2025.

L. Ma, Y. Yu, and P. Drineas, Stochastic Rounding increases small singular values, 2026.

L. Ma, C. Boutsikas, and P. Drineas, Stochastic rounding regularizes least-squares regression, 2026.

References

J. V. Neumann and H. H. Goldstine, Numerical inverting of matrices of high order, *Bulletin of the American Mathematical Society*, 1947.

G. E. Forsythe, Round-off errors in numerical integration on automatic machinery, *Bulletin of the American Mathematical Society*, 1950.

G. W. Stewart, A second-order perturbation expansion for small singular values, *Linear Algebra and its Applications*, 1984.

G. W. Stewart and J. G. Sun, Matrix perturbation theory, *Academic Press*, 1990.

S. M. Rump, Inversion of extremely ill-conditioned matrices in floating-point, *Japan Journal of Industrial and Applied Mathematics*, 2009.

S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, Deep learning with limited numerical precision, *International Conference on Machine Learning*, 2015.

M. Croci, M. Fasi, N. J. Higham, T. Mary, and M. Mikaitis, Stochastic rounding: implementation, error analysis and applications, *Royal Society Open Science*, 2022.

I. Dumitriu and Y. Zhu, Extreme singular values of inhomogeneous sparse random rectangular matrices, *Bernoulli*, 2024.