Finding Structure with Randomness

Joel A. Tropp

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

Joint with P.-G. Martinsson and N. Halko
Applied Mathematics, Univ. Colorado at Boulder

Research supported in part by NSF, DARPA, and ONR

Top 10 Scientific Algorithms

list (here, the list is in chronological order; however, the articles appear in no particular order):

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

With each of these algorithms or approaches, there is a person or group receiving credit for inventing or

Source: Dongarra and Sullivan, *Comput. Sci. Eng.*, 2000.

The Decompositional Approach

"The underlying principle of the decompositional approach to matrix computation is that it is not the business of the matrix algorithmicists to solve particular problems but to construct computational platforms from which a variety of problems can be solved."

- A decomposition solves not one but many problems
- Often expensive to compute but can be reused
- Shows that apparently different algorithms produce the same object
- Facilitates rounding-error analysis
- Can be updated efficiently to reflect new information
- Has led to highly effective black-box software

Source: Stewart, 2000.

What's Wrong with Classical Methods?

- Nothing... unless the matrices are large-scale
- **Problem:** Major cost for numerical algorithms is data transfer
- **One Solution:** Design multiplication-rich algorithms
- Matrix multiplication is efficient in many architectures:
  - Graphics processing units
  - Multi-core and parallel processors
  - Distributed systems

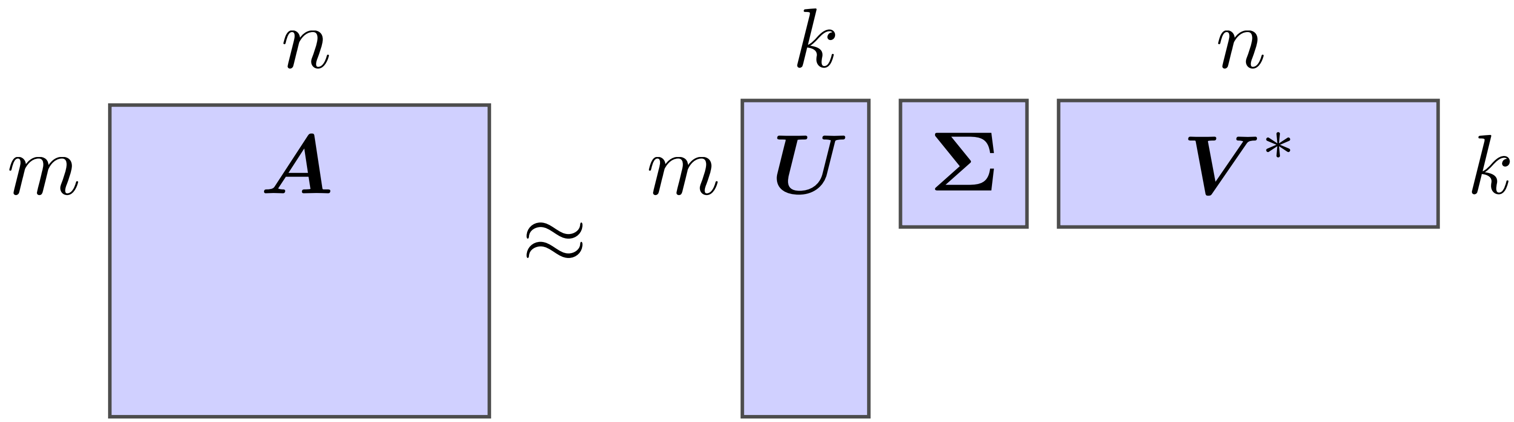
Source: Demmel and coauthors, 2003–present

Randomized

Truncated SVD

Truncated Singular Value Decomposition

$\boldsymbol{A} \approx \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$ where $\boldsymbol{U}, \boldsymbol{V}$ have orthonormal columns and $\boldsymbol{\Sigma}$ is diagonal:



Interpretation: k-SVD $=$ optimal rank-k approximation

Applications:

- Least-squares computations
- Principal component analysis
- Summarization and data reduction
- ...

Overview of Two-Stage Randomized Approach

Goal: Compute the k-SVD of an input matrix $\boldsymbol{A}$

Stage A: Finding the range

- Use a *randomized algorithm* to compute a k-dimensional basis $\boldsymbol{Q}$ that captures most of the range of $\boldsymbol{A}$:

$$\boldsymbol{Q} \text{ has orthonormal columns and } \boldsymbol{A} \approx \boldsymbol{Q}\boldsymbol{Q}^*\boldsymbol{A}.$$

Stage B: Constructing the decomposition

- Use the basis $\boldsymbol{Q}$ to reduce the problem size
- Apply *classical SVD algorithm* to the reduced problem
- Obtain k-SVD in factored form

Total Costs for Approximate k-SVD

Two-Stage Randomized Algorithm:

$$2 \text{ multiplies } (m \times n \times k) \quad + \quad k^2(m+n) \quad \text{flops}$$

Classical Sparse Methods (Krylov)**:**

$$k \text{ multiplies } (m \times n \times 1) \quad + \quad k^2(m+n) \quad \text{flops}$$

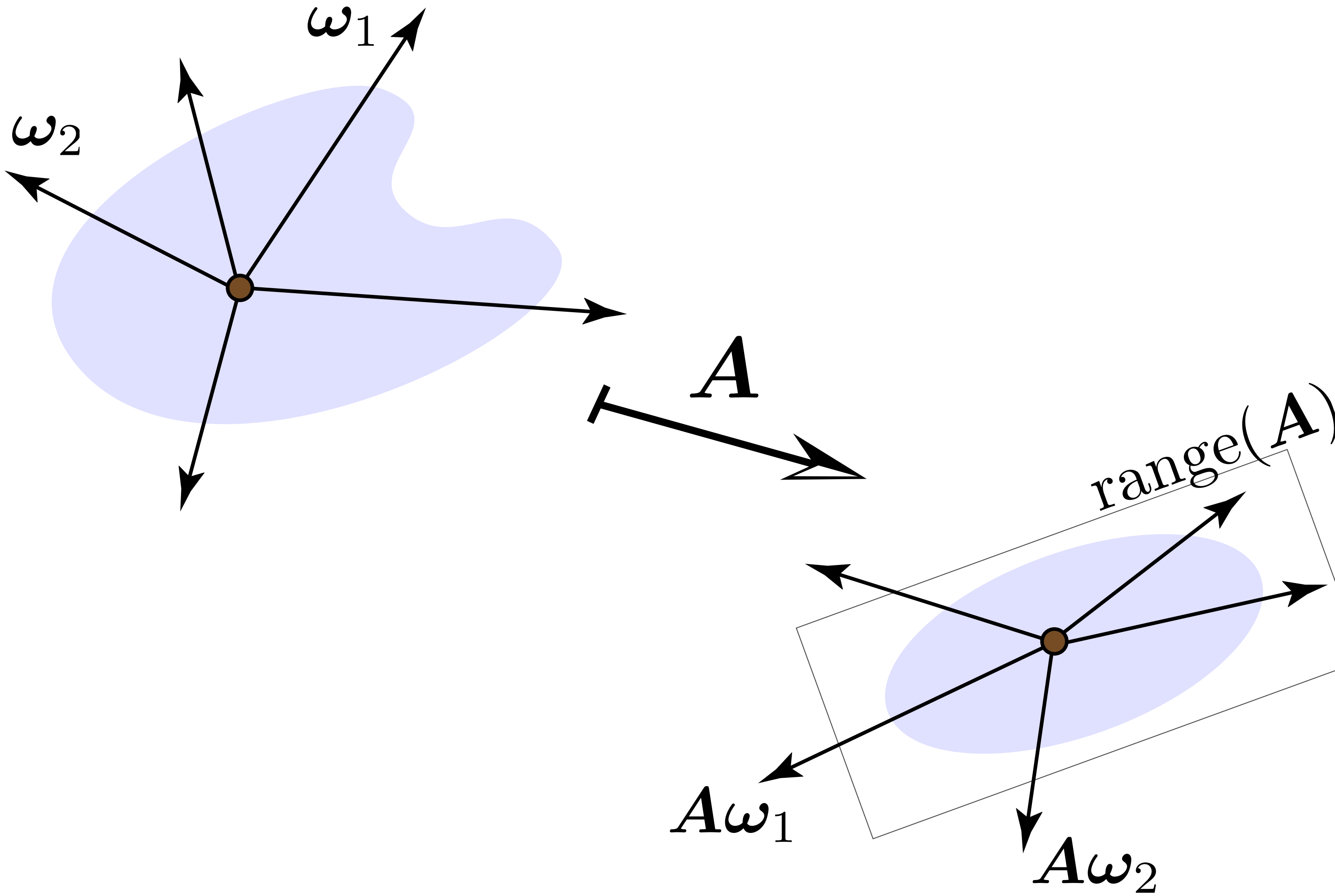
Classical Dense Methods (RRQR + full SVD)**:**

$$\text{Not based on multiplies} \quad + \quad mnk \quad \text{flops}$$

Similar approaches produce partial QR, Cholesky, column subsets, ...

Randomized
Range Finder

Randomized Range Finder: Intuition



Prototype for Randomized Range Finder

Input: An $m \times n$ matrix $\boldsymbol{A}$, number ℓ of samples

Output: An $m \times \ell$ matrix $\boldsymbol{Q}$ with orthonormal columns

1. Draw an $n \times \ell$ random matrix $\boldsymbol{\Omega}$.
2. Form the matrix product $\boldsymbol{Y} = \boldsymbol{A\Omega}$.
3. Construct an orthonormal basis $\boldsymbol{Q}$ for the range of $\boldsymbol{Y}$.

Total Cost: 1 multiply $(m \times n \times \ell) + \mathrm{O}(\ell^2 n)$ flops

Sources: NLA community: Stewart (1970s). GFA: Johnson–Lindenstrauss (1984) et seq. TCS: Boutsidis, Deshpande, Drineas, Frieze, Kannan, Mahoney, Papadimitriou, Sarlós, Vempala (1998–present). SciComp: Martinsson, Rokhlin, Szlam, Tygert (2004–present).

Implementation Issues

Q: How do we pick the number of samples ℓ?

A: Adaptively, using a randomized error estimator.

Q: How does the number ℓ of samples compare with the target rank k?

A: Remarkably, $\ell = k + 5$ or $\ell = k + 10$ is usually adequate!

Q: What random matrix $\mathbf{\Omega}$?

A: For many applications, standard Gaussian works brilliantly.

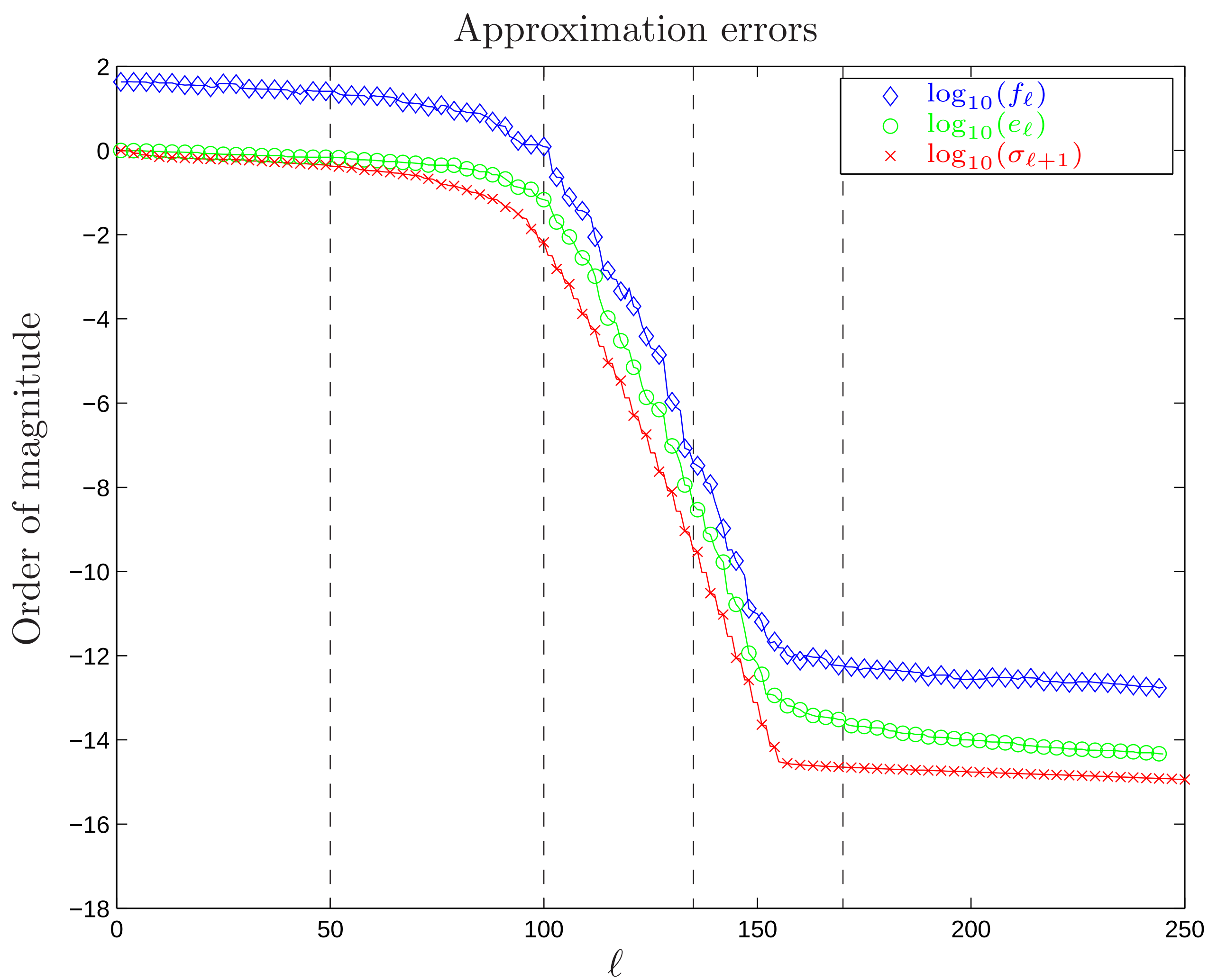
Q: How do we perform the matrix–matrix multiply?

A: Exploit the computational architecture.

Q: How do we compute the orthonormal basis?

A: Carefully... Double Gram–Schmidt or Householder reflectors.

Approximating a Helmholtz Integral Operator



Total number of samples

Error Bound for Randomized Range Finder

Theorem 1. [HMT 2009] **Assume**

- *the matrix* $\boldsymbol{A}$ *is* $m \times n$ *with* $m \geq n$;
- *the target rank is* k;
- *the optimal error* $\sigma_{k+1} = \min_{\operatorname{rank}(\boldsymbol{B}) \leq k} \|\boldsymbol{A} - \boldsymbol{B}\|$;
- *the test matrix* $\boldsymbol{\Omega}$ *is* $n \times (k+p)$ *standard Gaussian.*

Then *the basis* $\boldsymbol{Q}$ *computed by the randomized range finder satisfies*

$$\mathbb{E}\,\|\boldsymbol{A} - \boldsymbol{Q}\boldsymbol{Q}^*\boldsymbol{A}\| \leq \left[1 + \frac{4\sqrt{k+p}}{p-1} \cdot \sqrt{n}\right] \sigma_{k+1}.$$

The probability of a substantially larger error is negligible.

Randomized Range Finder + Power Scheme

Problem: The singular values of a data matrix often decay slowly

Remedy: Apply the randomized range finder to $(\boldsymbol{A}\boldsymbol{A}^*)^q \boldsymbol{A}$ for small q

---

Input: An $m \times n$ matrix $\boldsymbol{A}$, number ℓ of samples

Output: An $m \times \ell$ matrix $\boldsymbol{Q}$ with orthonormal columns

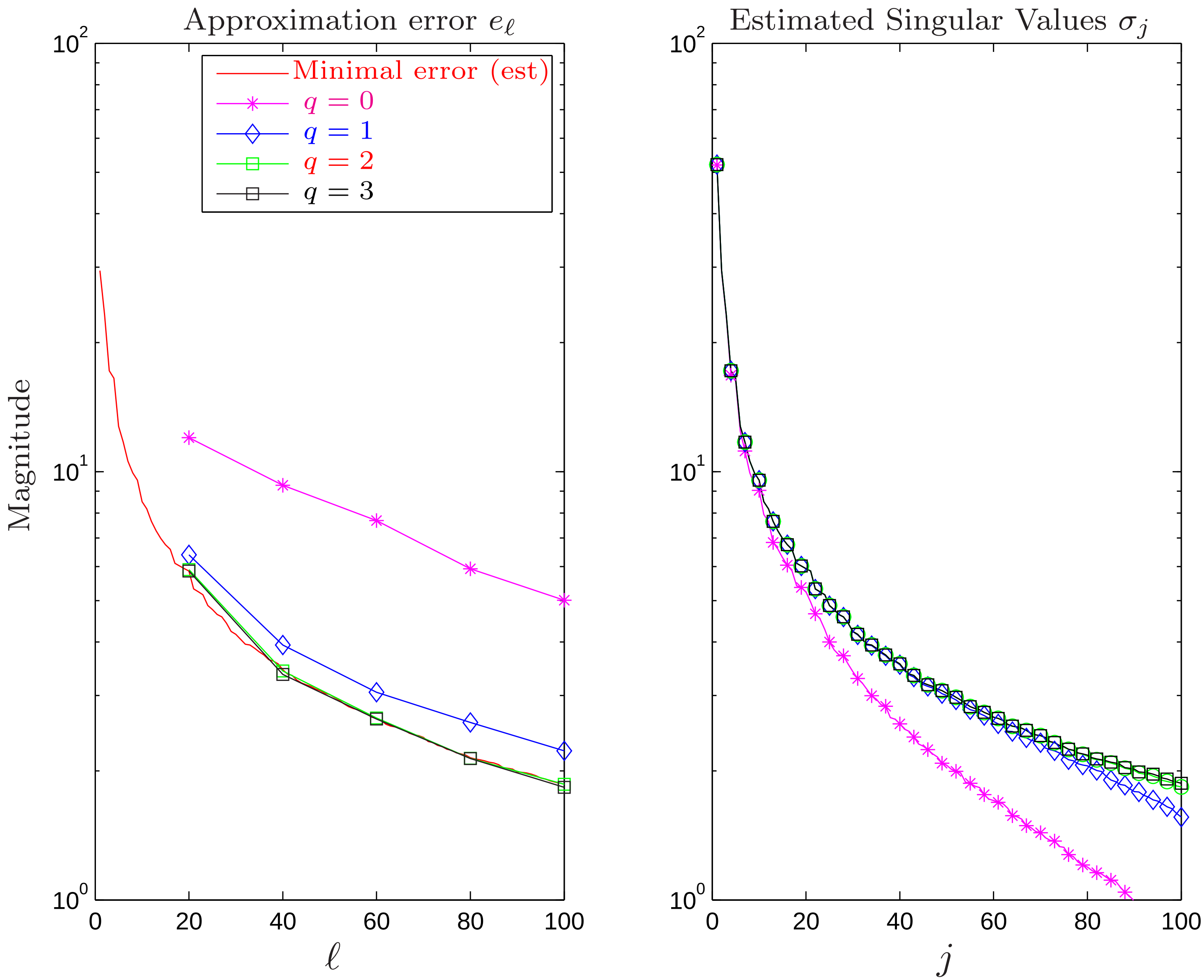
1. Draw an $n \times \ell$ random matrix $\boldsymbol{\Omega}$.
2. Carefully form the matrix product $\boldsymbol{Y} = (\boldsymbol{A}\boldsymbol{A}^*)^q \boldsymbol{A}\boldsymbol{\Omega}$.
3. Construct an orthonormal basis $\boldsymbol{Q}$ for the range of $\boldsymbol{Y}$.

Total Cost: $2q + 1$ multiplies $(m \times n \times \ell) + \mathrm{O}(q\ell^2 n)$

Eigenfaces

- Database consists of $7,254$ photographs with $98,304$ pixels each
- Form $98,304 \times 7,254$ data matrix $\widetilde{\boldsymbol{A}}$
- **Total storage:** 5.4 Gigabytes (uncompressed)
- Center each column and scale to unit norm to obtain $\boldsymbol{A}$
- The dominant left singular vectors are called **eigenfaces**
- Attempt to compute first 100 eigenfaces using power scheme

Source: Image from Scholarpedia article "Eigenfaces," accessed 12 October 2009



Total number of samples

Rank of singular value

Error Bound for Power Scheme

Theorem 2. [HMT 2009] Assume

- *the matrix* $\boldsymbol{A}$ *is* $m \times n$ *with* $m \geq n$;
- *the target rank is* k;
- *the optimal error* $\sigma_{k+1} = \min_{\operatorname{rank}(\boldsymbol{B}) \leq k} \|\boldsymbol{A} - \boldsymbol{B}\|$;
- *the test matrix* $\boldsymbol{\Omega}$ *is* $n \times (k+p)$ *standard Gaussian.*

Then *the basis* $\boldsymbol{Q}$ *computed by the power scheme satisfies*

$$\mathbb{E}\,\|\boldsymbol{A} - \boldsymbol{Q}\boldsymbol{Q}^*\boldsymbol{A}\| \leq \left[1 + \frac{4\sqrt{k+p}}{p-1} \cdot \sqrt{n}\right]^{1/(2q+1)} \sigma_{k+1}.$$

- The power scheme drives the extra factor to one exponentially fast!

To learn more...

E-mail: `jtropp@cms.caltech.edu`

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

Papers:

- HMT, "Finding structure with randomness: Probabilistic algorithms for computing approximate matrix decompositions," *SIREV* 2011
- T, "Improved analysis of the subsampled randomized Hadamard transform," *Adv. Adapt. Data Anal.*, to appear
- T, "User-friendly tail bounds for sums of random matrices," *Found. Comput. Math.*, to appear