Randomized methods for low-rank approximation of matrices and tensors

Yuji Nakatsukasa
Oxford University

Based on joint work with
Behnam Hashemi (Leicester), and Maike Meier (Oxford)

Computational Mathematics for Data Science, TUD, 2023

Algorithms in Numerical Linear Algebra (NLA)

For $Ax = b, Ax = \lambda(B)x, A = U\Sigma V^T$

1. **Classical** (dense) algorithms (LU, QR, Golub-Kahan)
   - $(+)$ Incredibly reliable, backward stable
   - $(-)$ Cubic complexity $O(n^3)$
2. **Iterative** (e.g. Krylov) algorithms
   - $(+)$ Fast convergence for 'good' matrices: clustered eigenvalues or (GMRES) or well-conditioned (LSQR)
   - $(-)$ If not, need preconditioner
3. **Randomized** algorithms
   - $(+)$ Next slide(s)
   - $(-)$ Lack of reproducibility, might lose nice properties, e.g. structure

What can randomization do for you?

1. Sketch and **solve/precondition**
   - least-squares [Rokhlin-Tygert (08)], [Drineas-Mahoney-Muthukrishnan-Sarlós (10)], [Avron-Maymounkov-Toledo (10)], [Meng-Saunders-Mahoney 14]
2. **Near**-optimal solution with lightning speed
   - e.g. SVD [Halko-Martinsson-Tropp (11)], [Woodruff (14)]
3. **Sample** to approximate
   - Monte Carlo style; often comes with error estimates
   - e.g. matrix multiplication [Drineas-Kannan-Mahoney (06)], trace estimation [Avron-Toledo (11)], [Musco-Musco-Woodruff (20)]
4. Avoid pathological situations by perturbation/blocking
   - e.g. eigenvalues [Banks-Vargas-Kulkarni-Srivastava (19)], block Lanczos [Musco-Musco 15], [Tropp 18]

What can randomization do for you?

1. Sketch and **solve/precondition**
   - least-squares [Rokhlin-Tygert (08)], [Drineas-Mahoney-Muthukrishnan-Sarlós (10)], [Avron-Maymounkov-Toledo (10)], [Meng-Saunders-Mahoney 14]
2. **Near**-optimal solution with lightning speed **Part I: low-rank SVD, Part III: low-rank tensor (Tucker)**
   - e.g. SVD [Halko-Martinsson-Tropp (11)], [Woodruff (14)]
3. **Sample** to approximate **(Part II: rank estimation)**
   - Monte Carlo style; often comes with error estimates
   - e.g. matrix multiplication [Drineas-Kannan-Mahoney (06)], trace estimation [Avron-Toledo (11)], [Musco-Musco-Woodruff (20)]
4. Avoid pathological situations by perturbation/blocking
   - e.g. eigenvalues [Banks-Vargas-Kulkarni-Srivastava (19)], block Lanczos [Musco-Musco 15], [Tropp 18]

Sketching: Key idea in randomized linear algebra

Roughly: to solve a problem w.r.t. $\boxed{A}$, form random matrix Y

and work with $\boxed{Y^T A}$ (or sometimes $\boxed{Y^T A X}$)

Key insight: the sketch inherits A's low-dimensional structure if present

Success stories in

- **Low-rank approximation** [Halko-Martinsson-Tropp 11, Woodruff 14, N. 20 etc]
- **Least-squares** [Rokhlin-Tygert 09, Avron-Maymounkov-Toledo 10]
- **Linear sytems and eigenvalue problems** [Balabanov-Grigori 22, N.-Tropp 21]
- Rank estimation [Meier-N. 21]
- and many others

Sketching for least-squares problems

For A: $n \times k$, $n \gg k$

$$\min_x \left\| A\, x - b \right\|_2 \quad \Rightarrow \quad \min_{\hat{x}} \left\| SA\, \hat{x} - Sb \right\|_2$$

With "reasonable/random" **sketch** $S \in \mathbb{C}^{s \times n}$ ($s > k$, say $s = 2k$),

$$(1-\epsilon)\|Av-b\|_2 \leq \|S(Av-b)\|_2 \leq (1+\epsilon)\|Av-b\|_2,$$

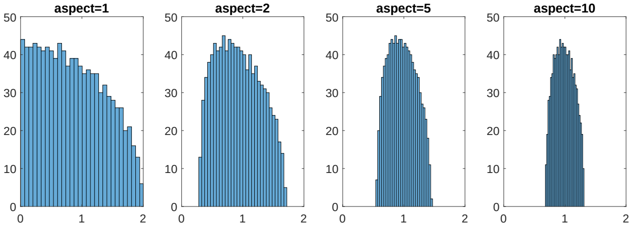
for some ϵ (not small, e.g. $\epsilon = \frac{1}{2}$) "subspace embedding". Hence the sketched solution $\hat{x}$ satisfies

$$\|A\hat{x}-b\|_2 \leq \frac{1+\epsilon}{1-\epsilon}\|Ax-b\|_2.$$

- if $\|Ax-b\|_2$ is small, $\hat{x}$ is a great solution!
- SA in $O(nk\log n)$ cost: SRFT, or $O(\mathrm{nnz}(A))$ with sparse sketch [Sarlos 06, Clarkson-Woodruff 17]
- For full accuracy do $SA = QR$, solve $\min \|AR^{-1}y - b\|_2$ via LSQR [Rokhlin-Tygert (08)], Blendenpik [Avron-Maymounkov-Toledo 10]

Explaining why sketching works via M-P

Marchenko-Pastur: 'Rectangular random matrices are well-conditioned'



$\sigma_i(G)$ for $G_{ij} \sim N(0,1)$ supported in $[\sqrt{m}-\sqrt{n}, \sqrt{m}+\sqrt{n}]$

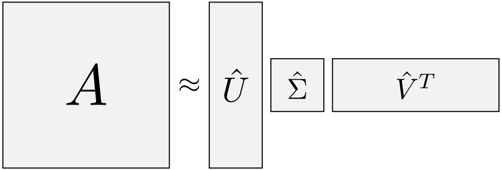
Claim: $\|Av-b\|_2 \approx \|S(Av-b)\|_2$ for all v ($\approx$: 'same up to $O(1)$ factor')

- Let $[A,b]=QR$. $S[A,b]=(SQ)R$. Can write $\|Av-b\|_2 = \|Qw\|_2$ and $\|S(Av-b)\|_2 = \|(SQ)w\|_2$.
- Now SQ is rectangular+random$\Rightarrow \sigma_i(SQ) \approx 1$ by M-P.
- Hence $\|(SQ)w\|_2 \approx \|Qw\|_2$ for all w.

Related to J-L Lemma, RIP, oblivious subspace embedding etc

(Most) important result in Numerical Linear Algebra

Given $A \in \mathbb{R}^{m\times n}$ $(m \geq n)$, find low-rank (rank r) approximation



, $\quad \hat{\Sigma} \in \mathbb{R}^{r\times r}$

- Optimal solution $A_r = U_r \Sigma_r V_r^T$ via truncated SVD
  $U_r = U(:, 1:r), \Sigma_r = \Sigma(1:r, 1:r), V_r = V(:, 1:r)$, giving

$$\|A - A_r\| = \|\mathrm{diag}(\sigma_{r+1}, \ldots, \sigma_n)\|$$

  in any unitarily invariant norm [von Neumann 37, Horn-Johnson 85]
- But that costs $O(mn^2)$; look for faster approximation
- Low-rank matrices everywhere [Beckermann-Townsend 17]

Part I: Randomized low-rank matrix approximation

[Halko-Martinsson-Tropp, SIREV 2011]

1. Form a random matrix $X \in \mathbb{R}^{n \times r}$.
2. Compute AX and its QR factorization $AX = QR$.
3. $A \approx \boxed{Q}\ \boxed{Q^T A}$ is low-rank approx.

- $O(mnr)$ cost for dense A, can be reduced to $O(mn \log n + mr^2)$ via FFT and interp. decomp. (slightly worse accuracy)
- mr^2 dominant if $r > \sqrt{n}$ or e.g. A sparse
- Near-optimal approximation guarantee: for any $\hat{r} < r$,

$$\mathbb{E}\|A - \hat{A}\|_F \leq \left(1 + \frac{r}{r - \hat{r} - 1}\right) \|A - A_{\hat{r}}\|_F$$

where $A_{\hat{r}}$ is the (optimal) rank $\hat{r}$-truncated SVD

Generalized Nyström

Generalized Nyström (GN) : [N. 2020]

$$A \approx AX(Y^TAX)^\dagger Y^TA = \boxed{AX}\,\boxed{(Y^TAX)^\dagger}\,\boxed{Y^TA}$$

- $X \in \mathbb{R}^{n\times r}, Y \in \mathbb{R}^{m\times(r+\ell)}$, $\ell = cr$ (we choose $c = 0.5$)
  - e.g. **Gaussian** $X_{ij} \sim N(0,1)$
  - or **SRFT** $X = DFS$, D : diag, F: FFT, S: subsampling (or hashing)
- Near-optimal cost, essentially AX and Y^TA. Single-pass
- Near-optimal accuracy, comparable to HMT, Nyström

Generalized Nyström

stabilized Generalized Nyström (SGN) : [N. 2020]

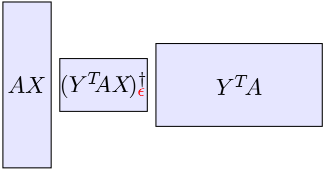
$$A \approx AX(Y^TAX)^{\dagger}_{\epsilon}Y^TA = \boxed{AX}\;\boxed{(Y^TAX)^{\dagger}_{\epsilon}}\;\boxed{Y^TA}$$

- $X \in \mathbb{R}^{n\times r}, Y \in \mathbb{R}^{m\times(r+\ell)}$, $\ell = cr$ (we choose $c = 0.5$)
  - e.g. **Gaussian** $X_{ij} \sim N(0,1)$
  - or **SRFT** $X = DFS$, D : diag, F: FFT, S: subsampling (or hashing)
- Near-optimal cost, essentially AX and Y^TA. Single-pass
- Near-optimal accuracy, comparable to HMT, Nyström
- Numerically stable with ϵ-pseudoinverse $(U\Sigma V^T)^{\dagger}_{\epsilon} = V\Sigma^{\dagger}_{\epsilon}U^T$

Generalized Nyström

stabilized Generalized Nyström (SGN) : [N. 2020]

$$A \approx AX(Y^TAX)^\dagger_\epsilon Y^TA =$$



- $X \in \mathbb{R}^{n\times r}, Y \in \mathbb{R}^{m\times(r+\ell)}$, $\ell = cr$ (we choose $c = 0.5$)
  - e.g. **Gaussian** $X_{ij} \sim N(0,1)$
  - or **SRFT** $X = DFS$, D : diag, F: FFT, S: subsampling (or hashing)
- Near-optimal cost, essentially AX and Y^TA. Single-pass
- Near-optimal accuracy, comparable to HMT, Nyström
- Numerically stable with ϵ-pseudoinverse $(U\Sigma V^T)^\dagger_\epsilon = V\Sigma^\dagger_\epsilon U^T$
- Key tool for convergence+stability analysis: **Marchenko-Pastur**

Quick proof of why Range(AX) is good

If $A = U_1 \Sigma_1 V_1^T + E$ ($\|E\|$ small), then

$$AX = U_1 \Sigma_1 {\color{blue}V_1^T X} + EX,$$

${\color{blue}V_1^T X}$ Gaussian if X is, and rectangular

So by M-P $\|(V_1^T X)^\dagger\| = O(1)$. Right-multiply $(V_1^T X)^\dagger V_1^T$ to get

$$AX (V_1^T X)^\dagger V_1^T + \tilde{E} = U_1 \Sigma_1 V_1^T + \tilde{E} \approx A$$

Hence Range$(A) \underset{\sim}{\subset}$ Range(AX)

Approximants of form $AX(Y^TAX)^\dagger Y^TA$

(or $A(A^TA)^qX(Y^TA(A^TA)^qX)^\dagger Y^TA$)

Ω: random matrix (e.g. Gaussian, SRFT)

	X, Y	q	stable?	cost for dense A
HMT 2011	$X=\Omega, Y=AX$	0	$\surd$	$O(mnr)$
Nyström ($A \succ 0$)	$Y=X=\Omega$	0	$(\times)$	$O(mn\log n + mr^2)$
HMT+Nyström	$Y=X=Q, A\Omega=QR$	1	$(\times)$	$O(mnr)$
Subspace iter	$X=\Omega, Y=\tilde{\Omega}$	>1	$(\surd)$	$O(mnrq)$
TYUC19	(4 sketch matrices)	0	$(\surd)$	$O(mn\log n + mr^2)$
TYUC17	$X=\Omega, Y=\tilde{\Omega}$	0	$(\surd)$	$O(mn\log n + mr^2)$
Clarkson-Woodruff09(C-W)	$X=\Omega, Y=\tilde{\Omega}$	0	$(\times)$	$O(mn\log n + r^3)$
Demmel-Grigori-Rusciano19	C-W+extra term	0	$(\times)$	$O(mn\log n + mr^2)$
This work, GN	$X=\Omega, Y=\tilde{\Omega}$	0	$\surd$	$O(mn\log n + r^3)$

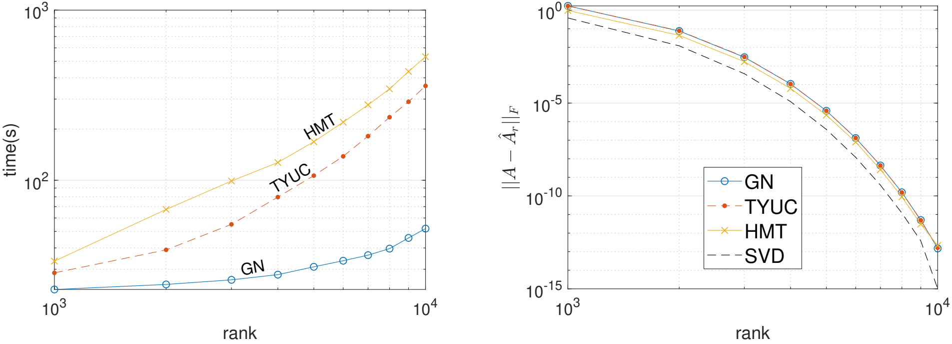
$(\times)$: unstable examples exist (though often perform ok)

$(\surd)$: conjectured to be stable (no proof)

- GN Combines **stability** and **near-optimal complexity**
- explicit constants available: GN $10mn\log n + \frac{7}{3}r^3$ flops

Experiments: dense matrix

Dense 50000×50000 matrix w/ geom. decaying σ_i

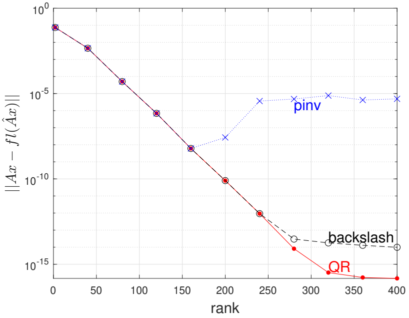


HMT: Halko-Martinsson-Tropp 11, TYUC: Tropp-Yurtsever-Udell-Cevher 17

- GN and TYUC have same accuracy (as they should)
- GN faster, up to ≈ 10x

Experiments: implementation of $(Y^TAX)^\dagger$ and stability

$$A \approx AX(Y^TAX)^\dagger Y^TA$$



- pinv (direct computation of pseudoinverse) is unsurprisingly unstable
- backslash is better but not perfect
- QR-based $\hat{A}_r = ((AX)R^{-1})(Q^T(Y^TA))$ (recommended) implementation is provenly stable

Part I in a nutshell

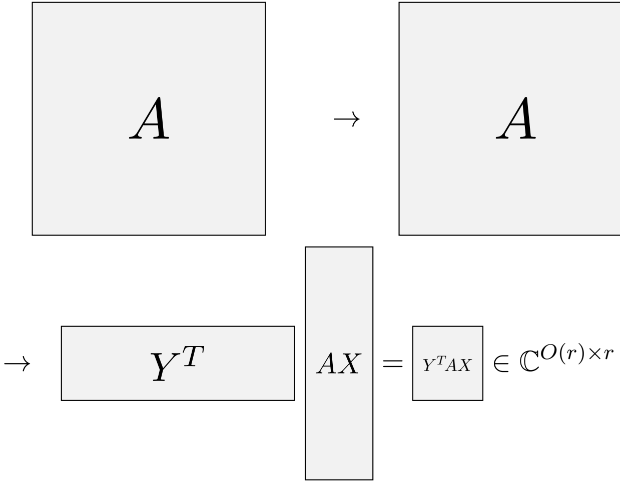
```
n = 1000; % size
A = gallery('randsvd',n,1e100);
r = 200;  % rank

X = randn(n,r); Y = randn(n,1.5*r);
AX = A*X;
YA = Y'*A;
YAX = YA*X;
[Q,R] = qr(YAX,0);    % stable implementation of pseudoinverse
At = (AX/R)*(Q'*YA);

norm(At-A,'fro')/norm(A,'fro')
ans = 2.8138e-15
```

For details, please see arXiv 2009.11392
"Fast and stable randomized low-rank matrix approximation"

Rank estimation main idea: random embedding preserves $O(\sigma_i)$



X, Y: Gaussian (or SRFT), scaled s.t. $\sigma_i(Q^T X), \sigma_i(YQ) \in [1-\delta, 1+\delta]$.

Key fact: $\frac{\sigma_i(A)}{\sigma_i(Y^T AX)} = O(1)$ for $i = 1, 2, \ldots, r$

The rank estimation algorithm

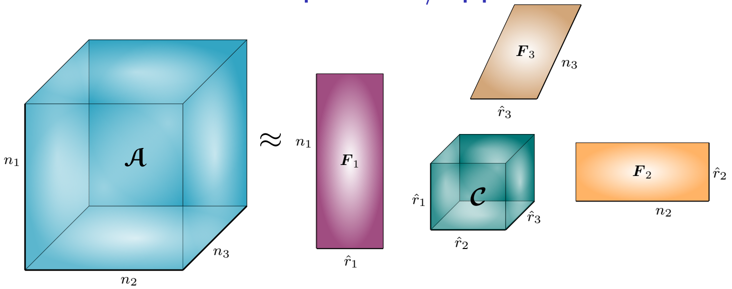
Algorithm Given $A \in \mathbb{C}^{m\times n}$, tolerance ϵ and an upper bound for rank r_1, compute approximate ϵ-rank.

1: Set $\tilde{r}_1 = \text{round}(1.1r_1)$ to oversample by 10%.
2: Draw $n \times \tilde{r}_1$ random embedding matrix X.
3: Sketch: Compute the $m \times \tilde{r}_1$ matrix AX.
4: Set $r_2 = 1.5\tilde{r}_1$, draw an $r_2 \times m$ SRFT embedding matrix Y.
5: Form the $r_2 \times \tilde{r}_1$ matrix Y^TAX.
6: Compute the first r_1 singular values of Y^TAX.
7: Output smallest $\hat{r}$ s.t. $\sigma_{\hat{r}+1}(Y^TAX) \le \epsilon$.

- Complexity: $O(mn\log n + r^3)$
- When done within GN $AX(Y^TAX)^{\dagger}Y^TA$, extra cost is marginal

Please see [Meier-N. arXiv 2020] for details

Part III: Tucker decomposition/approximation of tensors



$\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots n_d}$

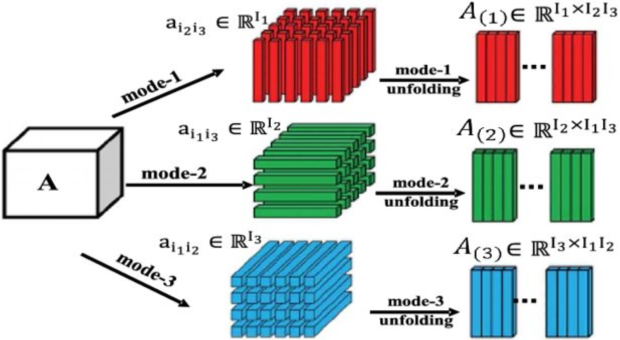
Tucker decomposition:

$$\mathcal{A} := \mathcal{C} \times_1 F_1 \times_2 F_2 \cdots \times_d F_d$$

- Factor matrix $F_i \in \mathbb{R}^{n_i \times \hat{r}_i}$, $(\hat{r}_1, \ldots, \hat{r}_d) \leq (n_1, \ldots, n_d)$, often "$\ll$"
- Easy to force F_i orthonormal (not necessary)

Other tensor decompositions (not covered here): CP, tensor train

Unfoldings



[Image from Ouamane et al (2017)]

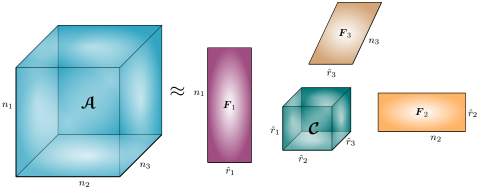
If $\mathcal{C} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, $M \in \mathbb{R}^{m_k \times n_k}$, then

$$\mathcal{B} = \mathcal{C} \times_k M \in \mathbb{R}^{n_1 \times \cdots n_{k-1} \times m_k \times n_{k+1} \times \cdots \times n_d}$$

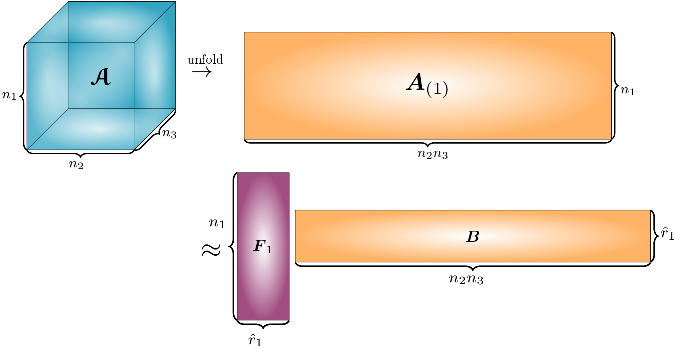
is the mode-k product of $\mathcal{C}$ and M if $B_{(k)} = MC_{(k)}$.

Big-picture idea

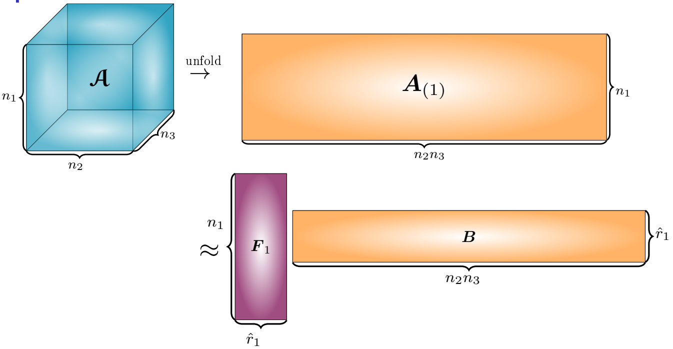
Idea: if



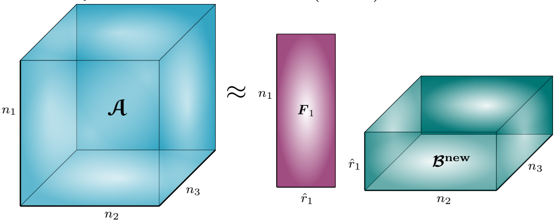
then



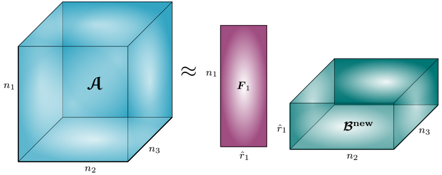
Big-picture idea cont'd



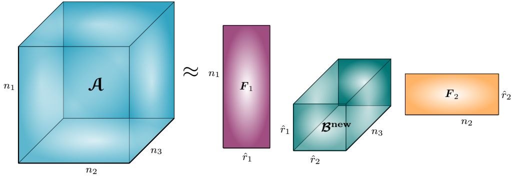
This implies with $B = \text{unfold}(\mathcal{B}^{\text{new}})$



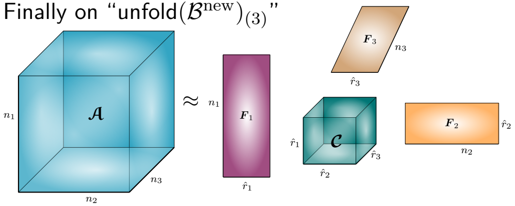
RTSMS:overview



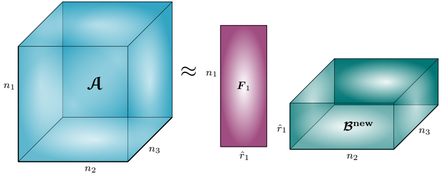
Repeat: work on "unfold$(\mathcal{B}^{\text{new}})_{(2)}$"



Finally on "unfold$(\mathcal{B}^{\text{new}})_{(3)}$"

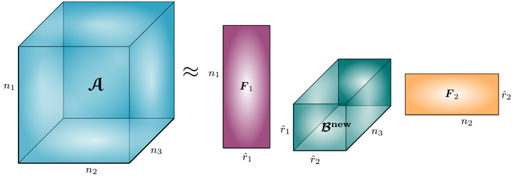


RTSMS:overview

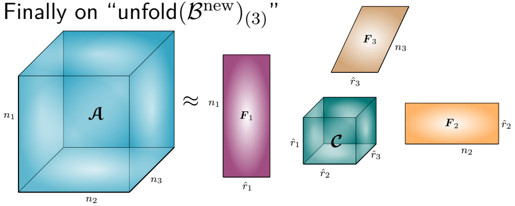


$\leftarrow$ Question: how to do this?

Repeat: work on "unfold$(\mathcal{B}^{\text{new}})_{(2)}$"

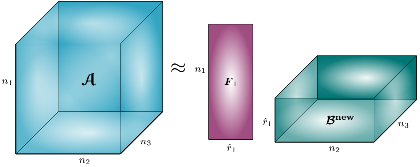


Finally on "unfold$(\mathcal{B}^{\text{new}})_{(3)}$"

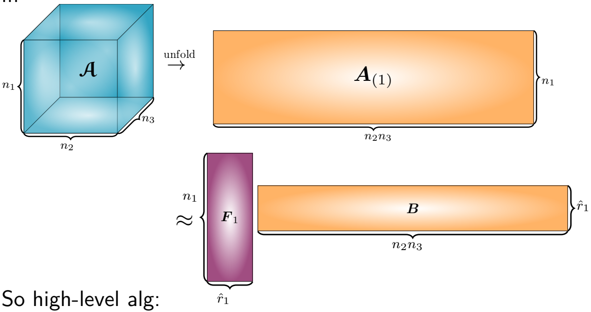


RTSMS:overview

Repost:



iff

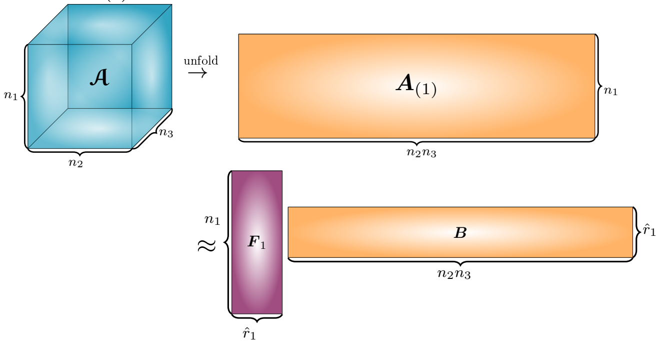


So high-level alg:

1. Unfold current core tensor to get (fat) matrix $A_{(1)}$
2. Find low-rank approximation $A_{(1)} \approx F_1 B^{(2)}$

Low-rank approximation of unfolding

To find $A_{(1)} \approx F_1 B^{(2)}$



One can use (alg may find F first or B first)

- SVD: STHOSVD [Vannieuwenhoven-Vandebril-Meerbergen 12]
- HMT: R-STHOSVD [Minster-Saibaba-Kilmer 20]
- GN: (roughly) RTSMS (this work)
- Other approaches: HOSVD on unfoldings of original tensor $\mathcal{A}$ (more computation, perhaps more parallel) [Sun-Guo-Luo-Tropp-Udell (20) etc]

RTSMS (Randomized Tucker via Single-Mode-Sketch)

From GN: Taking Gaussian $\boxed{\Omega} \in \mathbb{R}^{r_1 \times n_1}$,

$$\boxed{A_{(1)}} \approx \boxed{\hat{F}} \, \boxed{\Omega A_{(1)}}$$

Then find $\hat{F}$. In GN, Ω_2 iid Gaussian, $A_{(1)} \approx A_{(1)}\Omega_2(\Omega A_{(1)}\Omega_2)^{\dagger}\Omega A_{(1)}$

Theorem

Let $\hat{\mathcal{A}}$ be the output of RTSMS with Gaussian sketches. Then

$$\mathbb{E}\|\hat{\mathcal{A}} - \mathcal{A}\|_F \leq \sum_{j=1}^{d} \left(\prod_{i=1}^{j} \sqrt{1 + \frac{\hat{r}_i}{\ell_i - 1}} \sqrt{1 + \frac{\hat{r}_i - \ell_i}{\hat{r}_i - \ell_i - r_i - 1}} \right) \|\mathcal{A} - \mathcal{A}_{\mathrm{opt}}\|_F,$$

where $\mathcal{A}_{\mathrm{opt}}$ is the best Tucker approx., $1 < \ell_i \leq \hat{r}_i - r_i$.

RTSMS (Randomized Tucker via Single-Mode-Sketch)

From GN: Taking Gaussian $\boxed{\Omega} \in \mathbb{R}^{r_1 \times n_1}$,

$$\boxed{A_{(1)}} \approx \boxed{\hat{F}}\,\boxed{\Omega A_{(1)}}$$

Then find $\hat{F}$. In GN, Ω_2 iid Gaussian, $A_{(1)} \approx A_{(1)}\Omega_2(\Omega A_{(1)}\Omega_2)^{\dagger}\Omega A_{(1)}$
but then $\Omega_2 \in \mathbb{R}^{(n_2 n_3 \cdots n_d) \times O(\hat{r}_1)}$, **enormous** (storage cost)
Instead: in RTSMS we obtain $\hat{F}$ via the least-squares problem

$$\min_{F \in \mathbb{R}^{n_1 \times \hat{r}_1}} \left\| \boxed{A_{(1)}^T \Omega^T}\,\boxed{\hat{F}^T} - \boxed{A_{(1)}^T} \right\|_2$$

RTSMS: solving LS

$$\min_{\hat{F}\in\mathbb{R}^{n_1\times\hat{r}_1}} \left\| A_{(1)}^T\Omega_1^T \, \hat{F}^T - A_{(1)}^T \right\|_F$$

- Massively overdetermined $(n_2\cdots n_d)\times\hat{r}_1$
- Many right-hand sides $(A_{(1)}^T \in \mathbb{R}^{(n_2\cdots n_d)\times n_1})$
- $A_{(1)}^T\Omega_1^T$ is extremely ill-conditioned (by assumption/construction)

Which means

- Sketching is natural+attractive approach
- Important to avoid sketching cost for RHS, $SA_{(1)}^T$
- Stability issues: Natural approaches (sketch-to-solve, Blendenpik, even backslash) don't work

RTSMS: solving LS

As before, sketch for efficiency:

$$\min_{\hat{F}\in\mathbb{R}^{n_1\times\hat{r}_1}} \left\| \boxed{\color{red}{S}} \left(\boxed{A_{(1)}^T\Omega_1^T} \boxed{\hat{F}^T} - \boxed{A_{(1)}^T} \right) \right\|_F$$

- To reduce sketching cost for $SA_{(1)}^T$, let $S \in \mathbb{R}^{s\times n_2 n_3}$ be subsampling matrix (row-submatrix of $I_{n_2 n_3}$), indices chosen via **leverage scores** of $A_{(1)}^T\Omega_1^T$ (i.e., row norms of orthonormal basis), also estimated via randomization
- Rows are chosen randomly with probability proportional to leverage scores
- Rank adaptivity: computation gives rank estimate almost for free

LS and sketched LS

Fact about general (sketched) least-squares problems:

Theorem

Let $A = QR$ be thin QR factorization with $Q \in \mathbb{R}^{m\times n}$, and let $\hat{X}_*$ denote the solution for $\min_X \|S(AX - B)\|_F$, $S \in \mathbb{R}^{s\times m}$, $m > s > n$. Then

$$\|A\hat{X}_* - B\|_F \leq \frac{\|S\|_2}{\sigma_{\min}(S^T Q)} \min_X \|AX - B\|_F. \tag{1}$$

- Important that $\sigma_{\min}(S^T Q)$ not small (as in DEIM), and $\|S\|_2$ not enormous
- Good subset selection (leverage scores, QRCP, GEPP, Batson-Spielman-Srivastava etc) achieves this

Solving ill-conditioned LS

To improve stability of $\min_{\hat{F}} \|S(A_{(1)}^T \Omega_1^T \hat{F}^T - A_{(1)}^T)\|_F$ (ill-conditioned)

1. Tikhonov regularization: For a fixed/small $\lambda > 0$,

$$\min_{\hat{F}^{(1)} \in \mathbb{R}^{n_1 \times \hat{r}_1}} \|S_1(A_{(1)}^T \Omega_1^T (\hat{F}^{(1)})^T - A_{(1)}^T)\|_F^2 + \lambda \|\hat{F}^{(1)}\|_F^2.$$

Equivalent to $\min_{\hat{F}} \left\| \begin{bmatrix} S_1 A_{(1)}^T \Omega_1^T \\ \sqrt{\lambda} I \end{bmatrix} \hat{F} - \begin{bmatrix} S_1 A_{(1)}^T \\ 0 \end{bmatrix} \right\|_F^2.$

Solving ill-conditioned LS

To improve stability of $\min_{\hat{F}} \|S(A_{(1)}^T \Omega_1^T \hat{F}^T - A_{(1)}^T)\|_F$ (ill-conditioned)

1. Tikhonov regularization: For a fixed/small $\lambda > 0$,

$$\min_{\hat{F}^{(1)} \in \mathbb{R}^{n_1 \times \hat{r}_1}} \|S_1(A_{(1)}^T \Omega_1^T (\hat{F}^{(1)})^T - A_{(1)}^T)\|_F^2 + \lambda \|\hat{F}^{(1)}\|_F^2.$$

   Equivalent to $\min_{\hat{F}} \left\| \begin{bmatrix} S_1 A_{(1)}^T \Omega_1^T \\ \sqrt{\lambda} I \end{bmatrix} \hat{F} - \begin{bmatrix} S_1 A_{(1)}^T \\ 0 \end{bmatrix} \right\|_F^2.$

2. Iterative refinement: Compute residual $B := A_{(1)}^T - \hat{F}^{(1)} \Omega_1 A_{(1)}$, and solve

$$\min_{\hat{F}^{(2)} \in \mathbb{R}^{n_1 \times \hat{r}_1}} \|S_2(A_{(1)}^T \Omega_1^T (\hat{F}^{(2)})^T - B)\|_F^2 + \lambda \|\hat{F}^{(2)}\|_F^2.$$

   Overall solution: $F = \hat{F}^{(1)} + \hat{F}^{(2)}$, yielding $A_{(1)} \approx F \Omega A_{(1)}$

RTSMS summary

Algorithm RTSMS: Given $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ and target tolerance *tol*, find approximate Tucker decomposition.

1: Set $\mathcal{B}^{\text{old}} := \mathcal{A}$.
2: **for** $i = 1, \ldots, d$ **do**
3: Find rank r_i via randomized rank estimator s.t. $\sigma_{r_i}(B^{\text{old}}_{(i)}) \lesssim tol$ (unless r_i given)
4: Draw Gaussian $\Omega_i \in \mathbb{R}^{\hat{r}_i \times n_i}$ where $\hat{r}_i := \text{round}(1.5\, r_i)$.
5: Compute $\mathcal{B}^{\text{new}} = \mathcal{B}^{\text{old}} \times_i \Omega_i$.
6: Find F_i of size $n_i \times \hat{r}_i$ to minimize $\|\mathcal{B}^{\text{new}} \times_i F_i - \mathcal{B}^{\text{old}}\|_F$, using leverage scores+regularization+iterative refinement
7: Update $\mathcal{B}^{\text{old}} := \mathcal{B}^{\text{new}}$.
8: **end for**
9: Set $\mathcal{C} := \mathcal{B}^{\text{new}}$.

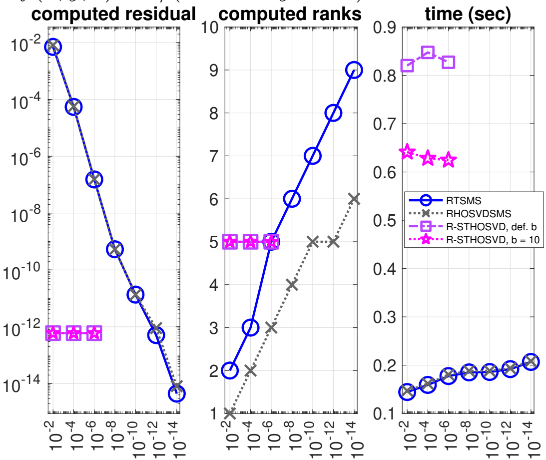
Comparison

Table: Costs for computing rank $(r, r, \ldots, r)$ Tucker of an order-d tensor $n \times n \cdots \times n$, $r \ll n$. $\hat{r} = r + p$ (p: oversampling, e.g. $p = 5$ or $p = 0.5r$).

Algorithm	dominant cost	sketch size	dominant operation
HOSVD [De Lathauwer et al 00]	dn^{d+1}		SVD of d unfoldings each of size $n \times n^{d-1}$
STHOSVD [Vannieuwenhoven et al 12]	n^{d+1}		SVD of $A_{(1)}$ which is $n \times n^{d-1}$. (Later unfoldings are smaller due to truncation)
R-HOSVD [Minster-Saibaba-Kilmer 20]	drn^d	$\hat{r} \times n^{d-1}$	computing $A_{(i)}\Omega_i$ where Ω_i of size $n^{d-1} \times \hat{r}$ and then forming $Q_i^T A_{(i)}$ for all i
R-STHOSVD [Minster-Saibaba-Kilmer 20]	rn^d	$\hat{r} \times n^{d-1}$	forming $A_{(1)}\Omega_1$ with Ω_1 of size $n^{d-1} \times \hat{r}$. Subsequent unfoldings and sketching matrices are smaller
single-pass [Sun et al.(20)]	rn^d	$\hat{r} \times n^{d-1}$	sketching by structured (Khatri-Rao product) dimension reduction maps
RTSMS	rn^d $(n^d \log n)$	$\hat{r} \times n$	computing $\Omega_1 A_{(1)}$ with Ω_1 of size $\hat{r} \times n^{d-1}$

Experiments

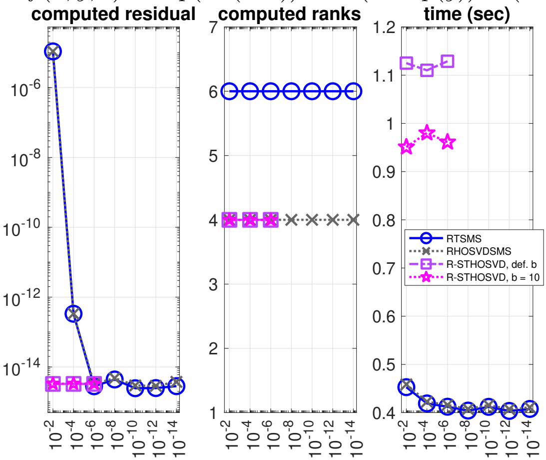
Runge function $f(x, y, z) = 1/(5 + x^2 + y^2 + z^2)$



- RHOSVDSMS: RTSMS followed by orthogonalization of F_i
- R-STHOSVD: [Minster-Saibaba-Kilmer 2020]

More experiments

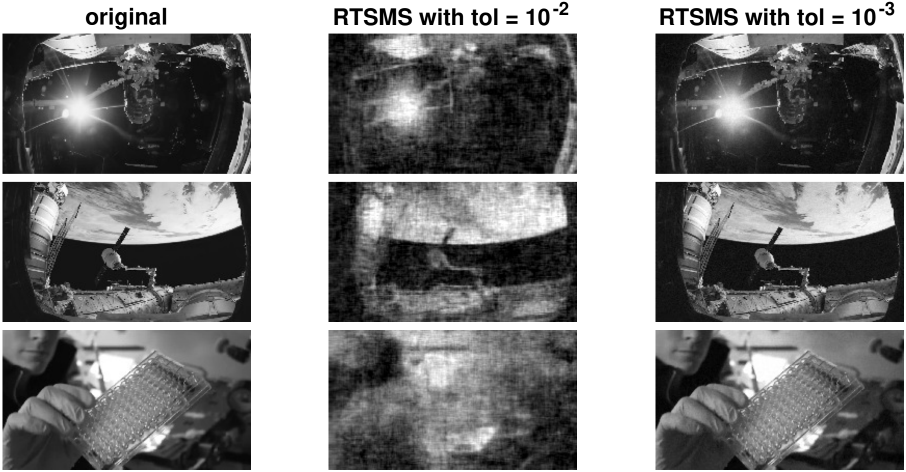
Wagon function $f(x,y,z) = \exp(\sin(50x)) + \sin(60\exp(y))\sin(60z) + \cdots$



- RHOSVDSMS: RTSMS followed by orthogonalization of F_i
- R-STHOSVD: [Minster-Saibaba-Kilmer 2020]

Compressing videos

$\mathcal{A}$: 3D tensor $483 \times 720 \times 1280$; 483 frames of a video



Summary

- Randomization for all sorts of NLA problems (we've seen low-rank approx (matrix, tensors), rank estimation, least squares, leverage scores)
- For tensors, single-mode-sketch$\rightarrow$small sketch, economical
- Challenging least-squares problem, stability improved by subsampling$+$regularization$+$iterative refinement (no proof)

[B. Hashemi and Y. Nakatsukasa, arXiv soon].

Summary

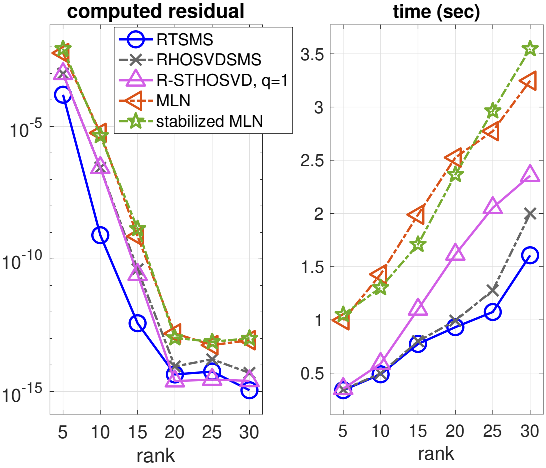
- Randomization for all sorts of NLA problems (we've seen low-rank approx (matrix, tensors), rank estimation, least squares, leverage scores)
- For tensors, single-mode-sketch$\rightarrow$small sketch, economical
- Challenging least-squares problem, stability improved by subsampling$+$regularization$+$iterative refinement (no proof)

[B. Hashemi and Y. Nakatsukasa, arXiv soon].

Post position available! (starting Mar 2024–Feb 2025)

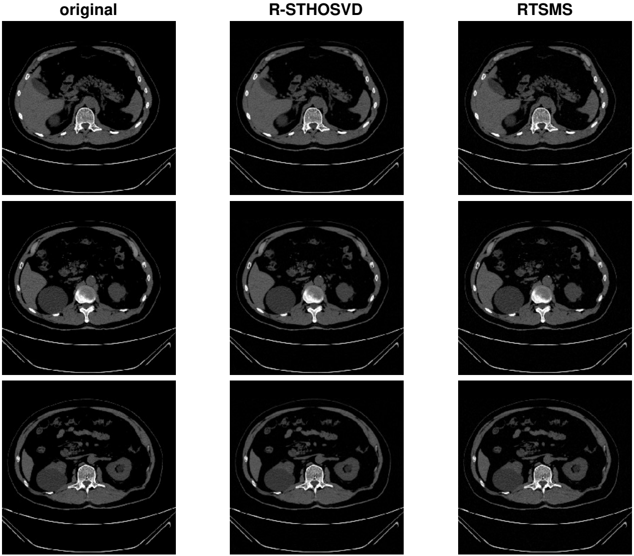
Fixed-rank experiments

Hilbert tensor $100 \times 100 \times 100 \times 100$, $A_{i,j,k,l} = \frac{1}{i+j+k+l-3}$.



MLN: [Bucci-Robol 23] (based on GN but rather different)

Tomography example



Analysis: basic facts

For any $\hat{A}$ of form $\hat{A} = (AX(Y^TAX)^\dagger Y^T)A$, (incl. HMT, GN, Nyström)

- $\hat{A} = \mathcal{P}_{AX,Y}A$, where $\mathcal{P}_{AX,Y} := AX(Y^TAX)^\dagger Y^T$ is (usually oblique) projection
- Also $A(X(Y^TAX)^\dagger Y^TA) = A\mathcal{P}_{X,A^TY}$
- Error is

$$
\begin{aligned}
E &= A - X(Y^TAX)^\dagger Y^TA = (I - \mathcal{P}_{AX,Y})A \\
&= A(I - \mathcal{P}_{X,A^TY}) = (I - \mathcal{P}_{AX,Y})A(I - \mathcal{P}_{X,A^TY}).
\end{aligned}
$$

Also

$$E = (I - \mathcal{P}_{AX,Y})A = (I - \mathcal{P}_{AX,Y})A(I - XM^T)$$

for any M, because $(I - \mathcal{P}_{AX,Y})AX = 0$.

Analysis for HMT

$$\hat{A} = (AX(Y^TAX)^\dagger Y^T)A = \mathcal{P}_{AX,Y}A,$$

where $Y = AX$, so $\mathcal{P}_{AX,Y} =: \mathcal{P}_{AX}$ is orthogonal projector, $\|\mathcal{P}_{AX}\|_2 = \|I - \mathcal{P}_{AX}\|_2 = 1$

- Error is $E_{\text{HMT}} = (I - \mathcal{P}_{AX})A(I - XM^T)$, so
$$\|E_{\text{HMT}}\| = \|(I - \mathcal{P}_{AX})A(I - XM^T)\| \le \|A(I - XM^T)\|.$$
- Take M s.t. $XM^T = X(V^TX)^\dagger V^T = \mathcal{P}_{X,V}$ is oblique projection w/ row space V^T (top $\hat{r}$ sing. vecs. of A), $V^T(I - \mathcal{P}_{X,V}) = 0$, so $A(I - \mathcal{P}_{X,V}) = A(I - VV^T)(I - \mathcal{P}_{X,V})$.
- Thus with $\Sigma_2 = \text{diag}(\sigma_{\hat{r}+1}, \dots, \sigma_n)$,

$$\begin{aligned}\|E_{\text{HMT}}\| &\le \|A(I - VV^T)(I - \mathcal{P}_{X,V})\| = \|\Sigma_2 V_\perp V_\perp^T(I - \mathcal{P}_{X,V})\| \\ &\le \|\Sigma_2\|\|(I - \mathcal{P}_{X,V})\|_2 = \|\Sigma_2\|\|\mathcal{P}_{X,V}\|_2 = \|\Sigma_2\|\|X(V^TX)^\dagger\|_2\end{aligned}$$

'rectangular Gaussians are well-cond.': $\|X(V^TX)^\dagger\|_2 \lesssim \frac{\sqrt{m}+\sqrt{r}}{\sqrt{r}-\sqrt{\hat{r}}} =$ "$O(1)$"

Analysis for Generalized Nyström

$$\hat{A} = (AX(Y^TAX)^\dagger Y^T)A = \mathcal{P}_{AX,Y}A,$$

$E = (I - \mathcal{P}_{AX,Y})A = (I - \mathcal{P}_{AX,Y})A(I - XM^T)$ choose M such that $XM^T = X(V^TX)^\dagger V^T = \mathcal{P}_{X,V}$, we have

$$\begin{aligned}\|E\| &= \|(I - \mathcal{P}_{AX,Y})A(I - \mathcal{P}_{X,V})\| \\ &\le \|(I - \mathcal{P}_{AX,Y})A(I - VV^T)(I - \mathcal{P}_{X,V})\| \\ &\le \|A(I - VV^T)(I - \mathcal{P}_{X,V})\| + \|\mathcal{P}_{AX,Y}A(I - VV^T)(I - \mathcal{P}_{X,V})\|.\end{aligned}$$

- Note $\|A(I - VV^T)(I - \mathcal{P}_{X,V})\|$ exact same as HMT error
- Extra term $\|\mathcal{P}_{AX,Y}\|_2 = O(1)$ as before if $c > 1$ in $Y \in \mathbb{R}^{m\times cr}$
- Overall, about $(1 + \|\mathcal{P}_{AX,Y}\|_2) \approx (1 + \frac{\sqrt{n}+\sqrt{r+\ell}}{\sqrt{r+\ell}-\sqrt{r}})$ times bigger expected error than HMT, **still near-optimal**

Precise analysis for Generalized Nyström

Theorem (Reproduces TYUC 2017 Thm.4.3)

Suppose X, Y are Gaussian. Then

$$\sqrt{\mathbb{E}\|E_{\mathrm{GN}}\|_F^2} \le \sqrt{1 + \frac{r+\ell}{\ell-1}}\sqrt{\mathbb{E}\|E_{\mathrm{HMT}}\|_F^2}$$

PROOF. Write $\mathcal{P}_{AX,Y}A = Q(Q^T + Z)A$, where $Q = \mathrm{orth}(AX)$, so that $E_{\mathrm{GN}} = (I - \mathcal{P}_{AX,Y})A = (I - QQ^T)A + QZA = E_{\mathrm{HMT}} + QZA$. We have

$$QZA = Q((Y^TQ)^\dagger Y^T - Q^T)A = Q(Y^TQ)^\dagger (Y^TQ_\perp)Q_\perp^T A$$

because $((Y^TQ)^\dagger Y^T - Q^T)Q = 0$. If Y is Gaussian then Y^TQ and $Y^TQ_\perp$ are independent Gaussian, so bound follows.

Stability analysis sketch: $fl(\hat{A}) = \hat{A}_r + \epsilon$

$\hat{A} = (AX(Y^TAX)^\dagger_\epsilon)Y^TA$. Each row of $AX(Y^TAX)^\dagger_\epsilon$ is underdetermined linear system, solve via SVD or (rank-revealing) QR.

Define $s_i^T = [AX(Y^TAX)^\dagger_\epsilon]_i$, ith row

$$s_i = ((Y^TAX)^T)^\dagger_\epsilon [AX]_i^T = (X^TA^TY)^\dagger_\epsilon [AX]_i^T =: M^\dagger_\epsilon [AX]_i^T.$$

Computed version satisfies, by [ASNA Ch. 21] ($\hat{U}$: computed Range(M))

$$\hat{s}_i = (\hat{U}^TM + \epsilon)^\dagger(\hat{U}^T[AX]_i^T + \epsilon) = (M + \epsilon_i)^\dagger_\epsilon([AX]_i^T + \epsilon)_\epsilon.$$

Thus

$$\begin{aligned}
[fl(AX(Y^TAX)^\dagger_\epsilon Y^TA)]_i &= fl([AX + \epsilon]_i(Y^TAX + \epsilon_i)^\dagger_\epsilon Y^TA) \\
&= [AX]_i(Y^T\tilde{A}X)^\dagger_\epsilon Y^TA + \epsilon\|[AX]_i(Y^T\tilde{A}X)^\dagger_\epsilon\|\|Y^TA\| \\
&= [AX]_i(Y^T\tilde{A}X)^\dagger_\epsilon Y^TA + \epsilon = [\hat{A}_r]_i + \epsilon
\end{aligned}$$

Row-wise stability follows from

$\|AX(Y^TAX)^\dagger\| = O(1), \quad \|AX(Y^T\tilde{A}X)^\dagger_\epsilon\| = O(1)$ (shown separately).

Fast computation of leverage scores

Approximating Leverage scores of $M \in \mathbb{R}^{N\times n}$, $N \gg n$:

1. Sketch and QR $SA = QR$.
2. Row norms of $AR^{-1}G$, where G is $n \times O(1)$

Complexity: $O(Nn\log N)$

Idea:

- AR^{-1} is well-conditioned (as in Blendenpik), so roughly row-norms$\propto$leverage scores
- Estimate row-norm via $AR^{-1}G$ (trace/norm estimation)

Part II: Rank estimation

In most low-rank algorithms, the rank r is required as input

- If r too low: need to resketch and recompute
- If r too high: wasted computation

A fast rank estimator is thus highly desirable

Definition

$\mathrm{rank}_\epsilon(A)$: integer i s.t. $\sigma_i(A) > \epsilon \geq \sigma_{i+1}(A)$.

This work: $O(mn \log n + r^3)$ algorithm for rank estimation

[with Maike Meier (Oxford), arXiv 2021]

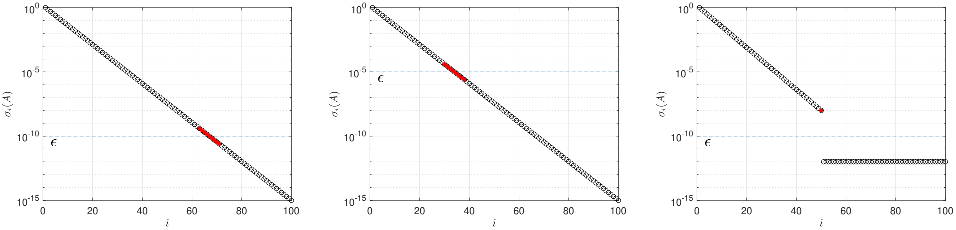
- In many cases, extra cost is much lower (e.g. $O(r^2)$)
- Key idea: **Sample** the singular values via sketching, $Y^T A X$

Goal of a rank estimator

It is usually not necessary (or even possible, with subcubic work) to find the exact ϵ-rank.

We aim to find $\hat{r}$ s.t.

- $\sigma_{\hat{r}+1}(A) = O(\epsilon)$ (say, $\sigma_{\hat{r}+1}(A) < 10\epsilon$): $\hat{r}$ is not a severe underestimate, and
- $\sigma_{\hat{r}}(A) = \Omega(\epsilon)$ (say, $\sigma_{\hat{r}}(A) > 0.1\epsilon$): $\hat{r}$ is not a severe overestimate.

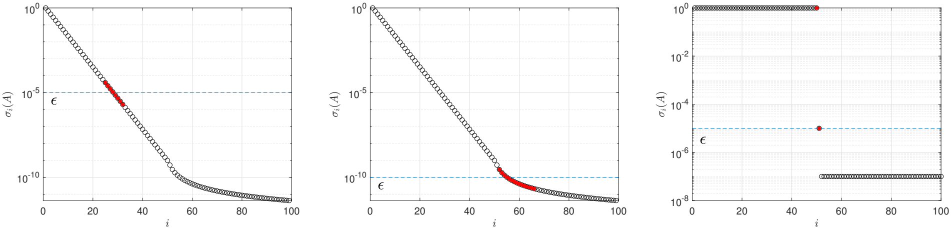


Goal of a rank estimator

It is usually not necessary (or even possible, with subcubic work) to find the exact ϵ-rank.

We aim to find $\hat{r}$ s.t.

- $\sigma_{\hat{r}+1}(A) = O(\epsilon)$ (say, $\sigma_{\hat{r}+1}(A) < 10\epsilon$): $\hat{r}$ is not a severe underestimate, and
- $\sigma_{\hat{r}}(A) = \Omega(\epsilon)$ (say, $\sigma_{\hat{r}}(A) > 0.1\epsilon$): $\hat{r}$ is not a severe overestimate.



Consequently, it suffices to estimate $\sigma_i(A)$ to their order of magnitude

Previous studies on rank estimation

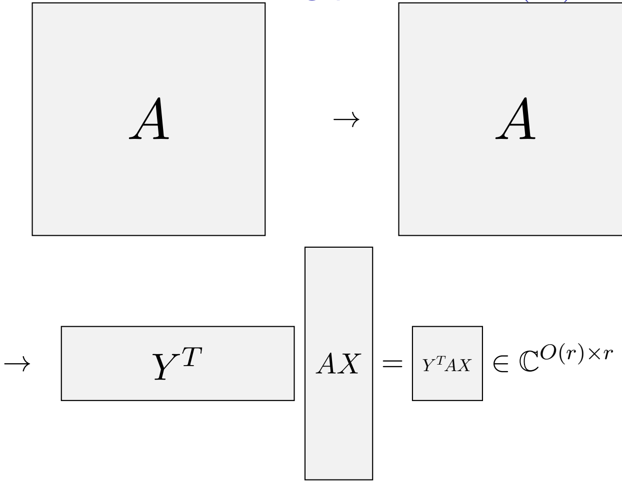
- Based on full factorization (e.g. Duersch-Gu 2020, Martinsson-Quintana-Orti-Heavner 2019)
  - cubic $O(mn^2)$complexity
- Ubaru-Saad (2016): polynomial approximation and spectral density estimates using Krylov subspace methods
  - complexity difficult to predict
- Andoni-Nguyen (2013): theory that suggest rankest possible, no algorithm

Our algorithm: based on random sketches AX, Y^TAX

Key fact: $\sigma_i(AX)/\sigma_i(A) = O(1)$ for leading i, and $\sigma_i(Y^TAX)/\sigma_i(AX) = O(1)$

- Study of $\sigma_i(AX)$ is covariance estimate
  - Usually, at least n samples required
  - But **leading** sing vals good with many fewer samples

Main idea: random embedding preserves $O(\sigma_i)$



X, Y: Gaussian (or SRFT), scaled s.t. $\sigma_i(Q^TX), \sigma_i(YQ) \in [1-\delta, 1+\delta]$.
Key fact: $\frac{\sigma_i(A)}{\sigma_i(Y^TAX)} = O(1)$ for $i = 1, 2, \ldots, r$

$\sigma_i(AX)/\sigma_i(A) = O(1)$ for leading i

Let $G \in \mathbb{C}^{n\times r}$ and

$$AG = U_1\Sigma_1(V_1^*G) + U_2\Sigma_2(V_2^*G) = U_1\Sigma_1 G_1 + U_2\Sigma_2 G_2,$$

Lemma

For $i = 1, \ldots, r$,

$$\sigma_{\min}(\hat{G}_{\{i\}}) \le \frac{\sigma_i(AG)}{\sigma_i(A)} \le \sqrt{\sigma_{\max}(\tilde{G}_{\{r-i+1\}})^2 + \left(\frac{\sigma_{r+1}(A)\sigma_{\max}(G_2)}{\sigma_i(A)}\right)^2}$$

$\hat{G}_{\{i\}} \in \mathbb{C}^{i\times r}$: first i rows of G_1, and $\tilde{G}_{\{r-i+1\}}$ last $r-i+1$ rows of G_1. If G is standard Gaussian, $\hat{G}_{\{i\}}$, $\tilde{G}_{\{r-i+1\}}$, and G_2 are independent standard Gaussian.

PROOF: Courant-Fisher minimax characterization.

$\sigma_i(AX)/\sigma_i(A) = O(1)$ cont'd

$$\sigma_{\min}(\hat{G}_{\{i\}}) \le \frac{\sigma_i(AG)}{\sigma_i(A)} \le \sqrt{\sigma_{\max}(\tilde{G}_{\{r-i+1\}})^2 + \left(\frac{\sigma_{r+1}(A)\sigma_{\max}(G_2)}{\sigma_i(A)}\right)^2}$$

When X **scaled Gaussian** (embedding)

Theorem

Let $X \in \mathbb{R}^{n\times r}$ with $X_{ij} \sim N(0, 1/r)$. Then for $i = 1, \dots, r$

$$1 - \sqrt{\frac{i}{r}} \le \mathbb{E}\frac{\sigma_i(AX)}{\sigma_i(A)} \le 1 + \sqrt{\frac{r-i+1}{r}} + \frac{\sigma_{r+1}}{\sigma_i}\left(1 + \sqrt{\frac{n-r}{r}}\right).$$

Failure probability decays squared-exponentially

Proof: Marchenko-Pastur ("rectangular random matrices are well-conditioned")

- Interpretation: $\frac{\sigma_i(AX)}{\sigma_i(A)} \approx 1$, esp. for small r

$\sigma_i(AX)/\sigma_i(A) = O(1)$ cont'd

$$\sigma_{\min}(\hat{G}_{\{i\}}) \le \frac{\sigma_i(AG)}{\sigma_i(A)} \le \sqrt{\sigma_{\max}(\tilde{G}_{\{r-i+1\}})^2 + \left(\frac{\sigma_{r+1}(A)\sigma_{\max}(G_2)}{\sigma_i(A)}\right)^2}$$

When X general embedding

Theorem

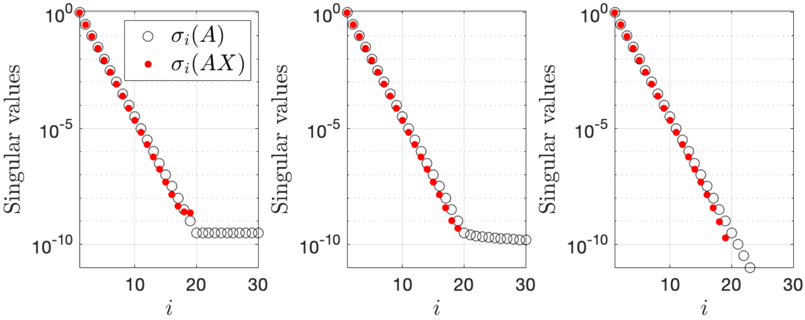
Let $\tilde{V}_1$ be A's top right singvecs, and suppose $\sigma_i(V_1^T X) \in [1-\epsilon, 1+\epsilon]$ for some $\epsilon < 1$. Then, for $i = 1, \ldots, \tilde{r}$

$$1 - \epsilon \le \frac{\sigma_i(AX)}{\sigma_i(A)} \le \sqrt{(1+\epsilon)^2 + \left(\frac{\sigma_{\tilde{r}+1}(A)\|X\|_2}{\sigma_i(A)}\right)^2}.$$

ϵ-subspace embedding, (e.g. SRFT (subsampled random Fourier transform), i.e. $X = DFS$, D : diag, F: FFT, S: subsampling), also effective choices for X

Experiments $\sigma_i(AX)/\sigma_i(A) = O(1)$

$A \in \mathbb{R}^{1000\times 1000}$



- Leading singvals estimated reliably (when they decay)
- Tail effect nonnegligible (esp. for last $i \approx r$)
- Hence trust only leading (say 90%) samples

2nd step: $\sigma_i(Y^T AX)/\sigma_i(AX) = O(1)$

Corollary (Combines Boutsidis-Gittens (13) and Tropp (11))

Let $AX \in \mathbb{R}^{m\times r_1}$, *with* $m \geq r_1$, *and let* $Y \in \mathbb{R}^{n\times r_2}$ *be an SRFT matrix. Let* $0 < \epsilon < 1/3$ *and* $0 < \delta < 1$. *If*

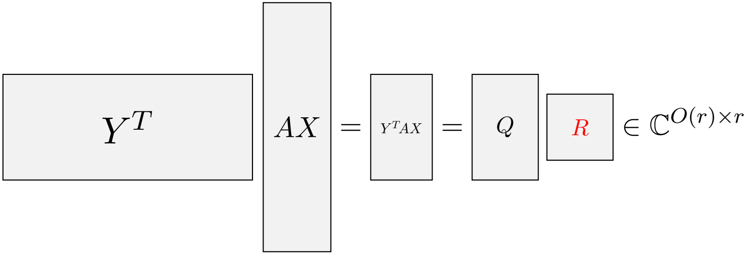
$$r_2 \geq 6\eta\epsilon^{-2}\left[\sqrt{r_1} + \sqrt{8\log(m/\delta)}\right]^2 \log(r_1/\delta),$$

then with failure probability at most 3δ

$$\sqrt{1-\epsilon} \leq \frac{\sigma_i(Y^T AX)}{\sigma_i(AX)} \leq \sqrt{1+\epsilon},$$

for each $i = 1, \ldots, r_1$.

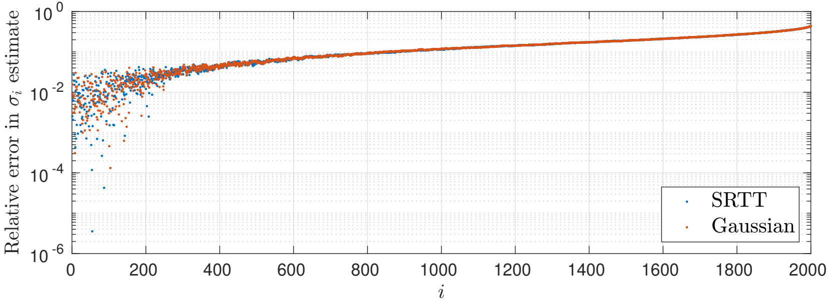
$\sigma_i(Y^TAX)/\sigma_i(AX) = O(1)$



- Approximate orthogonalization: ideas from Blendenpik etc [Avron-Maymounkov-Toledo 10]
- In generalized Nyström, $Y^TAX = QR$ already computed + rank-revealing QR $\Rightarrow \sigma_i(Y^TAX) \approx \text{diag}(R)$; only $O(r)$ extra cost

Experiments: $\sigma_i(Y^TAX)/\sigma_i(AX) = O(1)$

$AX \in \mathbb{R}^{10^5 \times 2000}$



- $|\frac{\sigma_i(Y^TAX)}{\sigma_i(AX)} - 1|$ small esp. for leading singvals
- Reasonable estimates even for $i \approx r$

The rank estimation algorithm

Algorithm Given $A \in \mathbb{C}^{m\times n}$, tolerance ϵ and an upper bound for rank r_1, compute approximate ϵ-rank.

1: Set $\tilde{r}_1 = \text{round}(1.1 r_1)$ to oversample by 10%.
2: Draw $n \times \tilde{r}_1$ random embedding matrix X.
3: Form the $m \times \tilde{r}_1$ matrix AX.

2. Approximate orthogonalization:

4: Set $r_2 = 1.5\tilde{r}_1$, draw an $r_2 \times m$ SRFT embedding matrix Y.
5: Form the $r_2 \times \tilde{r}_1$ matrix $Y^T AX$.

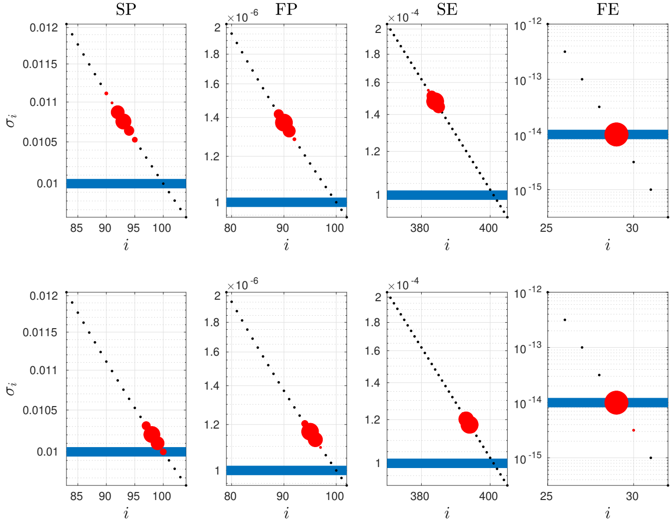
3. Singular value estimates:

6: Compute the first r_1 singular values of $Y^T AX$.
7: Output smallest $\hat{r}$ s.t. $\sigma_{\hat{r}+1}(Y^T AX) \le \epsilon$.

Complexity: $O(mn\log n + r^3)$

Experiments: rank estimation

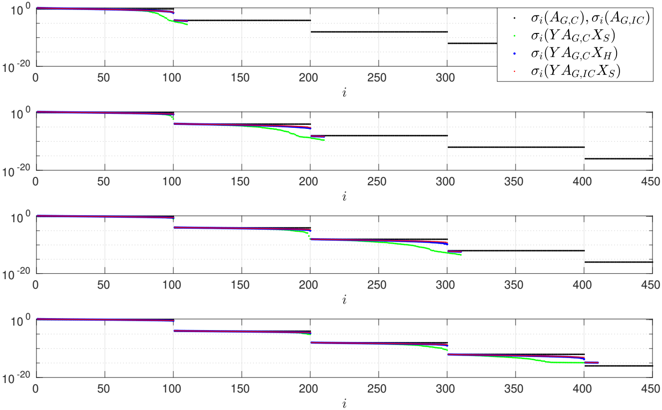
SP/FP: slow/fast polynomial decay in $\sigma_i(A)$, SE/FE: slow/fast exponential decay



Out of 100 runs; dot area reflects frequency

Experiments: gaps in singular values

$A_{G,IC}$: incoherent singvecs, $A_{G,C}$: coherent singvecs $(V = I)$



For coherent problems, *Hashed* (not subsampled) RFT helpful [Cartis-Fiala-Shao 21]

For details, please see preprint Meier-N. "Fast randomized numerical rank estimation" arXiv 2021.