A fast algorithm for approximating the SVD of a matrix

Joint work with Edo Liberty, Vladimir Rokhlin, and Franco Woolfe

Mark Tygert
Program in Applied Mathematics
Yale University

2007

Outline of the talk

1. Recent randomized work used in ours
2. Low-rank matrix approximation via SVDs and IDs
3. Steps in our algorithm
4. Empirical results of the algorithm
5. Proof of accuracy bounds
6. Applications and conclusions

Recent randomized work used in ours

- T. Sarlós (2006, 2007)
- N. Ailon and B. Chazelle (2006)
- A. Deshpande, P. Drineas, A. Frieze, R. Kannan, M. W. Mahoney, S. Muthukrishnan, and S. Vempala (2004, 2005, 2006)

Cost of the randomized algorithm

The algorithm typically needs $\mathcal{O}(mn \ln(k) + (m+n)\, l^2)$ floating-point operations in order to construct a nearly optimal rank-k approximation to the $m \times n$ matrix A.

The classical pivoted "$Q\,R$" decomposition algorithms such as Gram-Schmidt need at least kmn flops in order to construct a similarly accurate rank-k approximation.

The constant hidden by the $\mathcal{O}$-notation appears to be small enough so that the randomized algorithm is at least as fast as classical algorithms even when k is rather small or large.

SVD of a matrix, 1

Given an $m \times n$ matrix A, there exist

orthonormal $m \times 1$ vectors u^1, u^2, ..., u^{k-1}, u^k,

orthonormal $n \times 1$ vectors v^1, v^2, ..., v^{k-1}, v^k, and

nonnegative numbers $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{k-1} \geq \sigma_k$, such that

$$A = \sum_{j=1}^{k} u^j \, \sigma_j \, (v^j)^*, \tag{1}$$

where k is the rank of A.

SVD of a matrix, 2

In matrix notation,

$$A = U \Sigma V^*, \tag{2}$$

$$U = \left(\, u^1 \, \middle| \, u^2 \, \middle| \, \ldots \, \middle| \, u^{k-1} \, \middle| \, u^k \, \right), \tag{3}$$

$$V = \left(\, v^1 \, \middle| \, v^2 \, \middle| \, \ldots \, \middle| \, v^{k-1} \, \middle| \, v^k \, \right), \tag{4}$$

$$\Sigma_{i,j} = \begin{cases} \sigma_j, & i = j \\ 0, & i \neq j. \end{cases} \tag{5}$$

The columns of U are orthonormal;
the columns of V are orthonormal.

Low-rank matrix approximation

Pick a positive integer k less than the rank of A. Then,

$$\min_{\mathrm{rank}(B)=k} \|A - B\| = \sigma_{k+1}(A), \tag{6}$$

where $\|A - B\|$ denotes the spectral norm of $A - B$.

The minimum is attained by the following matrix:

$$B = \sum_{j=1}^{k} u^j \, \sigma_j(A) \, (v^j)^*. \tag{7}$$

Why low rank?

- Compression of subblocks of matrices arising from the discretization of smoothing integral operators and their inverses.

- Classic statistical method (principal component analysis):

- Noise removal.

- Data mining.

ID of a matrix, 1

An interpolative decomposition of an $m \times n$ matrix A consists of an $m \times k$ matrix E, and a $k \times n$ matrix B, such that

1. some subset of the rows of E makes up the $k \times k$ identity,

2. every entry of E has an absolute value of at most 2,

3. the rows of B constitute a subset of the rows of A, and

4. $A_{m \times n} = E_{m \times k} \cdot B_{k \times n}$.

ID of a matrix, 2

1. Gu and Eisenstat (1996) provided fail-safe and quite efficient algorithms for accurately approximating a numerically low-rank matrix with an interpolative decomposition.

2. Their fail-safe algorithms are based on the Cramer rule; algorithms based on pivoted, reorthogonalized Gram-Schmidt work well in most circumstances, and are a tad more efficient.

3. The ID approximation is accurate to about the square root of the matrix size times the best possible accuracy (that provided by the SVD).

Step 1 of the algorithm

In order to identify the range of A, we test A

on the l columns of a special random matrix R

(to be specified later in the talk), *i.e.*, we form

$$Y_{m\times l} = A_{m\times n} \cdot R_{n\times l}. \tag{8}$$

Step 2 of the algorithm

Via a pivoted $Q\,R$ algorithm (*e.g.*, Gu and Eisenstat, 1996), we choose k rows of Y which span the row space of Y to a precision $\approx \sigma_{k+1}(Y)$. The rows of A in the same positions span the row space of A to a precision of around $\sigma_{k+1}(A)$.

We assemble a $k \times l$ matrix Z from the k chosen rows of Y.

We assemble a $k \times n$ matrix B from the k chosen rows of A.

Step 3 of the algorithm

Via the $Q\,R$ decomposition of Y in Step 2,

we construct an $m \times k$ matrix E such that

$E_{m\times k}$ and $Z_{k\times l}$ together form an ID, and

$$\|E_{m\times k} \cdot Z_{k\times l} - Y_{m\times l}\| \approx \sigma_{k+1}(Y_{m\times l}). \tag{9}$$

Then, $E_{m\times k}$ & $B_{k\times n}$ constitute an ID, and

$$\|E_{m\times k} \cdot B_{k\times n} - A_{m\times n}\| \approx \sigma_{k+1}(A_{m\times n}) \tag{10}$$

with high probability.

A randomly subsampled randomized DFT

We define R to be the $n \times l$ random matrix given by

$$R_{n\times l} = \frac{1}{\sqrt{l}} \cdot D_{n\times n} \cdot F_{n\times n} \cdot S_{n\times l}, \tag{11}$$

where S is an $n \times l$ random matrix whose entries are 0's, aside from a single 1 placed uniformly at random in each column, F is the $n \times n$ discrete Fourier transform matrix, & D is a diagonal $n \times n$ matrix with i.i.d. diagonal entries, each of which is distributed uniformly over the unit circle.

Recap of the randomized algorithm

1. Form $Y_{m\times l} = A_{m\times n} \cdot R_{n\times l}$ ($R_{n\times l}$ is a randomly subsampled randomized DFT).

2. Find k rows of $Y_{m\times l}$ spanning most of the row space of $Y_{m\times l}$. Collect together these k rows of $Y_{m\times l}$ into the matrix $Z_{k\times l}$. Collect together the corresponding rows of $A_{m\times n}$ into $B_{k\times n}$.

3. Compute $E_{m\times k}$ such that $E_{m\times k}$ and $Z_{k\times l}$ constitute an ID, and $\|E_{m\times k} \cdot Z_{k\times l} - Y_{m\times l}\| \approx \sigma_{k+1}(Y_{m\times l})$.

Then, $E_{m\times k}$ and $B_{k\times n}$ constitute an ID, and with high prob.

$$\|E_{m\times k} \cdot B_{k\times n} - A_{m\times n}\| \approx \sigma_{k+1}(A_{m\times n}). \qquad (12)$$

Conversion from ID form to SVD form

1. Find $Q_{m\times k}$ with orthonormal columns and a triangular $R_{k\times k}$, such that $E_{m\times k} = Q_{m\times k} \cdot R_{k\times k}$.
2. Find $P_{n\times k}$ with orthonormal columns and a triangular $\Delta_{k\times k}$, such that $B_{k\times n} = \Delta_{k\times k} \cdot (P_{n\times k})^*$.
3. Form $C_{k\times k} = R_{k\times k} \cdot \Delta_{k\times k}$.
4. Find a diagonal $\Sigma_{k\times k}$ whose entries are all ≥ 0, a unitary T, and a unitary $W_{k\times k}$, such that $C_{k\times k} = T_{k\times k} \cdot \Sigma_{k\times k} \cdot (W_{k\times k})^*$.
5. Form $U_{m\times k} = Q_{m\times k} \cdot T_{k\times k}$ (so U has orthonormal columns).
6. Form $V_{n\times k} = P_{n\times k} \cdot W_{k\times k}$ (so V has orthonormal columns).

Combining the above formulae yields that

$$E_{m\times k} \cdot B_{k\times n} = U_{m\times k} \cdot \Sigma_{k\times k} \cdot (V_{n\times k})^*. \qquad (13)$$

Verification scheme, 1

We apply the difference Δ between the $m \times n$ matrix A to be approximated and its approximation to 6 random vectors $x^{(j)}$ whose entries are distributed as i.i.d. centered Gaussian r.v.'s.

Whenever $\|\Delta\| \geq 8\sqrt{n}\,\varepsilon$, then at least one of the six numbers

$$\frac{\|\Delta\, x^{(j)}\|}{\|x^{(j)}\|}, \quad j = 1, 2, 3, 4, 5, 6 \tag{14}$$

ends up greater than ε, aside from a one in a million chance. If $\|\Delta\|$ is even greater, the chance to exceed ε is even better. When $\|\Delta\| \leq \varepsilon$, all of the numbers (14) are always at most ε.

Therefore, we can with high probability filter out discrepancies with spectral norms meaningfully greater than ε, while always passing all discrepancies whose spectral norms are at most ε.

Verification scheme, 2

The power (or Lanczos) method has better guaranteed bounds than the tool on the previous slide when their operation counts are the same (Kuczyński & Woźniakowski, 1992; Dixon, 1983).

However, the power and Lanczos methods require successive applications of the matrix being approximated (as well as its transpose) to a series of vectors being generated on–the–fly, whereas the method of the preceding slide requires only the application of the matrix being approximated to a collection of independently generated vectors.

The scheme of the preceding slide therefore parallelizes easily.

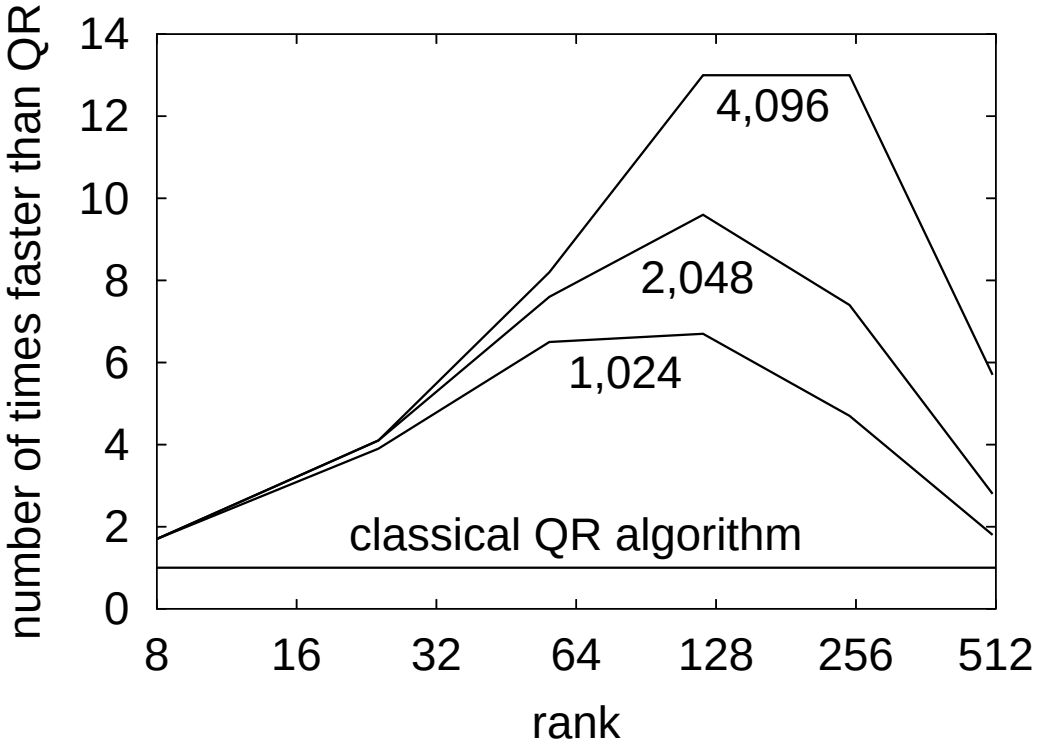
Cost of the randomized algorithm

The algorithm typically needs $\mathcal{O}(mn \ln(k) + (m+n)\, l^2)$ floating-point operations in order to construct a nearly optimal rank-k approximation to the $m \times n$ matrix A.

The classical pivoted "$Q\,R$" decomposition algorithms such as Gram-Schmidt need at least kmn flops in order to construct a similarly accurate rank-k approximation.

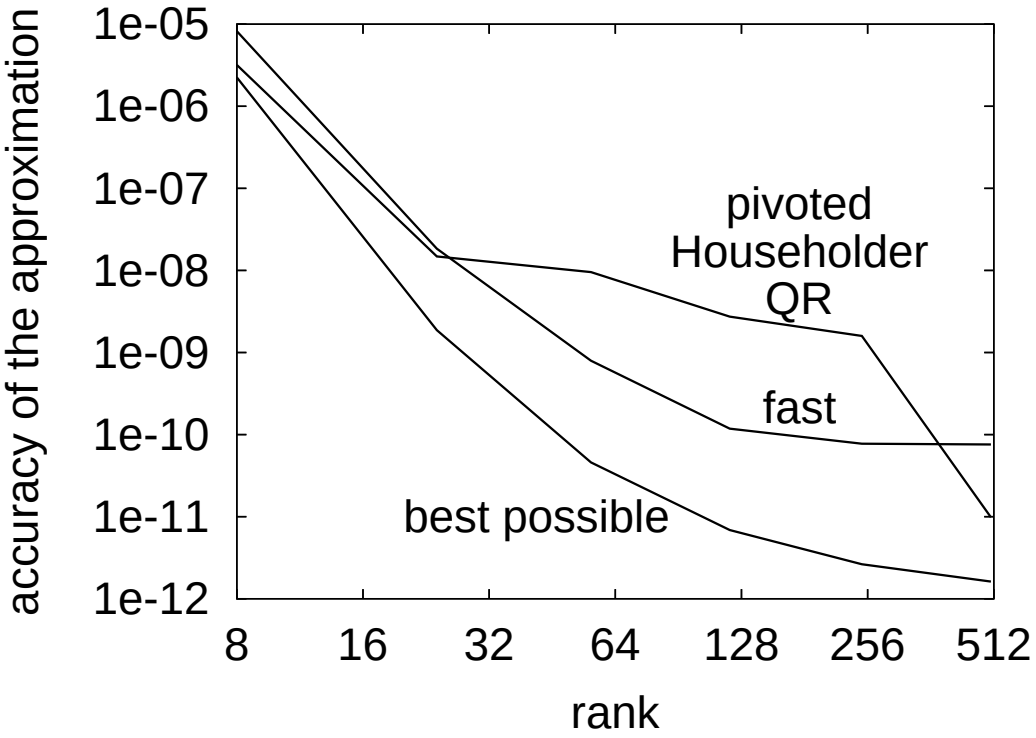
The constant hidden by the $\mathcal{O}$-notation appears to be small enough so that the randomized algorithm is at least as fast as classical algorithms even when k is rather small or large.

Speed gain on square matrices of various sizes



The time taken to verify the approximation is included in the fast, but not in the classical timings.

Empirical accuracy on 2,048-long convolution



The estimates of the accuracy of the approximation are accurate to at least two digits of relative precision.

Stratagem for proof of accuracy bounds

1. Apply the triangle inequality, breaking the accuracy bound into the sum of two contributions, one which involves optimization over a space of dimension k, and the other requiring the existence of a good bound in a high-dimensional space.

2. Observe that Steps 2 & 3 of the algorithm optimize over the space of dimension k directly via brute-force computations.

3. To prove the existence of a bound in the high-dimensional space, either appeal to Ailon-Chazelle ('06) & Sarlós ('06/7), or use their methods as motivation for a direct proof.

Proof of accuracy bounds, 1

Suppose A is an $m \times n$ matrix, R is an $n \times l$ matrix,

P is some $l \times n$ matrix, E is some $m \times k$ matrix, &

B is a $k \times n$ matrix whose rows are also rows of A.

Then, the triangle inequality yields that

$$\begin{aligned} & \|A - E\,B\| && (15)\\ \leq\ & \|A - A\,R\,P\| + \|A\,R\,P - E\,B\,R\,P\| + \|E\,B\,R\,P - E\,B\| && (16)\\ \leq\ & (1 + \|E\|)\,\|A - A\,R\,P\| + \|A\,R - E\,B\,R\|\,\|P\|. && (17) \end{aligned}$$

By choosing E, B, & P appropriately, we would like

$$\|A - E\,B\| \lesssim \sigma_{k+1}(A). \qquad (18)$$

Proof of accuracy bounds, 2

Again,

$$\|A - E\,B\| \leq (1 + \|E\|)\,\|A - A\,R\,P\| + \|A\,R - E\,B\,R\|\,\|P\|. \quad (19)$$

Step 2 chooses the rows of $Z = B\,R$ to be a subset of the rows of $Y = A\,R$, and Step 3 builds a matrix E, such that E and $Z = B\,R$ together constitute an ID of $Y = A\,R$, and

$$\|A\,R - E\,B\,R\| \leq \sqrt{4mk}\,\sigma_{k+1}(A\,R) \leq \sqrt{4mk}\,\|R\|\,\sigma_{k+1}(A). \quad (20)$$

Combining (19), (20), and the fact that E is part of an ID (so that $\|E\| \leq \sqrt{4mk}$) yields that

$$\|A - E\,B\| \leq (1 + \sqrt{4mk})\,\|A - A\,R\,P\| + \sqrt{4mk}\,\|P\|\,\|R\|\,\sigma_{k+1}(A). \quad (21)$$

Proof of accuracy bounds, 3

Again, formula (21) is

$$\|A - E\,B\| \leq (1 + \sqrt{4mk})\,\|A - A\,R\,P\| + \sqrt{4mk}\,\|P\|\,\|R\|\,\sigma_{k+1}(A). \tag{22}$$

But,

$$\|R\| = \|D\,F\,S\|/\sqrt{l} \leq \|D\|\,\|F\|\,\|S\|/\sqrt{l} \leq \sqrt{n}. \tag{23}$$

Hence, it suffices to show that in principle

there exists some $l \times n$ matrix P such that

$\|A - A\,R\,P\|$ is of the order of $\sigma_{k+1}(A)$, &

$\|P\|$ is not too large, with high probability.

Proof of accuracy bounds, 4

The key to showing that such a matrix P exists is the following

Lemma. Suppose that $\delta, \varepsilon \in \mathbf{R}$, and $k, l, n \in \mathbf{Z}$, such that $\delta, \varepsilon < 1$, $\delta, \varepsilon, k, l, n > 0$, and $n > l > (1 + 1/\varepsilon)^2 \, k^2/\delta$. Suppose also that U is a $k \times n$ matrix whose rows are orthonormal.

Then,

$$\sigma_1(U\,R) \leq \sqrt{1+\varepsilon} \tag{24}$$

$$1/\sigma_k(U\,R) \leq \sqrt{1+\varepsilon} \tag{25}$$

with probability at least $1 - \delta$.

Hence, when acting from the right on the row space of $U_{k\times n}$, $R_{n\times l}$ preserves norms up to a distortion factor of $\sqrt{1+\varepsilon}$.

Proof of accuracy bounds, 5

The lemma follows from the straightforward computation that

$$(U\,R)\,(U\,R)^* = \mathbf{1} + E, \tag{26}$$

where the expectation of the sum of the squares of the entries of the $k \times k$ matrix E is at most $k^2/l < \delta\,\varepsilon^2/(1+\varepsilon)^2$; therefore,

$$\|E\| < \varepsilon/(1+\varepsilon) \tag{27}$$

with probability at least $1-\delta$. Combining (26) and (27) yields

$$\|U\,R\|^2 = \|\mathbf{1} + E\| \leq 1 + \|E\| \leq 1 + \varepsilon \tag{28}$$

$$\left\|\Big((U\,R)\,(U\,R)^*\Big)^{-1}\right\| = \left\|(\mathbf{1}+E)^{-1}\right\| \leq \sum_{j=0}^{\infty} \|E\|^j \leq 1 + \varepsilon \tag{29}$$

with probability at least $1-\delta$.

Proof of accuracy bounds, 6

Simple (but tedious) linear-algebraic manipulations complete the proof of our current accuracy bounds. The basic idea is to choose an orthonormal basis for the row space of A as the rows of U in the lemma on the preceding two slides. Technically, the bound on the greatest singular value of $U\,R$ is not necessary.

However, please beware:

Our theoretical bounds require $l > k^2$, whereas our experimental tests always succeeded with $l > k + 5$. Of course, we can always run our verification check (for negligible cost) after each run of the algorithm in order to ascertain that the algorithm succeeded.

A peculiarity

We are able to recover A from $A\,R$ to precision $\approx \sigma_{k+1}(A)$ by multiplying the product $A\,R$ from the right by a matrix, say P, which depends on R, but is independent of $\sigma_1(A)$, ..., $\sigma_k(A)$.

In principle, we could recover A better by taking into account $\sigma_1(A)$, ..., $\sigma_k(A)$, but doing so makes the analysis unwieldy.

It seems to be sufficient to recover A solely by knowing which subspace constitutes A's range, without knowing the structure of A within that subspace.

Applications

- Compression of subblocks of matrices arising from the discretization of smoothing integral operators and their inverses.

- Classic statistical method (principal component analysis):

- Noise removal.

- Data mining.

Conclusion

There exists an algorithm for rank-k matrix approximation (or for computing the top k singular values and vectors) with advantages over the classical pivoted $Q\,R$ algorithms such as Gram-Schmidt:

1. Substantially faster (for most ranks k of the approximation), costing $\mathcal{O}(n^2 \ln(k) + nl^2)$ — not $\mathcal{O}(n^2 k)$ — for an $n \times n$ matrix.

2. Uses less storage when the input matrix is to be preserved, especially for matrices evaluated on-the-fly.

3. Reliably operates accurately on any matrix.

4. Parallelizes naturally.

mark.tygert@yale.edu http://www.cs.yale.edu/∼tygert