Randomized TSVD Algorithms

Joel A. Tropp

Computing + Mathematical Sciences
California Institute of Technology
jtropp@cms.caltech.edu

Coauthors:
Nathan Halko; Gunnar Martinsson (UT-Austin);
Rob Webber (Caltech); Yuji Nakatsukasa (Oxford);
Alp Yurtsever (Umeå); Madeleine Udell (Cornell); Volkan Cevher (EPFL);
Katie Bouman, Aviad Levis (Caltech); Charles Gammie, David Lee (UIUC)

Research supported by ONR, AFOSR, NSF, DARPA, Sloan, and Moore.
The Famous Truncated SVD
The Singular Value Decomposition

\[ A_{m \times n} = U_{m \times m} \Sigma_{m \times n} V^*_{n \times n} \]

\[ \Lambda v_i = \sigma_i u_i \]

\[ \Lambda : \mathbb{R}^n \xrightarrow{V^*} \mathbb{R}^n \xrightarrow{\Sigma} \mathbb{R}^m \xrightarrow{U} \mathbb{R}^m \]

**Note**: \( U, V \) are orthogonal and \( \Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3, \ldots) \) is positive, decreasing.
Truncated Singular Value Decomposition (TSVD)

$A \approx U^r \Sigma^r V^{*r}$

- $U, V$ have orthonormal columns and $\Sigma$ is positive, diagonal, decreasing
- **Eckart–Young**: $r$-truncated SVD = best rank-$r$ Frobenius-norm approximation
- Approximately $r(m + n)$ degrees of freedom

**Applications:**

- Least-squares computations (linear regression)
- Principal component analysis (orthogonal regression; total least squares)
- Approximation, summarization, data reduction, visualization, ...
Randomized Matrix Computations
What’s Wrong with Classical TSVD Algorithms?

- Nothing... when the matrices are small

Climate Change

- Medium- to large-scale data (Gigabytes+)
- New architectures (multi-core, distributed, data centers, ...)
- New data presentations (off-core, dynamic, streaming)

The Role of Randomness

- Randomness is becoming a core tool for matrix computations
- Can solve problems that are impossible without randomness
- Can organize computations so they are cheaper (multiplication rich)
- Careful implementation and analysis remain essential!

Today: Practical randomized algorithms for TSVD computations
History of Randomized TSVD Algorithms

Classical Numerical Linear Algebra

- Random initialization for iterative methods (conventional wisdom)
- Guarantees for maximum eigenvalue (Dixon 1983; Kucziński & Woźniakowski 1992; ...)

Modern Numerical Linear Algebra

- Randomized low-rank matrix approximation (Martinsson, Rokhlin, Tygert 2004)
- One-pass matrix approximation (Woolfe, Liberty, Rokhlin, Tygert 2007)
- Randomized SVD framework, algorithms, and analysis (Halko, Martinsson, Tropp 2008–2011)
- Randomized block Krylov methods (Rokhlin & Tygert 2008; Halko et al. 2011; Tropp et al. 2017–2021)
- Practical streaming SVD algorithms (Tropp, Yurtsever, Udell, Cevher 2016–2019)
- Fast randomized algorithms for linear systems and eigenvalue problems (Nakatsukasa & Tropp 2021)
- Randomized block Krylov framework, algorithms, and analysis (Tropp & Webber, forthcoming)

CS Theory

- Gaussian dimension reduction (Johnson & Lindenstrauss 1984; Indyk & Motwani 1998)
- Randomized algorithms for LSI (Frieze, Kannan, Papadimitriou, Vempala 1998)
- Randomized linear algebra foundations (Drineas, Kannan, Mahoney 2004)
- Sketch-and-solve framework (Sarlós 2006)
- Streaming linear algebra foundations (Clarkson & Woodruff 2009)
- Qualitative analysis of randomized block Krylov methods (Musco & Musco 2015)
Spectral Decay in Scientific Data

Theme: Tradeoff between spectral decay and computational effort for TSVD
Randomized SVD Framework
Let $A \in \mathbb{R}^{m \times n}$ be an input matrix
Let $B \in \mathbb{R}^{n \times d}$ be a (random) test matrix

1. **Rangefinder**: Find orthobasis $Q \in \mathbb{R}^{m \times d}$ for range of input matrix:

   $$Q = \text{orth}(AB)$$

2. **Approximation**: Variational formulation of matrix approximation:

   $$\text{minimize}_{M \in \mathbb{R}^{d \times n}} \| A - QM \|_F^2$$

   **Solution**: $M_* = Q^* A$

3. **SVD**: Write rank-$d$ approximation $\hat{A}$ as an SVD:

   $$\hat{A} = QM_* = Q(U \Sigma V^*) = (QU) \Sigma V^*$$

Randomized SVD Prototype

Randomized SVD Framework

**Input:** Input matrix \( A \in \mathbb{R}^{m \times n} \) and (random) test matrix \( B \in \mathbb{R}^{n \times d} \)

**Output:** Approximate rank-\( d \) SVD in factored form: \( \tilde{A} = U \Sigma V^* \)

1. Form range sketch: \( Y = AB \)
2. Orthogonalize columns: \( Q = \text{orth}(Y) \)
3. Compress input matrix: \( M = Q^* A \)
4. Compute small dense SVD: \( M = \hat{U} \Sigma \hat{V}^* \)
5. Consolidate left unitary factor: \( U = Q \hat{U} \)

**Cost:** Two \( m \times n \times d \) matmuls with \( A \) and \( A^* \) plus \( (m + n)d^2 \) arithmetic

**Benefits:** Most arithmetic in matmuls (fast!) and very robust

**Questions:**
- What test matrix?
- Error bounds?

**Theme:** Tradeoff between spectral decay and computational effort
The HMT11 Randomized TSVD
Crazy Idea: Totally Random Test Matrix

\[ A \omega_1 \]

\[ A \omega_2 \]

\[ \text{range}(A) \]

Joel A. Tropp \( \text{(Caltech), Randomized TSVD Algorithms} \), UCLA EE Colloquium, 8 November 2021, Online.
Randomized SVD

Theorem (HMT 2011). Assume

- Input matrix $A \in \mathbb{R}^{m \times n}$
- Test matrix $B \in \mathbb{R}^{m \times d}$ is standard normal
- Form range sketch: $Q = \text{orth}(AB)$
- Compute rank-$d$ approximation: $\hat{A} = QQ^* A$

For $d \geq 1 + k/\varepsilon$, this randomized SVD algorithm guarantees

$$
\mathbb{E} \left\| A - \hat{A} \right\|_F^2 \leq (1 + \varepsilon) \cdot \left\| A - \left[ A \right]_k \right\|_F^2
$$

Key fact: Error is small when spectrum of $A$ decays quickly

Probability of a much larger error is negligible

Source: Halko et al. 2011, §10.2.
Example: Discretized Integral Operator

Matrix $A$ is a $200 \times 200$ discretization of

$$[S\sigma](x) = \text{const} \cdot \int_{\Gamma_1} \log \|x - y\| \sigma(y) \, dA(y)$$

for $x \in \Gamma_2$

Source: Halko et al. 2011, Figures 7.1–7.3.
Randomized Subspace Iteration (RSI)

- **Issue:** What if the spectrum decays more slowly?

- **RSI idea:** Adapt test matrix to input matrix by powering:
  - Draw standard normal matrix $\Omega \in \mathbb{R}^{n \times d}$
  - (Carefully) form test matrix $B = (AA^*)^q A\Omega$

- **Equivalent:** Apply [HMT11] to $(AA^*)^q A$. Enhances spectral decay!

- **RSI cost:** $2q + 2$ matmuls plus $(m + n)qd^2$ arithmetic

- **Spectral-norm error:** For $d = k + 2$ and $q \lesssim (\log n)/\varepsilon$,

\[
\mathbb{E} \left\| A - \hat{A} \right\| \leq (1 + \varepsilon) \cdot \left\| A - [A]_k \right\|
\]

- **In practice:** $q = 2$ or $q = 3$ often suffices. Not asymptotic!

Source: Halko et al. 2011, Algorithms 4.4 and 5.1, Corollary 10.10.
Example: Event Horizon Telescope

M87  Sgr A* (???)

Noise $\rightarrow$ SPDE $\rightarrow$ Random Field $+\rightarrow$ Static Envelope

advection  diffusion

Modal Decomposition of Dynamics

Target Random Field

Model 1

Model 2

Recovery of Black Hole Dynamics

Simulated Black Hole

Sources:

Noisy Full Data

EHT 2025

EHT 2017

Randomized Block Krylov
Randomized Block Krylov Methods

**Issue:** What if the spectrum does not really decay?

**BK idea:** Adapt test matrix to input matrix by including all powers:
- Draw standard normal matrix $\Omega \in \mathbb{R}^{n \times d}$
- (Carefully) form test matrix $B = [A\Omega \quad (AA^*)A\Omega \quad \ldots \quad (AA^*)^q A\Omega]$

**BK cost:** $2q + 2$ matmuls plus $(m + n)(qd)^2$ arithmetic

**Spectral-norm error:** For $d = k + 2$ and $q \lesssim (\log n) / \sqrt{\varepsilon}$,

$$E \left\| A - \hat{A} \right\| \leq (1 + \varepsilon) \cdot \left\| A - [A]_k \right\|$$

**In practice:** $q = 2$ or $q = 3$ often suffices. Not asymptotic!

**Loss:** Extra storage + orthogonalization

Sources: Rokhlin & Tygert 2008; Halko, Martinsson, Shkolnisky, Tygert 2011; Musco & Musco 2015; Tropp 2018–2021; Martinsson & Tropp 2020; Tropp & Webber 2021.
Example: Geophysical Imaging

Figure 6. Comparison between RTMs obtained by carrying out (a) conventional SVD, (b) the rSVD, and (c) BKI for $q = 1$. Difference plots with respect to the conventional SVD are included in (d and e). These plots show that the image obtained from a factorization based on BKI is significantly more accurate. Only the first eight singular values are used, i.e., $n_p = 8 \ll n_s$ with $n_s = 100$.

Source: Yang, Graff, Kumar, Herrmann 2021, Figure 6.
Sketchy SVD Framework

- Let $A \in \mathbb{R}^{m \times n}$ be an input matrix
- Let $B \in \mathbb{R}^{n \times d}$ be a (random) test matrix

1. **Rangefinder**: Find a basis $Q \in \mathbb{R}^{m \times d}$ for the range of the input matrix:

   $Y = AB$ and $Q = \text{orth}(Y)$

2. **Sketching**: For random $S \in \mathbb{R}^{s \times m}$, compressed matrix approximation:

   \[
   \minimize_{M \in \mathbb{R}^{d \times n}} \| S(A - QM) \|_F^2 \quad \text{Solution:} \quad M_* = (SQ)^\dagger (SA)
   \]

3. **SVD**: Write rank-$d$ approximation $\hat{A}$ as an SVD:

   $\hat{A} = QM_* = Q(U\Sigma V^*) = (QU)\Sigma V^*$

Example: One-Pass SVD

**Basic Sketchy SVD**

**Input:** Input matrix $A \in \mathbb{R}^{m \times n}$

**Output:** Approximate rank-$d$ SVD in factored form: $\hat{A} = U \Sigma V^*$

1. Form range sketch: $Y = AB$ for random $B \in \mathbb{R}^{n \times d}$
2. Form co-range sketch: $Z = SA$ for random $S \in \mathbb{R}^{s \times m}$
3. Orthogonalize columns of range sketch: $Q = \text{orth}(Y)$
4. Compute matrix approximation: $M = (SQ)^\dagger Z$
5. Compute small dense SVD: $M = \hat{U} \Sigma V^*$
6. Consolidate left unitary factor: $U = Q\hat{U}$

**Cost:** Two matmuls plus $(m + n)d^2$ arithmetic

**Benefits:** Takes linear measurements of matrix, only requires one pass

**Frobenius-norm error:** For $d = 1 + 2k$ and $s = 2 + 4k$,

$$\mathbb{E} \| A - \hat{A} \|_F \leq 4 \cdot \| A - [A]_k \|_F$$

**Loss:** Requires significant spectral decay

Reconstruction of von Kármán Street

Comments: Data: $m = 10,738; n = 5,001; 430$ MB. Algorithm: [TYUC19], sparse maps; rank $r = 5$; storage $T = 48(m + n)$. Compression: 71×.

Joel A. Tropp (Caltech), Randomized TSVD Algorithms, UCLA EE Colloquium, 8 November 2021, Online.
**Left Singular Vectors of von Kármán Street**

<table>
<thead>
<tr>
<th>Approximate [TYUC19]</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="example1.png" alt="Image" /></td>
<td><img src="example2.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="example3.png" alt="Image" /></td>
<td><img src="example4.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="example5.png" alt="Image" /></td>
<td><img src="example6.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="example7.png" alt="Image" /></td>
<td><img src="example8.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="example9.png" alt="Image" /></td>
<td><img src="example10.png" alt="Image" /></td>
</tr>
</tbody>
</table>

**Comments:** Data: $m = 10,738; n = 5,001; 430$ MB. Algorithm: [TYUC19], sparse maps; rank $r = 5$; storage $T = 48(m + n)$. Compression: $71 \times$.

Joel A. Tropp (Caltech), *Randomized TSVD Algorithms*, UCLA EE Colloquium, 8 November 2021, Online.
Singular Vectors of Sea Surface Temperature Data

Spatiotemporal Avg.

Seasonal

(Intra-)Seasonal

(Intra-)Seasonal

El Niño / La Niña

Comments: Data: $m = 691,150; n = 13,670; 75$ GB. Algorithm: [TYUC19], sparse maps; $k = 48; s = 839$. Compression ratio: $222 \times$.

Joel A. Tropp (Caltech), Randomized TSVD Algorithms, UCLA EE Colloquium, 8 November 2021, Online.
To learn more...

E-mail: jtropp@cms.caltech.edu

Web: http://users.cms.caltech.edu/~jtropp

Some papers: