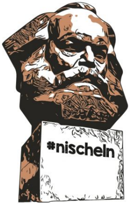
Randomized Low-Rank Approximation
in Finite and Infinite Dimensions

Daniel Kressner

Institute of Mathematics
daniel.kressner@epfl.ch
http://anchp.epfl.ch

EPFL



Workshop and Summer School on Applied Analysis 2023

Randomization in Numerical Linear Algebra...

- ... leads to new and cheap algorithms
- ... turns "statements that hold generically" into quantifiable results and algorithms
- ... replaces expensive components in classical algorithms by cheaper alternatives
- ... offers increased flexibility to exploit structure
- ... regularizes ill-conditioned problems

Randomization in Numerical Linear Algebra...

... leads to new and cheap algorithms

... turns “statements that hold generically” into quantifiable results and algorithms

... replaces expensive components in classical algorithms by cheaper alternatives

... offers increased flexibility to exploit structure

... regularizes ill-conditioned problems

... features prominently on Netflix (The Lincoln Lawyer S1E3, spotted by Petros Drineas)

Thesis? What is it about?

Randomization in NLA

Randomized Numerical Linear Algebra: Surveys

- Murray et al.'2023. Randomized numerical linear algebra. A perspective on the field with an eye to software.
  `https://arxiv.org/abs/2302.11474v2`
- Martinsson/Tropp'2020. Randomized numerical linear algebra: Foundations and algorithms. Acta Numerica.
- Drineas/Mahoney'2018. Lectures on randomized numerical linear algebra. AMS.
- Kannan/Vempala'2017. Randomized algorithms in numerical linear algebra. Acta Numerica.
- Woodruff'2014. Sketching as a tool for numerical linear algebra, Foundations and Trends in Computer Science.
- Halko/Martinsson/Tropp'2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review.

Randomized low-rank approximation
$=$ poster child of randomized NLA.

Rest of these lectures

1. Linear algebra fundamentals
2. Low-rank approximation in finite dimensions
3. Low-rank approximation in infinite dimensions

1. Linear algebra fundamentals

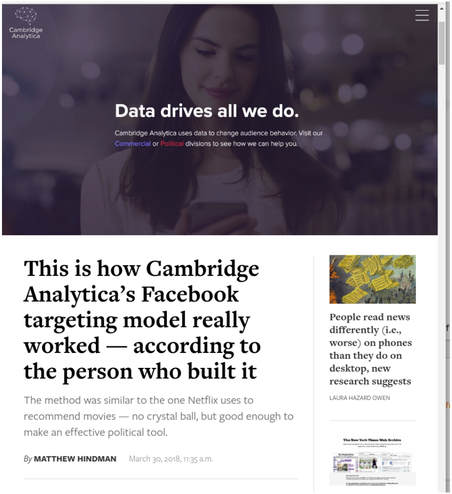
- Matrix rank
- SVD
- Best low-rank approximation

References: [Golub/Van Loan'2013][1], [Horn/Johnson'2013][2]

[1] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins University Press, Baltimore, MD, 2013.

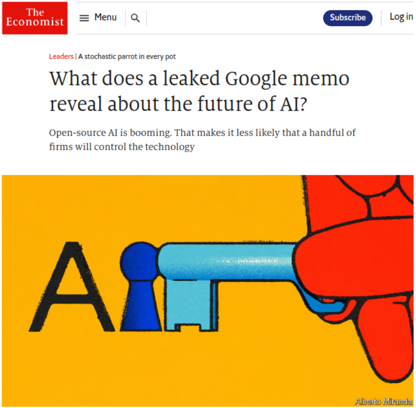
[2] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 2013.

From http://www.niemanlab.org



... his [Aleksandr Kogan's] message went on to confirm that his approach was indeed similar to SVD or other matrix factorization methods, like in the Netflix Prize competition, and the Kosinki-Stillwell-Graepel Facebook model. Dimensionality reduction of Facebook data was the core of his model.

Leaked Internal Google Document, May 2023


The Economist
Menu
Subscribe
Log in

Leaders | A stochastic parrot in every pot

What does a leaked Google memo reveal about the future of AI?

Open-source AI is booming. That makes it less likely that a handful of firms will control the technology

But the uncomfortable truth is, we aren't positioned to win this arms race and neither is OpenAI. While we've been squabbling, a third faction has been quietly eating our lunch... Open-source models are faster, more customizable, more private, and pound-for-pound more capable. They are doing things with \$100 and 13B params that we struggle with at \$10M and 540B. And they are doing so in weeks, not months. ...

In both cases, low-cost public involvement was enabled by a vastly cheaper mechanism for fine tuning called low rank adaptation, or LoRA [arXiv:2106.09685] ...

Rank and basic properties

Let $A \in \mathbb{R}^{m \times n}$. Then

$$\mathrm{rank}(A) := \dim(\mathrm{range}(A)).$$

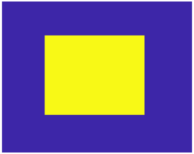
Rank and basic properties

Let $A \in \mathbb{R}^{m \times n}$. Then

$$\operatorname{rank}(A) := \dim(\operatorname{range}(A)).$$

Quiz

1. What is the rank of this matrix?



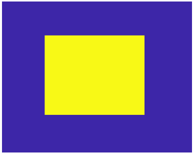
Rank and basic properties

Let $A \in \mathbb{R}^{m \times n}$. Then

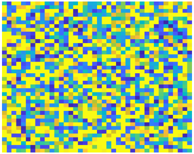
$$\mathrm{rank}(A) := \dim(\mathrm{range}(A)).$$

Quiz

1. What is the rank of this matrix?



2. What is the rank of `randn(40)`?



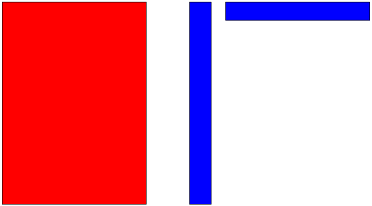
Rank and matrix factorizations

Lemma. A matrix $A \in \mathbb{R}^{m \times n}$ of rank r admits a factorization of the form

$$A = BC^T, \quad B \in \mathbb{R}^{m \times r}, \quad C \in \mathbb{R}^{n \times r}.$$

We say that A has low rank if $\mathrm{rank}(A) \ll m, n$.

Illustration of low-rank factorization:



	A	BC^T
#entries	mn	$mr + nr$

- Generically (and in most applications), A has full rank, that is, $\mathrm{rank}(A) = \min\{m, n\}$.
- Aim instead at approximating A by a low-rank matrix.

The singular value decomposition

Theorem (SVD). Let $A \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then there are orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$A = U\Sigma V^T, \quad \text{with} \quad \Sigma = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & 0 & \end{bmatrix} \in \mathbb{R}^{m\times n}$$

and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.

- $\sigma_1, \ldots, \sigma_n$ are called singular values
- $u_1, \ldots, u_n$ are called *left* singular vectors
- $v_1, \ldots, v_n$ are called *right* singular vectors
- $Av_i = \sigma_i u_i$, $A^T u_i = \sigma_i v_i$ for $i = 1, \ldots, n$.
- Singular values are always uniquely defined by A.
- Singular values are *never* unique. If $\sigma_1 > \sigma_2 > \cdots \sigma_n > 0$ then unique up to $u_i \leftarrow \pm u_i$, $v_i \leftarrow \pm v_i$.

The singular value decomposition

Theorem (SVD). Let $A \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then there are orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$A = U\Sigma V^T, \quad \text{with} \quad \Sigma = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & 0 & \end{bmatrix} \in \mathbb{R}^{m\times n}$$

and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.

Quiz: Which properties of A can be extracted from the SVD?

The singular value decomposition

Theorem (SVD). Let $A \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then there are orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$A = U\Sigma V^T, \quad \text{with} \quad \Sigma = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & 0 & \end{bmatrix} \in \mathbb{R}^{m\times n}$$

and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.

Quiz: Which properties of A can be extracted from the SVD?

$r = \text{rank}(A) =$ number of nonzero singular values of A,

$\mathrm{kernel}(A) = \mathrm{span}\{v_{r+1}, \dots, v_n\}$, $\text{range}(A) = \mathrm{span}\{u_1, \dots, u_r\}$

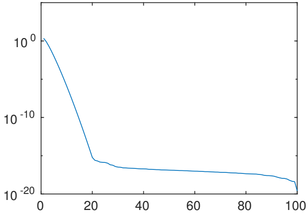
$\|A\|_2 = \sigma_1$, $\|A^\dagger\|_2 = 1/\sigma_r$, $\|A\|_F^2 = \sigma_1^2 + \cdots + \sigma_n^2$

$\sigma_1^2, \dots, \sigma_n^2$ eigenvalues of AA^T and A^TA.

SVD: Computational aspects

- Standard implementations (LAPACK, Matlab's `svd`, ...) require $\mathcal{O}(mn^2)$ operations to compute (economy size) SVD of $m \times n$ matrix A.
- Beware of roundoff error when interpreting singular value plots.

  Example: `semilogy(svd(hilb(100)))`



  - Kink is caused by roundoff error and does not reflect true behavior of singular values.
  - Exact singular values are known to decay exponentially.[3]
  - *Sometimes* more accuracy possible.[4].

[3]Beckermann, B. The condition number of real Vandermonde, Krylov and positive definite Hankel matrices. Numer. Math. 85 (2000), no. 4, 553–577.

[4]Drmač, Z.; Veselić, K. New fast and accurate Jacobi SVD algorithm. I. SIAM J. Matrix Anal. Appl. 29 (2007), no. 4, 1322–1342

Best low-rank approximation

For $k < n$, partition SVD as

$$U\Sigma V^T = \begin{bmatrix} U_k & * \end{bmatrix} \begin{bmatrix} \Sigma_k & 0 \\ 0 & * \end{bmatrix} \begin{bmatrix} V_k & * \end{bmatrix}^T, \quad \Sigma_k = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix}$$

Rank-k truncation:

$$A \approx \mathcal{T}_k(A) := U_k \Sigma_k V_k^T.$$

has rank at most k. By unitary invariance of $\|\cdot\| \in \{\|\cdot\|_2, \|\cdot\|_F\}$:

$$\|\mathcal{T}_k(A) - A\| = \big\|\mathrm{diag}(0, \ldots, 0, \sigma_{k+1}, \ldots, \sigma_n)\big\|.$$

In particular:

$$\|A - \mathcal{T}_k(A)\|_2 = \sigma_{k+1}, \qquad \|A - \mathcal{T}_k(A)\|_F = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_n^2}.$$

Nearly equal iff singular values decay quickly.

Best low-rank approximation

Theorem (Schmidt-Mirsky). Let $A \in \mathbb{R}^{m\times n}$. Then

$$\|A - \mathcal{T}_k(A)\| = \min\left\{\|A - B\| : \ B \in \mathbb{R}^{m\times n} \text{ has rank at most } k\right\}$$

holds for any unitarily invariant norm $\|\cdot\|$.

Proof: See Section 7.4.9 in [Horn/Johnson'2013] for general case.

Proof for $\|\cdot\|_2$: For any $B \in \mathbb{R}^{m\times n}$ of rank $\le k$, $\mathrm{kernel}(B)$ has dimension $\ge n-k$. Hence, $\exists w \in \mathrm{kernel}(B) \cap \mathrm{range}(V_{k+1})$ with $\|w\|_2 = 1$. Then

$$\begin{aligned}
\|A - B\|_2^2 &\ge \|(A-B)w\|_2^2 = \|Aw\|_2^2 = \|AV_{k+1}V_{k+1}^T w\|_2^2 \\
&= \|U_{k+1}\Sigma_{k+1}V_{k+1}^T w\|_2^2 \\
&= \sum_{j=1}^{r+1} \sigma_j |v_j^T w|^2 \ge \sigma_{k+1} \sum_{j=1}^{r+1} |v_j^T w|^2 = \sigma_{k+1}.
\end{aligned}$$

Best low-rank approximation

Theorem (Schmidt-Mirsky). Let $A \in \mathbb{R}^{m \times n}$. Then

$$\|A - \mathcal{T}_k(A)\| = \min\left\{\|A - B\| : \ B \in \mathbb{R}^{m \times n} \text{ has rank at most } k\right\}$$

holds for any unitarily invariant norm $\|\cdot\|$.

Quiz. Is the best rank-k approximation unique if $\sigma_k > \sigma_{k+1}$?

Best low-rank approximation

Theorem (Schmidt-Mirsky). Let $A \in \mathbb{R}^{m\times n}$. Then

$$\|A - \mathcal{T}_k(A)\| = \min\left\{\|A - B\| : \; B \in \mathbb{R}^{m\times n} \text{ has rank at most } k\right\}$$

holds for any unitarily invariant norm $\|\cdot\|$.

Quiz. Is the best rank-k approximation unique if $\sigma_k > \sigma_{k+1}$?

- If $\sigma_k > \sigma_{k+1}$ best rank-k approximation unique wrt $\|\cdot\|_F$.
- Wrt $\|\cdot\|_2$ only unique if $\sigma_{k+1} = 0$. For example, $\mathrm{diag}(2,1,\epsilon)$ with $0 < \epsilon < 1$ has infinitely many best rank-two approximations:

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 2-\epsilon/2 & 0 & 0 \\ 0 & 1-\epsilon/2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 2-\epsilon/3 & 0 & 0 \\ 0 & 1-\epsilon/3 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \ldots.$$

- If $\sigma_k = \sigma_{k+1}$ best rank-k approximation never unique. I_3 has several best rank-two approximations:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Some uses of low-rank approximation

- Data compression.
- Fast solvers for linear systems: Kernel matrices, integral operators, under the hood of sparse direct solvers (MUMPS, PaStiX), . . .
- Fast solvers for dynamical systems: Dynamical low-rank method.
- Low-rank compression / training of neural nets.
- Defeating quantum supremacy claims by Google/IBM. Science'2022:

NEWS | PHYSICS

Ordinary computers can beat Google's quantum computer after all

Superfast algorithm put crimp in 2019 claim that Google's machine had achieved "quantum supremacy"

2 AUG 2022 • 5:05 PM ET • BY ADRIAN CHO

Approximating the range of a matrix

Aim at finding a matrix $Q \in \mathbb{R}^{m\times k}$ with orthonormal columns such that

$$\text{range}(Q) \approx \text{range}(A).$$

QQ^T is orthogonal projector onto range(Q) $\rightsquigarrow$ Aim at minimizing

$$\|A - QQ^T A\|$$

for $\|\cdot\| \in \{\|\cdot\|_2, \|\cdot\|_F\}$. Because rank$(QQ^T A) \le k$,

$$\|A - QQ^T A\| \ge \|A - \mathcal{T}_k(A)\|.$$

Setting $Q = U_k$ one obtains

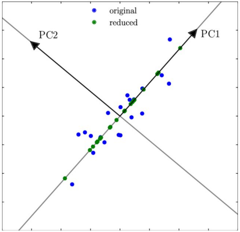
$$U_k U_k^T A = U_k U_k^T U \Sigma V^T = U_k \Sigma_k V_k^T = \mathcal{T}_k(A).$$

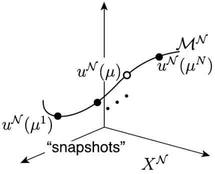
$\rightsquigarrow$ $Q = U_k$ is optimal.

Low-rank approximation and range approximation
are essentially the same tasks!

Two popular uses of range approximation

Principal component analysis (PCA): Dominant left singular vectors of data matrix $X = [x_1, \ldots, x_n]$ (with mean subtracted) provide directions of maximum variance, 2nd maximum variance, etc.





Proper orthogonal decomposition (POD), reduced basis methods: Collect snapshots of time-dependent and/or parameter-dependent equations and perform model reduction by projection to dominant left singular vectors U_k of snapshot matrix.

When to expect good low-rank approximations

Smoothness.

Example 1: Snapshot matrix with snapshots depending smoothly on time/parameter

$$\begin{bmatrix} u(t_1) & u(t_2) & \cdots & u(t_n) \end{bmatrix}$$

$$\approx \underbrace{\begin{bmatrix} p_1 & p_2 & \cdots & p_k \end{bmatrix}}_{\text{low-dim. polynomial basis}} \times \underbrace{\begin{bmatrix} \ell_1(t_1) & \ell_1(t_2) & \cdots & \ell_1(t_n) \\ \ell_2(t_1) & \ell_2(t_2) & \cdots & \ell_2(t_n) \\ \vdots & \vdots & & \vdots \\ \ell_2(t_1) & \ell_2(t_2) & \cdots & \ell_2(t_n) \end{bmatrix}}_{\text{Vandermonde-like matrix}}$$

where $u(t) \approx p(t) = p_1\ell_1(t) + \cdots + p_n\ell_n(t)$ (polynomial approximation of degree k).

When to expect good low-rank approximations

Smoothness.

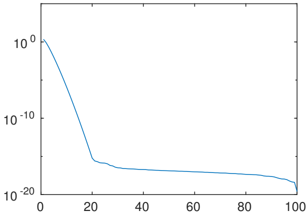
Example 2: Kernel matrix for smooth (low-dimensional) kernel:

$$K = \begin{bmatrix} \kappa(x_1, x_1) & \cdots & \kappa(x_1, x_n) \\ \vdots & & \vdots \\ \kappa(x_n, x_1) & \cdots & \kappa(x_n, x_n) \end{bmatrix}, \quad \kappa : \Omega \times \Omega \to \mathbb{R}.$$

Hilbert matrix:

$$K = \left[\frac{1}{i+j-1}\right]_{i,j=1}^{n}$$

Kernel $\kappa(x, y) = 1/(x + y - 1)$
smooth on $\Omega = [1, n]$



`semilogy(svd(hilb(100)))`

When to expect good low-rank approximations

Algebraic structure.

If X satisfies low-rank Sylvester matrix equation:

$$AX + XB = \text{low rank}$$

and spectra of A, B are disjoint then singular values of X (usually) decay exponentially[5].

- Basis of fast solvers for matrix equations.
- Captures many structured matrices: Vandermonde, Cauchy, Pick, ... matrices.

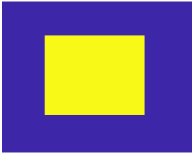
[5] Bernhard Beckermann and Alex Townsend. "On the singular values of matrices with displacement structure". In: *SIAM J. Matrix Anal. Appl.* 38.4 (2017), pp. 1227–1248.

When *not* to expect good low-rank approximations

In most over situations:

- Kernel matrices with singular/non-smooth kernels
- Snapshot matrices for time-dependent / parametrized solutions featuring a slowly decaying Kolmogoroff N-width.
- Images
- White noise
- ...

$\exists$ Exceptions to these rules:



2. Randomized low-rank approximation
(in finite dimensions)

- Randomized SVD / HMT
- Streaming and generalized Nyström
- Beyond Gaussian random matrices
- Learning structured matrices

References: [HMT][6] [Nakatsukasa][7]

---

[6] N. Halko, P. G. Martinsson, and J. A. Tropp. "Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions". In: *SIAM Rev.* 53.2 (2011), pp. 217–288.

[7] Yuji Nakatsukasa. *Fast and stable randomized low-rank matrix approximation*. 2020. arXiv: 2009.11392.

Landscape of low-rank approximation methods

If A is small, say, $m, n = \mathcal{O}(10^2)$:
Don't think twice, compute full SVD.

If A is large or VERY LARGE, choice of method depends on access model:

- Matrix-vector products $y \leftarrow Ax$

  Examples: Explict dense/sparse/data-sparse matrix A. Implicit, e.g., application of A involves a solver: $A = B_{22} - B_{12}B_{11}^{-1}B_{12}$ with large sparse B_{11}.

  Methods: Randomized SVD / HMT, Block Lanczos, Single-vector Lanczos, generalized Nyström.
- Entries $A(i,j), A(:,j), A(i,:)$

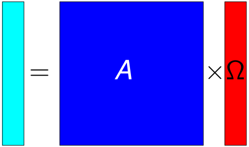
  Examples: Kernel method, distance matrices, boundary element methods.

  Methods: Deterministic sampling (adaptive cross approximation / CUR, Nyström) and randomized sampling.
- (Semi-)analytical techniques: Exponential sum approx, Taylor/polynomial approx, rational approx, random Fourier features.

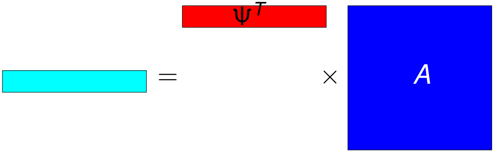
Other BIG DATA / streaming access models exist in TCS literature.

General idea of sketching

1. Use "thin" random matrices Ω, Ψ to create sketches of A:
   - Sketch of columns:



   - Optional sketch of rows:



2. Approximate A from sketch(es).

Gaussian random matrices

Multivariate normal distribution $X \sim \mathcal{N}(\mu, \Sigma)$ with mean $\mu \in \mathbb{R}^n$ and (positive definite) covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$ has density

$$f_X(x) = \frac{1}{\sqrt{(2\pi)^n \det(\Sigma)}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

$X \sim \mathcal{N}(0, I_n)$ is called a Gaussian random vector.

Orthogonal invariance: For an orthogonal matrix Q, QX is again a Gaussian random vector.

A matrix is a Gaussian random matrix if its columns are independent Gaussian random vectors.

Lemma

Let $[V, V_\perp] \in \mathbb{R}^{n \times n}$ be orthogonal and let Ω be an $n \times m$ Gaussian random matrix. Then $V^T\Omega$ and $V_\perp^T\Omega$ are independent Gaussian random matrices.

Sketching a rank-k matrix

If A has rank k then

$$A = U_k \Sigma_k V_k^T \quad \rightsquigarrow \quad A\Omega = U_k \Sigma_k \underbrace{V_k^T \Omega}_{k \times k \text{ Gaussian random}}$$

$V_k^T \Omega$ is invertible almost surely.

Sketching a rank-k matrix

If A has rank k then

$$A = U_k \Sigma_k V_k^T \quad \rightsquigarrow \quad A\Omega = U_k \Sigma_k \underbrace{V_k^T \Omega}_{k \times k \text{ Gaussian random}}$$

$V_k^T \Omega$ is invertible almost surely. Why?

Sketching a rank-k matrix

If A has rank k then

$$A = U_k \Sigma_k V_k^T \quad \rightsquigarrow \quad A\Omega = U_k \Sigma_k \underbrace{V_k^T \Omega}_{k \times k \text{ Gaussian random}}$$

$V_k^T \Omega$ is invertible almost surely. Hence:

- range(A) = range($A\Omega$)
- $A = QQ^T A$, where $Q \in \mathbb{R}^{m \times k}$ is ONB of $A\Omega$

Exact recovery of range of A from sketch.

A first randomized algorithm for low-rank approx

Randomized Algorithm:

1. Draw Gaussian random matrix $\Omega \in \mathbb{R}^{n\times k}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Exact recovery: If A has rank r, we recover $\widehat{A} = A$ with probability 1.

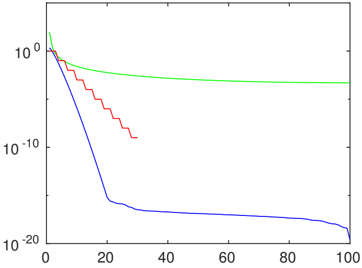
Three test matrices

(a) The 100×100 Hilbert matrix A defined by $A(i,j) = 1/(i+j-1)$.

(b) The matrix A defined by $A(i,j) = \exp(-\gamma|i-j|/n)$ with $\gamma = 0.1$.

(c) 30×30 diagonal matrix with diagonal entries

$$1, 0.99, 0.98, \frac{1}{10}, \frac{0.99}{10}, \frac{0.98}{10}, \frac{1}{100}, \frac{0.99}{100}, \frac{0.98}{100}, \ldots$$



Singular values of test matrices

Randomized algorithm applied to test matrices

errors measured in spectral norm:

(a) Hilbert matrix, $k = 5$:

Exact	mean	std
0.0019	0.0092	0.0099

(b) Matrix with slower decay, $k = 25$:

Exact	mean	std
0.0034	0.012	0.002

(c) Matrix with staircase sv, $k = 7$:

Exact	mean	std
0.010	0.038	0.025

Randomized algorithm applied to test matrices

errors measured in Frobenius norm:

(a) Hilbert matrix, $k = 5$:

Exact	mean	std
0.0019	0.0093	0.0099

(b) Matrix with slower decay, $k = 25$:

Exact	mean	std
0.011	0.024	0.001

(c) Matrix with staircase sv, $k = 7$:

Exact	mean	std
0.014	0.041	0.024

Randomized SVD

Add oversampling. (usually small) integer p

Randomized Algorithm:

1. Draw standard Gaussian random matrix $\Omega \in \mathbb{R}^{n\times(k+p)}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Problem: $\hat{A}$ has rank $k + p > k$.

Solution: Compress $B \approx \mathcal{T}_k(B) \ \rightsquigarrow\ Q\mathcal{T}_k(B)$ has rank k .

Error:

$$
\begin{aligned}
\|Q\mathcal{T}_k(B) - A\| &= \|Q\mathcal{T}_k(B) - QB + QB - A\| \\
&\leq \|\mathcal{T}_k(B) - B\| + \|(I - QQ^T)A\|
\end{aligned}
$$

Randomized SVD applied to test matrices

errors measured in spectral norm:

(a) Hilbert matrix, $k = 5$:

Exact	mean	std	
0.0019	0.0092	0.0099	$p = 0$
0.0019	0.0026	0.0019	$p = 1$
0.0019	0.0019	0.0001	$p = 2$

(b) Matrix with slower decay, $k = 25$:

Exact	mean	std	
0.0034	0.012	0.002	$p = 0$
0.0034	0.011	0.0017	$p = 1$
0.0034	0.010	0.0015	$p = 2$
0.0034	0.0064	0.0008	$p = 10$
0.0034	0.0037	0.0002	$p = 25$

(c) Matrix with staircase sv, $k = 7$:

Exact	mean	std	
0.010	0.038	0.025	$p = 0$
0.010	0.021	0.012	$p = 1$
0.010	0.012	0.005	$p = 2$

Analysis: general considerations

Goal: Say something sensible about $\|(I - QQ^T)A\|$. Expected value, failure bounds, ... wrt random matrix Ω.

Often, analysis of randomized NLA can be separated into two phases

1. Structural bound: Derive bound that holds for (almost) *every* Ω.

   This bound usually depends on Ω and dependence needs to be simple enough to facilitate 2nd phase.

2. Stochastic analysis: Derive expected value, failure bounds for structural bound using random matrix theory, concentration results, ...

Analysis: structural bound

Goal: Bound $\|(I - \Pi_{A\Omega})A\|_F$, where $\Pi_{A\Omega} = QQ^T$ is orthogonal projector onto range of $A\Omega$.

Analysis: structural bound

Goal: Bound $\|(I - \Pi_{A\Omega})A\|_F$, where $\Pi_{A\Omega} = QQ^T$ is orthogonal projector onto range of $A\Omega$.

Problems: Implicit dependence on Ω, relation to SVD?

Important observation: Because of

$$(I - \Pi_{A\Omega})A\Omega = 0,$$

the *oblique* projector $\tilde{\Pi} = \Omega(V_k^T\Omega)^\dagger V_k^T$ satisfies

$$\begin{aligned}
\|(I - \Pi_{A\Omega})A\|_F &= \|(I - \Pi_{A\Omega})A(I - \tilde{\Pi})\|_F \\
&\leq \|A(I - \tilde{\Pi})\|_F \\
&\leq \|A(I - V_k V_k^T)(I - \tilde{\Pi})\|_F,
\end{aligned}$$

where we used

$$(I - V_k V_k^T)(I - \tilde{\Pi}) = (I - \tilde{\Pi}).$$

in the last step.

Analysis: structural bound

$$\|(I - \Pi_{A\Omega})A\|_F \leq \|A(I - V_k V_k^T)(I - \tilde{\Pi})\|_F$$

Interpretation: "Gold standard" $A(I - V_k V_k^T)$ distored by oblique projection.

Quick but suboptimal argument:

$$\|A(I - VV^T)(I - \tilde{\Pi}^T)\|_F \leq \|A(I - VV^T)\|_F \|I - \tilde{\Pi}\|_2 = \|\Sigma_2\|_F \|\tilde{\Pi}\|_2$$

Deviation from gold standard $\|\Sigma_2\|_F$ determined by $\|\tilde{\Pi}\|_2 \leq \|(\Omega^T V)^\dagger\|_2 \|\Omega\|_2$.
Drawback: Involves big matrix Ω, which will lead to suboptimal constants for Gaussian random matrices.
Quiz: We used $\|I - \tilde{\Pi}\|_2 = \|\Pi\|_2$; how does one prove this relation?

Analysis: structural bound

More refined argument:

$$\begin{aligned}\|A(I - V_k V_k^T)(I - \tilde{\Pi})\|_F^2 &= \|A(I - V_k V_k^T)\|_F^2 + \|A(I - V_k V_k^T)\tilde{\Pi}\|_F^2 \\ &= \|\Sigma_2\|_F^2 + \|\Sigma_2(V_\perp^T \Omega)(V_k^T \Omega)^\dagger\|_F^2\end{aligned}$$

Final structural bound:

$$\|(I - QQ^T)A\|_F^2 \leq \|\Sigma_2\|_F^2 + \|\Sigma_2 \Omega_2 \Omega_1^\dagger\|_F^2.$$

with $\Omega_1 = V_k^T \Omega$ and $\Omega_2 = V_\perp^T \Omega$.

Bounding expectation

Goal: Bound expected value of

$$\|(I - QQ^T)A\|_F^2 \leq \|\Sigma_2\|_F^2 + \|\Sigma_2\Omega_2\Omega_1^\dagger\|_F$$

for independent Gaussian random matrices Ω_1, Ω_2.

To analyze red term, we use

$$\mathbb{E}\|\Sigma_2\Omega_2\Omega_1^\dagger\|_F^2 = \mathbb{E}\big(\mathbb{E}\big(\|\Sigma_2\Omega_2\Omega_1^\dagger\|_F^2 \,|\, \Omega_1\big)\big) = \|\Sigma_2\|_F^2 \cdot \mathbb{E}\|\Omega_1^\dagger\|_F^2.$$

(See exercises for proof that $\mathbb{E}\|A\Omega B\|_F^2 = \|A\|_F^2\|B\|_F^2$ for Gaussian matrix Ω and constant matrices A, B.)

Analysis: $k = 1, p = 0$

For $k = 1$, $p = 0$, we have

$$(V_1^T \Omega)^\dagger = \omega_1^{-1}, \quad \omega_1 \sim \mathcal{N}(0, 1).$$

Problem: ω_1^{-1} (reciprocal of standard normal random variable) is Cauchy distribution with undefined mean and variance.

Need to consider $p \geq 2$.

Analysis: $k = 1, p \geq 2$

For $k = 1$ we have $\|\Omega_1^\dagger\|_F^2 = 1/\|\Omega_1\|_F^2$, where $\|\Omega_1\|_F^2$ is a sum of $p+1$ squared independent standard normal random variables.

Pdf for $X \sim \mathcal{N}(0,1)$ given by $f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$. Pdf for $Y = X^2$ zero for nonpositive values. For $y > 0$, we obtain

$$\begin{aligned} \Pr(0 \leq Y \leq y) &= \Pr(-\sqrt{y} \leq X \leq \sqrt{y}) \\ &= \frac{2}{\sqrt{2\pi}} \int_0^{\sqrt{y}} e^{-x^2/2}\,\mathrm{d}x \\ &= \frac{1}{\sqrt{2\pi}} \int_0^{y} e^{-t/2}\,\mathrm{d}t, \end{aligned}$$

Y is called chi-squared distribution (1 degree of freedom): $Y \sim \chi_1^2$.

$\|\Omega_1\|_F^2 \sim \chi_{p+1}^2$ chi-squared distribution with $p+1$ d.o.f.; pdf

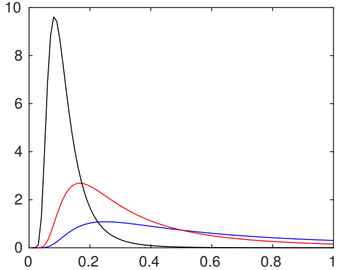
$$f_{\Omega_1}(x) = \frac{2^{-(p+1)/2}}{\Gamma((p+1)/2)} x^{(p+1)/2-1} \exp(-x/2)), \quad x > 0.$$

Analysis: $k = 1, p \geq 2$

$$\|\Omega_1^\dagger\|_F^2 = \frac{1}{\|\Omega_1\|_F^2} = \Big(\sum_{i=1}^{p} \Omega_{1,i}^2\Big)^{-1} \sim \mathrm{Inv} - \chi^2(p+1),$$

the inverse-chi-squared distribution with $p+1$ degrees of freedom.
Pdf given by

$$\frac{2^{-(p+1)/2}}{\Gamma((p+1)/2)} x^{-(p+1)/2-1} \exp(-1/(2x)).$$



pdf for $p = 1$, $p = 3$, $p = 9$

Analysis: $k = 1, p \geq 2$

Textbook results:

- $\mathbb{E}\|\Omega_1\|_F^2 = p + 1, \quad \mathbb{E}\|\Omega_1^\dagger\|_F^2 = (p-1)^{-1}$

Tail bound by [Laurent/Massart'2000]:

- $\mathbb{P}\big[\|\Omega_1\|_F^2 \leq p + 1 - t\big] \leq \exp\left(-\frac{t^2}{4(p+1)}\right)$

Theorem

For $k = 1, p \geq 2$, we have

$$\mathbb{E}\|(I - QQ^T)A\|_F \leq \sqrt{1 + \frac{1}{p-1}}\|\Sigma_2\|_F.$$

Probability of deviating from this upper bound decays exponentially, as *indicated* by tail bound for χ^2_{p+1}.

Analysis: general $k, p \geq 2$

Again use

$$\mathbb{E}\|\Sigma_2\Omega_2\Omega_1^\dagger\|_F^2 = \|\Sigma_2\|_F^2 \cdot \mathbb{E}\|\Omega_1^\dagger\|_F^2.$$

By standard results in multivariate statistics, we have

$$\mathbb{E}\|\Omega_1^\dagger\|_F^2 = \frac{k}{p-1}.$$

Sketch of argument:

- $\Omega_1\Omega_1^T \sim W_k(k+p)$ (Wishart distribution with $k+p$ degrees of freedom)
- $(\Omega_1\Omega_1^T)^{-1} \sim \mathcal{W}_k^{-1}(k+p)$ (inverse Wishart distribution with $r+p$ degrees of freedom)
- $\mathbb{E}(\Omega_1\Omega_1^T)^{-1} = \frac{1}{k-1} I_k$; see Page 96 in [Muirhead'1982][8]
- Result follows from $\|\Omega_1^\dagger\|_F^2 = \|\Omega_1^T(\Omega_1\Omega_1^T)^{-1}\|_F^2 = \operatorname{trace}\big((\Omega_1\Omega_1^T)^{-1}\big)$

[8] R. J. Muirhead, Aspects of Multivariate Statistical Theory, Wiley, New York, NY, 1982.

Analysis: general k, $p \geq 2$

Together with $\mathbb{E}\|(I - QQ^T)A\|_F \leq \sqrt{\mathbb{E}\|(I - QQ^T)A\|_F^2}$, we obtain:

Theorem

For $p \geq 2$, we have

$$\mathbb{E}\|(I - QQ^T)A\|_F \leq \sqrt{1 + \frac{k}{p-1}}\|\Sigma_2\|_F,$$

$$\mathbb{E}\|(I - QQ^T)A\|_2 \leq \left(1 + \sqrt{\frac{k}{p-1}}\right)\|\Sigma_2\|_2 + \frac{e\sqrt{k+p}}{p}\|\Sigma_2\|_F.$$

For proof of spectral norm and tail bounds, see [HMT].

Variations on the randomized SVD

- Streaming and generalized Nyström
- Beyond Gaussian random matrices
- Learning structured matrices

Variation 1: Streaming

Motivation of streaming models:

Matrix/data arrives in chunks.
Each chunk should be processed cheaply.
Avoid storing the matrix as whole.

Examples:

- Incremental POD for high-dimensional differential equations.[9]
- PCA for massive data.
- Repeated localized / low-rank modifications of data matrix.

All captured by

$$A \to A_0 + A_1 + A_2 + \cdots$$

Assumption: Cheap to perform sketches of each A_k.

Goal: Design (randomized) method suitable for streamed data.

[9] J. A. Tropp et al. "Streaming Low-Rank Matrix Approximation with an Application to Scientific Simulation". In: *SIAM J. Sci. Comput.* 41.4 (Jan. 2019), A2430–A2463.

Variation 1: Streaming

Randomized SVD:

1. Draw standard Gaussian random matrix $\Omega \in \mathbb{R}^{n\times(k+p)}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Not suitable for streaming. Why?

Variation 1: Streaming

Randomized SVD:

1. Draw standard Gaussian random matrix $\Omega \in \mathbb{R}^{n\times(k+p)}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Idea:

- QQ^T is best/orthogonal projection of cols of A onto range($A\Omega$) $\rightsquigarrow$ needs to be relaxed.
- Consider any $\Psi \in \mathbb{R}^{m\times k+p+\ell}$ with $\ell \geq 2$ such that $\Psi^T A\Omega$ has (full) rank $k+p$. Then

$$\Pi_{\Omega,\Psi} := (A\Omega)(\Psi^T A\Omega)^\dagger \Psi^T A$$

is (oblique) projector onto range($A\Omega$).

Variation 1: Streaming

Generalized Nyström = algorithm for constructing approximation $\widehat{A} = \Pi_{\Omega,\Psi} A = (A\Omega)(\Psi^T A\Omega)^\dagger \Psi^T A$:

1. Draw independent Gaussian random matrices $\Omega \in \mathbb{R}^{n\times(k+p)}$, $\Psi \in \mathbb{R}^{n\times k+p+\ell}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Perform block mat-vec $W = A^T\Psi$.
4. Compute $S = W^T\Omega$ and $\tilde{Y} = YS^\dagger$ (via QR or SVD of S, possibly regularized [Nakatsukasa]).
5. Return $\widehat{A} = YW^T$ in factored form.

- Steps 2 and 3 *linear* in A and thus well suited for streaming model:

$$\begin{aligned} Y &= (A_0 + A_1 + \cdots)\Omega = A_0\Omega + A_1\Omega + \cdots \\ W &= (A_0 + A_1 + \cdots)^T\Psi = A_0^T\Psi + A_1^T\Psi + \cdots . \end{aligned}$$

  Only compute $A_j\Omega$, $A_j^T\Psi$ (cheap) and update Y, W in jth step.
  No storage of $A_j\Omega$, $A_j^T\Psi$ or A needed.
- Step 4 is not linear in A/not streaming, but it is cheap.

Variation 1: Streaming

Analysis of streaming [Tropp et al.'2019, Nakatsukasa]:

$$
\begin{aligned}
\|A-\widehat{A}\|_F^2 &= \|A-\Pi_{\Omega,\Psi}A\|_F^2 = \overbrace{\|A-QQ^TA\|_F^2}^{\text{Rand. SVD}} + \overbrace{\|QQ^TA-\Pi_{\Omega,\Psi}A\|_F^2}^{\text{Distortion of proj.}} \\
&= \cdots \le \|A-QQ^TA\|_F^2 + \|(\Psi^TQ)^\dagger(\Psi^TQ_\perp)Q_\perp^TA\|_F^2
\end{aligned}
$$

Using

$$
\begin{aligned}
& \mathbb{E}_{\Omega,\Psi}\|(\Psi^TQ)^\dagger(\Psi^TQ_\perp)Q_\perp^TA\|_F^2 \\
= \; & \mathbb{E}_\Omega\left[\mathbb{E}_\Psi\left[\|(\Psi^TQ)^\dagger(\Psi^TQ_\perp)Q_\perp^TA\|_F^2 \mid \Omega\right]\right] \\
\le \; & \left(1+\frac{k+p}{\ell-1}\right)\mathbb{E}_\Omega\left[\|Q_\perp^TA\|_F^2\right]
\end{aligned}
$$

In summary:

$$
\mathbb{E}\|A-\widehat{A}\|_F \le \sqrt{1+\frac{k}{p-1}}\sqrt{1+\frac{k+p}{\ell-1}}\|\Sigma_2\|_F.
$$

Variation 1: Streaming

- Streaming algorithms useful in the context of compressing structured tensors in Tucker format [Sun et al.'2019] and TT format [Daas et al.'2021, Shi et al.'2021, Ma/Solomonik'2022, Kressner et al.'2023]
- If A is symmetric positive definite, choose $\Psi = \Omega \rightsquigarrow$ approximation
$$\widehat{A} = (A\Omega)(\Omega^T A\Omega)^\dagger \Omega^T A$$
This saves half of the matrix multiplications!
Analysis more difficult.[10]

[10] A. Gittens and M. W. Mahoney. "Revisiting the Nyström method for improved large-scale machine learning". In: *J. Mach. Learn. Res.* 17 (2016).

Variation 2: Beyond Gaussian random matrices

Johnson–Lindenstrauss lemma: N points in $\mathbb{R}^n$ can be embedded (by linear projection) into a subspace of dimension $\mathcal{O}(\varepsilon^{-2}\log N)$ such that distances are preserved up to factor $1 \pm \varepsilon$.

Scaled Gaussian random matrices produce such embeddings $x \mapsto \Omega^T x$ with high probability. More generally:

JL property. A distribution over $\mathbb{R}^{n\times\ell}$ has the (ε,δ)-JL property if a random matrix Ω satisfies

$$\mathbb{P}\big(\big|\|\Omega^T x\|_2^2 - 1\big| > \varepsilon\big) < \delta$$

for an *arbitrary but fixed* $x \in \mathbb{R}^n$, $\|x\|_2 = 1$.

- A Gaussian random matrix (divided by $\sqrt{\ell}$) has the JL property when $\ell = \mathcal{O}(\varepsilon^{-2}\log(1/\delta))$.
- JL lemma is obtained from union bound: To preserve N^2 pairwise distances $\|x_i - x_j\|_2$ use $(\varepsilon, \delta/N^2)$ JL-property $\rightsquigarrow$ $\ell = O\big(\varepsilon^{-2}(\log N + \log(1/\delta))\big)$

Variation 2: Beyond Gaussian random matrices

JL property. An $n \times \ell$ random matrix Ω has the (ε, δ)-JL property if

$$\mathbb{P}\big(\big|\|\Omega^T x\|_2^2 - 1\big| > \varepsilon\big) < \delta$$

for an *arbitrary but fixed* $x \in \mathbb{R}^n$, $\|x\|_2 = 1$.

Generalization to subspaces:

Obvlious subspace embedding (OSE) property [Sarlos'2006]. An $n \times \ell$ random matrix Ω has the (k, ε, δ)-OSE property if

$$\mathbb{P}\big(\big|\|\Omega^T x\|_2^2 - 1\big| > \varepsilon\big) < \delta, \quad \forall x \in \mathcal{V},$$

for an *arbitrary but fixed* k-dimensional subspace $\mathcal{V} \subset \mathbb{R}^n$.

JL property $\rightarrow$ OSE property: Given ONB V of $\mathcal{V}$, OSE is equivalent to

$$y^T(\Omega^T V)^T \Omega^T V y \approx 1, \quad \forall y \in \mathbb{R}^k, \|y\|_2 = 1,$$

It is "enough" to test with 2^{100k} vectors on the unit sphere in order to capture norm of a matrix within factor 4. Union bound: $(\varepsilon/4, \delta/2^{100k})$-JL turns into (k, ε, δ)-OSE.

Gaussian random matrices: $\ell = O\big(\varepsilon^{-2}(k + \log(1/\delta)\big)$ gives OSE.

Variation 2: Beyond Gaussian random matrices

OSE property. An $n \times \ell$ random matrix Ω has the (k, ε, δ)-OSE property if

$$\mathbb{P}\Big(\big|\|\Omega^T x\|_2^2 - 1\big| > \varepsilon\Big) < \delta, \quad \forall x \in \mathcal{V},$$

for an *arbitrary but fixed* k-dimensional subspace $\mathcal{V} \subset \mathbb{R}^n$.

Given ONB V of $\mathcal{V}$, OSE implies

$$\big\|(\Omega^T V)^\dagger\big\|_2 = \frac{1}{\sigma_{\min}(\Omega^T V)} = \frac{1}{\min\{\|\Omega^T x : \|x\|_2 = 1, x \in \mathcal{V}\}} \leq \frac{1}{1-\varepsilon}.$$

Recall structural bound for randomized SVD:

$$\|(I - QQ^T)A\|_F^2 \leq (1 + \|(V_k^T\Omega)^\dagger\|_2^2 \|V_\perp^T\Omega\|_2^2)\|\Sigma_2\|_F^2,$$

where V_k contains k dominant right singular vectors of A.

$\|(V_k^T\Omega)^\dagger\|_2^2$ controlled through OSE (with, say, $\varepsilon = 1/2$, while $\|V_\perp^T\Omega\|_2 \leq \|\Omega\|_2$ is usually bounded (except for Gaussian).

Variation 2: Beyond Gaussian random matrices

Examples:

- (scaled) Rademacher matrices
  $= n \times \ell$ matrices with iid ± 1 (50%/50%) entries.
  OSE holds[11] for $\ell = \mathcal{O}\big(k + \log(1/\delta)\big)$
- SRHT $=$ sub-sampled randomized Hadamard transform
  $\Omega = \sqrt{n/\ell} DHR$, where
  $D =$ diagonal with Rademacher diagonal entries
  $R = n \times \ell$ uniform random sampling matrix
  $H = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
  (zero padding if n is not a power of 2)
  OSE holds for $\ell = \mathcal{O}\big(k \log(1/\delta) \log(n/\delta)\big)$
  [Boutsidis/Gittens'2013]
- Subsampled Fourier transform.
  OSE holds for $\ell = \mathcal{O}\big((\sqrt{k} + \sqrt{\log(kn)})^2 \log k\big)$ with probability $\geq 1 - 1/k$ [HMT]

[11] Generally true for all matrices with columns from sub-Gaussian distribution.

Variation 2: Beyond Gaussian random matrices

- Sparse transforms

  One nonzero entry per row in Ω:
  OSE holds for $\ell = \mathcal{O}(k^2)$ with prob. $> 2/3$.
  [Nelson/Nguyen'2013].

  $\mathcal{O}(\log(k/\delta))$ entries per row in Ω:
  OSE holds for $\ell = \mathcal{O}(k \log(k/\delta)$. [Cohen'2016].
- TensorSketch
- CountSketch
- ...

Many of these embeddings become computationally advantageous over Gaussian random matrices iff k is sufficiently large.

Variation 3: Learning structured matrices

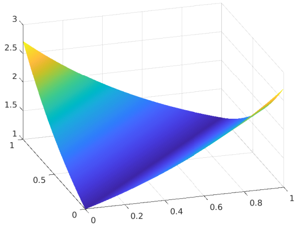
Motivation: Consider kernel matrix

$$K = \begin{bmatrix} \kappa(x_1, x_1) & \cdots & \kappa(x_1, x_n) \\ \vdots & & \vdots \\ \kappa(x_n, x_1) & \cdots & \kappa(x_n, x_n) \end{bmatrix}, \quad \kappa : D \times D \to \mathbb{R}.$$

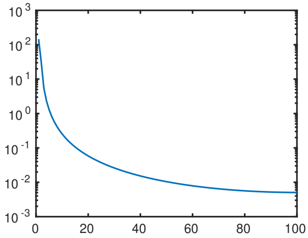
for 1D-kernel κ with diagonal singularity/non-smoothness. Example:

$$\kappa(x, y) = \exp(-|x - y|), \quad x, y \in [0, 1]$$

Function



Singular values

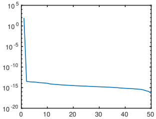


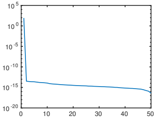
Variation 3: Learning structured matrices

Block partition K:

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} =$$

K_{11}



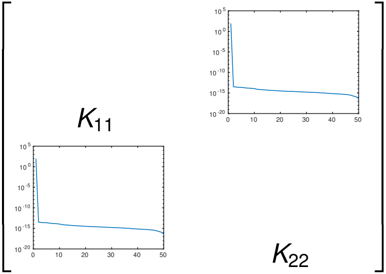


K_{22}

Variation 3: Learning structured matrices

Block partition K:

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} =$$



Basic idea of *peeling method* [Lin/Lu/Ying'2011]: Off-diagonal blocks can be "learnt" from

$$K \begin{bmatrix} \Omega_1 & 0 \\ 0 & \Omega_2 \end{bmatrix} = \begin{bmatrix} \star & K_{12}\Omega_2 \\ K_{21}\Omega_1 & \star \end{bmatrix}$$

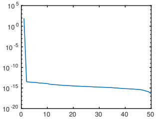
Compute QR decompositions

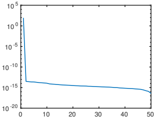
$$K_{12}\Omega_2 = Q_1 R_1, \quad K_{21}\Omega_1 = Q_2 R_2.$$

Variation 3: Learning structured matrices

Block partition K:

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} = \begin{bmatrix} K_{11} & \\ & K_{22} \end{bmatrix}$$





Compute

$$\begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}^T K = \begin{bmatrix} \star & Q_1^T K_{12} \\ Q_2^T K_{21} & \star \end{bmatrix}$$

Level 1 of peeling: Use randomized SVD to approximate off-diagonal blocks:

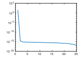
$$K_1 = \begin{bmatrix} 0 & Q_1 Q_1^T K_{12} \\ Q_2 Q_2^T K_{21} & 0 \end{bmatrix}$$

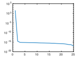
Variation 3: Learning structured matrices

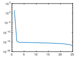
Level 2: Partition diagonal blocks of remainder:

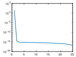
$$K - K_1 \approx \begin{bmatrix} \begin{matrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{matrix} & 0 \\ 0 & \begin{matrix} K_{33} & K_{34} \\ K_{43} & K_{44} \end{matrix} \end{bmatrix}$$

$$= \begin{bmatrix} \begin{matrix} K_{11} & \\ & K_{22} \end{matrix} & 0 \\ 0 & \begin{matrix} K_{33} & \\ & K_{44} \end{matrix} \end{bmatrix}$$









Variation 3: Learning structured matrices

Level 2:

$$(K - K_1) \begin{bmatrix} \Omega_1 & 0 \\ 0 & \Omega_2 \\ \Omega_3 & 0 \\ 0 & \Omega_4 \end{bmatrix} = \begin{bmatrix} \star & K_{12}\Omega_2 \\ K_{21}\Omega_1 & \star \\ \star & K_{34}\Omega_4 \\ K_{43}\Omega_3 & \star \end{bmatrix}$$

Use 4 randomized SVDs to reconstruct off-diagonal blocks on Level 2 $\leadsto K_2$.

Level 3 considers $K - K_1 - K_2$, etc.

- If every off-diagonal block on every level admits good rank-k approximation $\leadsto$ Recovery from $\mathcal{O}(k \log n)$ matrix-vector products.
- K is approximated in the HODLR format, one of the simplest hierarchical matrix formats.

Variation 3: Learning structured matrices

During the last years, several extensions/improvements:

- General $\mathcal{H}$-matrices $=$ general recursive block partition.
- HSS/$\mathcal{H}^2$-matrices impose additional nestedness conditions on the low-rank factors on different levels of the recursion and can be reconstructed with $\mathcal{O}(k)$ matrix-vector products.

Most recent developments:

- D. Halikias and A. Townsend. Structured matrix recovery from matrix-vector products. arXiv:2212.09841, (2022).
- J. Levitt and P.-G. Martinsson. Linear-complexity black-box randomized compression of rank-structured matrices, arXiv:2205.02990, (2022).

3. Randomized low-rank approximation

(in infinite dimensions)

Primary reference: [Boullé/Townsend][12]

[12] Nicolas Boullé and Alex Townsend. “Learning elliptic partial differential equations with randomized linear algebra”. In: *Found. Comput. Math.* (2022), pp. 1–31.

Infinite randomized SVD?

First step of randomized SVD applied to $A \in \mathbb{R}^{m \times n}$:

$$Y = A\Omega, \quad \Omega \text{ is } n \times k \text{ Gaussian random matrix.}$$

What is a suitable extension to a (Hilbert-Schmidt) operator $\mathcal{A} : H_1 \to H_2$ for infinite-dimensional Hilbert spaces H_1, H_2?

Example: Integral operator $\mathcal{A} : L^2(D_y) \to L^2(D_x)$ with

$$(\mathcal{A}f)(x) = \int_{D_y} g(x, y) f(y) \,\mathrm{d}y, \quad x \in D_x,$$

for some kernel $g \in L^2(D_x \times D_y)$.

Goal:

Learn $\mathcal{A}$ from applying it to a few "random" f.

Infinite randomized SVD?

First step of randomized SVD applied to $A \in \mathbb{R}^{m \times n}$:

$$Y = A\Omega, \quad \Omega \text{ is } n \times k \text{ Gaussian random matrix.}$$

What is a suitable extension to a (Hilbert-Schmidt) operator $\mathcal{A} : H_1 \to H_2$ for infinite-dimensional Hilbert spaces H_1, H_2?

Example: Integral operator $\mathcal{A} : L^2(D_y) \to L^2(D_x)$ with

$$(\mathcal{A}f)(x) = \int_{D_y} g(x, y) f(y) \, \mathrm{d}y, \quad x \in D_x,$$

for some kernel $g \in L^2(D_x \times D_y)$.

Goal:

Learn $\mathcal{A}$ from applying it to a few "random" f.

Proposal by [Boullé/Townsend]: Choose samples from Gaussian processes with prescribed regularity.

Preliminaries: HS operators

Assume that $\mathcal{A} : L^2(D_y) \to L^2(D_x)$ is Hilbert-Schmidt (HS), that is, for any ONB $\{e_i\}_{i=1}^{\infty}$ of $L^2(D_y)$ one has

$$\|\mathcal{A}\|_{\mathrm{HS}} := \left(\sum_i \|\mathcal{A}e_i\|_{L^2(D_x)} \right)^{1/2} < \infty.$$

Most important property: HS operators admit SVD. $\exists$ ONB $\{u_i\}_{i=1}^{\infty}$ of $L^2(D_x)$ and $\{v_i\}_{i=1}^{\infty}$ of $L^2(D_y)$ such that

$$\mathcal{A} = \sum_{i=1}^{\infty} \sigma_i u_i \langle v_i, \cdot \rangle_{L^2(D_y)}, \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq 0.$$

Implies that $\|\mathcal{A}\|_{\mathrm{HS}}^2 = \sigma_1^2 + \sigma_2^2 + \cdots$ and

$$\mathcal{T}_k(\mathcal{A}) := \sum_{i=1}^{k} \sigma_i u_i \langle v_i, \cdot \rangle_{L^2(D_y)}, \quad \|\mathcal{A} - \mathcal{T}_k(\mathcal{A})\|_{\mathrm{HS}}^2 = \sigma_{k+1}^2 + \sigma_{k+2}^2 + \cdots$$

is best rank-k approximation (gold standard).

Preliminaries: Gaussian processes

For symm. pos. def. $K \in \mathbb{R}^{n \times n}$, let $\mathcal{N}(0, K)$ denote multivariante normal distribution with zero mean and covariance matrix K.

Infinite-dimensional analogue: Stochastic process $F := \{F_t, t \in D\}$ is Gaussian if $(F_{t_1}, \ldots, F_{t_n})$ is multivariate Gaussian for every finite set of indices $t_1, \ldots, t_n \in D$.

Preliminaries: Gaussian processes

For symm. pos. def. $K \in \mathbb{R}^{n\times n}$, let $\mathcal{N}(0, K)$ denote multivariante normal distribution with zero mean and covariance matrix K.

Infinite-dimensional analogue: Stochastic process $F := \{F_t, t \in D\}$ is Gaussian if $(F_{t_1}, \dots, F_{t_n})$ is multivariate Gaussian for every finite set of indices $t_1, \dots, t_n \in D$.

Specific setting: Given *continuous* symm. pos. def. kernel $\kappa : D \times D \to \mathbb{R}$, suppose that $(F_{t_1}, \dots, F_{t_n})$ is multivariate Gaussian with zero mean and covariance matrix

$$(K)_{ij} = \kappa(t_i, t_j), \quad i, j = 1, \dots, n.$$

Corresponding integral operator $\mathcal{K} : L^2(D) \to L^2(D)$ admits spectral decomposition ($\Leftrightarrow$ Mercer representation of kernel):

$$\mathcal{K}(v(\cdot)) := \int_D \kappa(\cdot, y) v(y)\, \mathrm{d}y = \sum_{i=1}^{\infty} \lambda_i \langle \psi_i, v\rangle \psi_i(\cdot),$$

with orthon. eigenfunctions ψ_i and eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$.

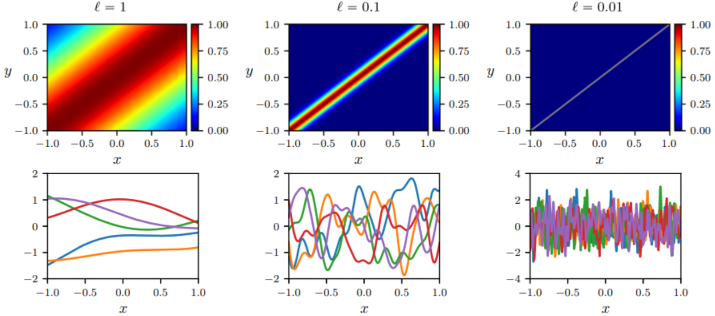
Preliminaries: Gaussian processes

Diagonalization of $\mathcal{K}$ implies Karhune-Loève expansion of stochastic field

$$F_t = \sum_{i=1}^{\infty} \lambda_i X_i \psi_i(t), \quad X_i \sim \mathcal{N}(0,1) \text{ iid.}$$

Decay of λ_i ~ smoothness of κ ~ characterization of regularity of F.

Popular: Squared-exp. $\kappa(x,y) = \exp(-|x-y|^2/(2\ell)^2)$ for $D = [-1,1]$



Kernel and samples for different ℓ (Picture taken from [BT]).

Other popular choice: Matérn kernel.

Preliminaries: Gaussian processes

Diagonalization of $\mathcal{K}$ implies Karhune-Loève expansion of stochastic field

$$F_t = \sum_{i=1}^{\infty} \lambda_i X_i \psi_i(t), \quad X_i \sim \mathcal{N}(0,1)\,\text{iid.}$$

Decay of $\lambda_i \sim$ smoothness of $\kappa \sim$ characterization of regularity of F.

To (approximately) sample from F_t: Consider truncated KL expansion

$$\sum_{i=1}^{m} \lambda_i X_i \psi_i(t), \quad X_i \sim \mathcal{N}(0,1)\,\text{iid}$$

$+$ finite element / spectral discretization in space.

Prescribe KL expansion: functions (polynomials) ψ_i and eigenvalues λ_i instead of κ to impose smoothness.

Randomized SVD $\rightarrow$ Hilbert-Schmidt operators

1. Draw standard Gaussian random matrix $\Omega \in \mathbb{R}^{n\times(k+p)}$.
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Line 1 replaced by
Sample $f_1, \ldots, f_{k+p} \sim F$ (Gaussian process).

Randomized SVD $\rightarrow$ Hilbert-Schmidt operators

1. Sample $f_1, \ldots, f_{k+p} \sim F$ (Gaussian process).
2. Perform block mat-vec $Y = A\Omega$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Line 2 replaced by
Apply operator: $h_1 = \mathcal{A}(f_1), \ldots, h_{k+p} = \mathcal{A}(f_{k+p})$.

Randomized SVD $\rightarrow$ Hilbert-Schmidt operators

1. Sample $f_1, \ldots, f_{k+p} \sim F$ (Gaussian process).
2. Apply operator: $h_1 = \mathcal{A}(f_1), \ldots, h_{k+p} = \mathcal{A}(f_{k+p})$.
3. Compute (economic) QR decomposition $Y = QR$.
4. Form $B = Q^T A$.
5. Return $\widehat{A} = QB$ (in factorized form)

Lines 3–5 replaced by
Return $\Pi_H \mathcal{A}$, where Π_H is orthogonal projection onto

$$\mathrm{span}\{h_1, \ldots, h_{k+p}\}$$

Randomized SVD for Hilbert-Schmidt operators

1. Sample $f_1, \ldots, f_{k+p} \sim F$ (Gaussian process).
2. Apply operator: $h_1 = \mathcal{A}(f_1), \ldots, h_{k+p} = \mathcal{A}(f_{k+p})$.
3. Return $\Pi_H \mathcal{A}$

Implementation of Step 3 depends on $\mathcal{A}$. For an integral op:

$$(\Pi_H A f)(x) = \int_{D_y} \underbrace{H(x)(H^* H)^{-1} H^* g(\cdot, y)}_{=: g_{k+p}(x,y)} f(y)\, \mathrm{d}y,$$

where

- $H(x) = [h_1(x), \ldots, h_{k+p}(x)]$
- $H^* H = \begin{bmatrix} \langle h_1, h_1 \rangle & \cdots & \langle h_1, h_{k+p} \rangle \\ \vdots & & \vdots \\ \langle h_{k+p}, h_1 \rangle & \cdots & \langle h_{k+p}, h_{k+p} \rangle \end{bmatrix}$,
  $H^* g(\cdot, y) = \begin{bmatrix} \langle h_1, g(\cdot, y) \rangle \\ \vdots \\ \langle h_{k+p}, g(\cdot, y) \rangle \end{bmatrix}$
- g_{k+p} is a reduced kernel of rank $k + p$

Analysis of randomized SVD for HS

Structural bound carries through without difficulties [BT]:

$$\|\mathcal{A} - \Pi_H \mathcal{A}\|_{\mathrm{HS}}^2 \le \|\Sigma_2\|_{\mathrm{HS}}^2 + \|\Sigma_2 \Omega_2 \Omega_1^\dagger\|_{\mathrm{HS}}^2$$

where:

- $\mathcal{A}$ is HS with SVD

$$\mathcal{A} = U_1 \Sigma V_1^* + \sum_{i=k+1}^{\infty} u_i \langle v_i, \cdot \rangle$$

- $\Sigma_2 = \mathrm{diag}(\sigma_1, \sigma_2, \ldots)$
- "$\Omega_2 = V_2^* F$"
- $\Omega_1 = V_1^* F = \begin{bmatrix} \langle v_1, f_1 \rangle & \cdots & \langle v_1, f_{k+p} \rangle \\ \vdots & & \vdots \\ \langle v_k, f_1 \rangle & \cdots & \langle v_k, f_{k+p} \rangle \end{bmatrix}$

Two key differences to analysis in fd case:

- Ω_1, Ω_2 are not independent
- Ω_1 is not a Gaussian matrix

Analysis of randomized SVD for HS

On the distribution of Ω_1:

- In finite dimensions:
  If $f \sim \mathcal{N}(0, K)$ then $V_1^* f \sim \mathcal{N}(0, V_1^* K V_1)$.
- In infinite dimensions, continuity argument via (truncated) KL expansion: Each column of $\Omega_1 = V_1^* F$ is independent and $\sim \mathcal{N}(0, K)$, with

$$k_{ij} = \int_{D_y} \int_{D_x} v_i(x) \kappa(x, y) v_j(y) \, \mathrm{d}x \mathrm{d}y$$

Difficulty: Eigenvalues of $\kappa(x, y)$ decay.

Analysis of randomized SVD for HS

On the distribution of Ω_1:

- In finite dimensions:
  If $f \sim \mathcal{N}(0, K)$ then $V_1^* f \sim \mathcal{N}(0, V_1^* K V_1)$.
- In infinite dimensions, continuity argument via (truncated) KL expansion: Each column of $\Omega_1 = V_1^* F$ is independent and $\sim \mathcal{N}(0, K)$, with

$$k_{ij} = \int_{D_y} \int_{D_x} v_i(x) \kappa(x, y) v_j(y) \,\mathrm{d}x\mathrm{d}y$$

Difficulty: Eigenvalues of $\kappa(x, y)$ decay.

On the bright side: $\Omega_1 \Omega_1^T$ has Wishart distribution (with covariance matrix K) covered by textbooks [Muirhead'09]:

$$\mathbb{E}\left[\|\Omega_1^\dagger\|_F^2\right] = \frac{\text{trace}(K^{-1})}{p-1}.$$

Analysis of randomized SVD for HS

$$\mathbb{E}\left[\|\mathcal{A} - \Pi_H \mathcal{A}\|_{\mathrm{HS}}\right] \quad \leq \quad \left(1 + \sqrt{\frac{\mathrm{trace}(K^{-1})\lambda_1(k+p)}{p-1}}\right)$$

$\times$ best rank-k approximation error

Interpretation of trace(K^{-1}):

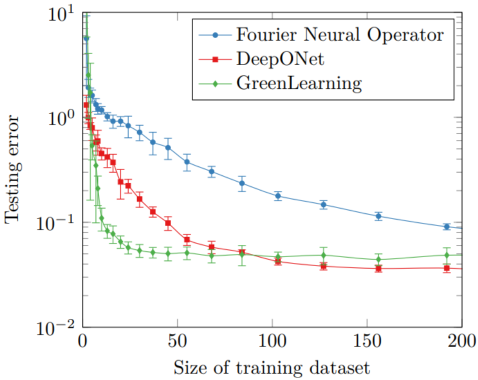
To avoid dominating best rank-k approximation error, KL eigenvalues (of GP) need to decay more slowly than (squared) singular values of $\mathcal{A}$.

Intiution: Kernel κ of GP less regular than kernel g of $\mathcal{A}$.

Randomized SVD for learning PDEs

Goal: Learn solution operator / Green's kernel for linear PDE from input ($=$source term) / output ($=$ solution) pairs.

GreenLearning[13] $=$ peeling $+$ infinite-dimensional randomized SVD.



[13] N. Boullé, D. Halikias, and A. Townsend. *Elliptic PDE learning is provably data-efficient*. 2023. arXiv: 2302.12888.

Conclusions

- Finite-dimensional randomized SVD preferred method for low-rank approximation if matrix-vector products is access model. Basic algorithm well understood.
- Infinite-dimensional setting still in its infancy.

Selected ongoing developments not discussed:

- Randomized SVD for trace estimation $\rightsquigarrow$ Hutch++ [Meyer et al.'2021].
- Randomized SVD for matrix function approximation [DK/Persson'2023].
- Potential of OSE for numerical linear algebra continues being explored: Solving least squares problems = BLENDENPIK, sketching Krylov subspaces for accelerating classical algorithms (CG, GMRES, ...), computing nullspaces, . . .