Fast Randomized Column Subset Selection Using Strong Rank-revealing QR

Alice Cortinovis and Lexing Ying

Abstract

Many large-scale matrices arising in applications have a low numerical rank, and while the truncated singular value decomposition gives a way to construct the *best* low-rank approximation with respect to all unitarily invariant norms, this is often too expensive to compute. For this reason, different types of low-rank approximation strategies have been analyzed in the literature, for example, approximations constructed from some rows and columns of the matrix. In practice, the strategy for choosing rows and columns depends on the properties and the size of the matrix. Several deterministic and randomized strategies for selecting rows and columns for CUR approximation have been developed; see, e.g., [1] for an overview.

This talk is concerned with the analysis of a randomized algorithm that selects suitable rows and columns. The algorithm is based on an initial uniformly random selection of rows and columns, followed by a refinement of this choice using a strong rank-revealing QR factorization. We show bounds on the error of the corresponding low-rank approximation (more precisely, the CUR approximation error) when the matrix is a perturbation of a low-rank matrix that can be factorized into the product of matrices with suitable incoherence and/or sparsity assumptions. The talk is based on the paper [2].

The column subset selection problem

Let $A \in \mathbb{R}^{n\times n}$ be the matrix we want to approximate (the discussion easily generalizes to rectangular matrices). Let us denote by $I, J \in \{1,\dots,n\}^\ell$ ordered index sets that correspond to rows and columns of A, respectively, for some $\ell \ll n$, and let us denote by $A(I,:) \in \mathbb{R}^{\ell\times n}$ and $A(:,J) \in \mathbb{R}^{n\times\ell}$ the submatrices of A corresponding to the rows indexed by I and the columns indexed by J, respectively. An approximation of A using these rows and columns has the form

$$A \approx A(:,J)MA(I,:),$$

for some matrix $M \in \mathbb{R}^{\ell\times\ell}$. The choice of M that minimizes the low-rank approximation error $\|A - A(:,J)MA(I,:)\|_F$ in the Frobenius norm is the orthogonal projection $M = A(:,J)^\dagger AA(I,:)^\dagger$, where $\dagger$ denotes the Moore-Penrose pseudoinverse of a matrix. The resulting approximation is usually called a "CUR approximation".

The quality of the low-rank approximation, that is, the norm of the error matrix $A - A(:,J)MA(I,:)$, depends on the choice of rows and columns, and can be bounded, in the spectral norm, by

$$\|A - A(:,J)MA(I,:)\|_2 \le \|A - A(:,J)A(:,J)^\dagger A\|_2 + \|A - AA(I,:)^\dagger A(I,:)\|_2, \tag{1}$$

where the two terms on the right-hand-side are the column and row subset selection error, respectively. For the remaining part of the talk, we focus on the problem of choosing columns, because the rows can be selected in the same way and the error of the corresponding CUR approximation is bounded as in (1).

The proposed strategy

The simplest method to select columns is to choose some columns uniformly at random, which gives good low-rank approximations in many cases of interest. In [3], it was shown that if A is a rank-k matrix that admits a low-rank decomposition with *incoherent* factors, uniform sampling of rows and columns allows to recover the matrix. Given a matrix $X \in \mathbb{R}^{n\times k}$ with orthonormal columns, the coherence of X is defined as

$$\mu := n \max_{\substack{1\le i\le n\\ 1\le j\le k}} |x_{ij}|^2,$$

and we say that X is μ-coherent. We say that a matrix is incoherent when μ is small. The concept of incoherence informally means that the information about the matrix is "evenly spread out" across all rows and columns.

The favorable property of uniform sampling can be extended to matrices that have low *numerical* rank [4]. When the matrix A does not satisfy these incoherence assumptions, heuristic approaches were considered, e.g., in [5, 6], where the idea is to refine the choice of the uniform sampled columns using a rank-revealing decomposition. The algorithm that we consider is the following.

Algorithm 1 Proposed algorithm for column subset selection

Require: Matrix A, number of indices ℓ_0, ℓ_a, ℓ_b
Ensure: Column index set J of cardinality $\ell_a + \ell_b$
1: Select ℓ_0 rows of A uniformly at random (index set I_0)
2: Select ℓ_a columns of $A(I_0, :)$ by sRRQR (index set J_a)
3: Select another ℓ_b columns of A uniformly at random (index set J_b)
4: Return the column index set $J = (J_a, J_b)$

Here, sRRQR denotes the strong rank-revealing QR factorization [7]. Informally, this is a partial pivoted QR factorization that ensures that the first ℓ_a columns of $A(I_0, :)$ are a good approximation of the range of the columns of $A(I_0, :)$. A rank-k sRRQR factorization for an $m \times n$ matrix can be computed in time $\mathcal{O}(mnk \log n)$, therefore the algorithm runs in time $\mathcal{O}(n\ell^2 \log n)$, where $\ell = \max\{\ell_0, \ell_a, \ell_b\}$; in particular, the cost is sublinear with respect to the size of the matrix.

When is there hope for Algorithm 1 to work?

Let us look at a few illustrative examples to see when Algorithm 1 is likely to return a good column set for low-rank approximation purposes. For example, if A is a matrix of all ones (and thus has rank 1), uniformly sampling just one single column gives a vector that spans the range of A. The singular vectors of A are as incoherent as they could possibly be. Now consider, instead, a matrix B which is made of zeros except for one entry: in this case, neither uniform sampling nor Algorithm 1 will be able to correctly locate the only important column with high probability. The singular vectors of B have coherence n, the highest possible value.

There is some interesting middle ground in which uniform sampling alone is not good enough, but the combination with sRRQR gives us a good column subset. For example, consider the case of a rank-2 matrix $C \in \mathbb{R}^{n\times n}$ that has entries $c_{1j} = c_{j1} = 1$ for $1 \le j \le n$ and zeros elsewhere. The row set I_0, chosen uniformly at random, will likely not include the first row. However, when looking at the matrix $C(I_0, :)$, the sRRQR algorithm will select a set J_a containing the first column, plus some other $\ell_a - 1$ columns sampled uniformly at random. Now, the set J will contain the first column

and at least another column; therefore, it is enough to span the range of C. We can decompose

$$C = \begin{bmatrix} \frac{1}{\sqrt{n}} & 1 \\ \frac{1}{\sqrt{n}} & 0 \\ \frac{1}{\sqrt{n}} & 0 \\ \vdots & \vdots \\ \frac{1}{\sqrt{n}} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{n} & 0 \\ 0 & \sqrt{n-1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{n-1}} & \frac{1}{\sqrt{n-1}} & \cdots & \frac{1}{\sqrt{n-1}} \end{bmatrix} = XZY^T.$$

Note that, for each $j = 1, 2$, one between the j-th column of X and the j-th column of Y is sparse and the other one is incoherent. This example suggests that when a matrix has a rank-k decomposition XZY^T (possibly, up to an additive error E), there is hope for Algorithm 1 to work when, for each $i = 1, \ldots, k$, one between the i-th columns of X and of Y is sparse, and the other is incoherent.

Analysis of column quality

Our analysis considers the case in which A has rank exactly k and the case in which A is a small perturbation of the exact case. For simplicity, we state our results in the perturbed case, with slightly simplified assumptions, and we omit explicit constants; the precise results are in our paper [2].

Assumptions. We assume that A admits an approximate rank-k factorization $A = XZY^T + E$, for some $X \in \mathbb{R}^{n\times k}$ and $Y \in \mathbb{R}^{n\times k}$, where X and Y have orthonormal columns, $Z \in \mathbb{R}^{k\times k}$ is diagonal, and the corresponding pairs of vectors of X and Y are either both incoherent (μ-coherent with a small value of μ) or one is sparse and the other one is incoherent. Moreover, we assume that $\|E\|_2 \leq \varepsilon$.

Main theorem. If the assumptions hold and we take ℓ_0, ℓ_a, ℓ_b to be a small multiple of μk, then the column index J returned by Algorithm 1 satisfies

$$\|A - A(:, J)A(:, J)^\dagger A\|_2 \leq \mathcal{O}\left(\varepsilon n \sqrt{\frac{k}{\ell}} \cdot \frac{\sigma_1(XZY^T)}{\sigma_k(XZY^T)}\right)$$

with high probability.

Sketch of proof ingredients. One important ingredient in the proof of our main result is the fact that selecting uniformly random rows from a matrix with orthonormal columns gives, with high probability, a well conditioned matrix [8]. The second ingredient is the sRRQR, which allows us to determine what are the most "important" columns in a given matrix (since this is used on a rectangular matrix which is much smaller than A, this is fast to do).

Intuitively, the columns corresponding to the index set J_a generated by lines 1 and 2 of Algorithm 1 are a good approximation to the part of A that corresponds to the pairs of vectors of X and Y that are of type (incoherent,incoherent) or (incoherent,sparse). The additional selection of ℓ_b uniformly random columns in line 3 ensures that, with high probability, also the information from the pairs of vectors of X and Y of type (sparse,incoherent) is taken care of.

Take-away messages and open questions

The analysis of Algorithm 1 shows that this combination of randomness and sRRQR is able to combine the speed of randomized algorithms with the reliability of sRRQR, for the matrices that admit a decomposition with the assumptions above. While it is difficult, in general, to check whether a matrix A admits a decomposition satisfying these assumptions, the objective of this talk is to shed some light on the excellent practical performance of simple sublinear-time algorithms for column and row subset selection. It is easier to think of XZY^T as the singular value decomposition of A or its best rank-k approximation, but actually, we do not require X and Y to have orthonormal columns, as long as they are well-conditioned. This flexibility allows us to apply our bounds to a larger class of matrices.

Our results do not cover all the matrices for which *there is hope*. For example, a scenario that is not covered by the current theory and is left for future work consists of matrices that have some pairs of vectors of X and Y for which one of them is incoherent and the other one does not have any specific assumption (that is, it may be coherent but not sparse).

It is possible to formulate an iterative version of Algorithm 1, such as the one considered in [6], in which one, after line 3, again performs an sRRQR factorization, adds some uniformly sampled rows, and then repeats this procedure a couple of times alternating between the selection of rows and columns. While the practical benefits of this "iterative refinement" for many matrices have been well documented, a theoretical analysis is still lacking and is an interesting direction for future research.

References

[1] P.-G. Martinsson and J. Tropp, Randomized numerical linear algebra: Foundations and algorithms, *Acta Numerica*, 29 (2020), pp. 403–572.

[2] A. Cortinovis and L. Ying, A sublinear-time randomized algorithm for column and row subset selection based on strong rank-revealing QR factorizations, *SIAM J. Matrix Anal. Appl. (to appear)*, 2024.

[3] A. Talwalkar and A. Rostamizadeh, Matrix coherence and the Nyström method, in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, 2010, pp. 572–579.

[4] J. Chiu and L. Demanet, Sublinear randomized algorithms for skeleton decompositions, *SIAM J. Matrix Anal. Appl.*, 34 (2013), pp. 1361–1383.

[5] Y. Li, H. Yang, E. R. Martin, K. L. Ho, and L. Ying, Butterfly factorization, *Multiscale Model. Simul.*, 13 (2015), pp. 714–732.

[6] J. Xia, Making the Nyström method highly accurate for low-rank approximations, *SIAM J. Sci. Comput.*, 46 (2024), pp. A1076–A1101.

[7] M. Gu and S. C. Eisenstat, Efficient algorithms for computing a strong rank-revealing QR factorization, *SIAM J. Sci. Comput.*, 17 (1996), pp. 848–869.

[8] J. A. Tropp, Improved analysis of the subsampled randomized Hadamard transform, *Adv. Adapt. Data Anal.*, 3 (2011), pp. 115–126.