Spectral Gap Error Bounds for Improving CUR Matrix Decomposition and the Nyström Method

David G. Anderson
Department of Mathematics

Simon S. Du
Department of Mathematics

Michael W. Mahoney
Department of Statistics

Christopher Melgaard
Department of Mathematics

Kunming Wu
Department of Mathematics

Ming Gu
Department of Mathematics

University of California, Berkeley

Abstract

The CUR matrix decomposition and the related Nyström method build low-rank approximations of data matrices by selecting a small number of representative rows and columns of the data. Here, we introduce novel *spectral gap error bounds* that judiciously exploit the potentially rapid spectrum decay in the input matrix, a most common occurrence in machine learning and data analysis. Our error bounds are much tighter than existing ones for matrices with rapid spectrum decay, and they justify the use of a constant amount of oversampling relative to the rank parameter k, i.e, when the number of columns/rows is $\ell = k + O(1)$. We demonstrate our analysis on a novel deterministic algorithm, *StableCUR*, which additionally eliminates a previously unrecognized source of potential instability in CUR decompositions. While our algorithm accepts any method of row and column selection, we implement it with a recent column selection scheme with strong singular value bounds. Empirical results on various classes of real world data matrices demonstrate that our algorithm is as efficient as, and often outperforms, competing algorithms.

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

1 Introduction

The CUR matrix decomposition approximates an arbitrary data matrix by selecting a subset of columns and a subset of rows to form a low-rank approximation [7, 13]. This method overcomes a fundamental drawback of standard PCA: that the principal components are dense. Dense components suffer from two main disadvantages: loss of sparsity and reduced interpretability. On the other hand, the CUR decomposition is a product of three matrices: two ($\mathbf{C}$ and $\mathbf{R}$ with c sampled columns and r sampled rows of $\mathbf{A}$ respectively) preserve the sparsity of the data matrix, while the third ($\mathbf{U}$) is a relatively small dense matrix. Thus the CUR approximation is cheaper to work with and to store. C and R also imply critical information in some applications [7, 13].

Notable applications of CUR include bioinformatics, document classification, image and video processing, securities trading, and web graphs [2, 13, 14, 15, 16]. Furthermore, the CUR decomposition is widely studied in machine learning because the Nyström method is a special case of CUR. The Nyström method approximates large kernel matrices that are used for kernel methods, manifold learning, and dimension reduction [6, 16, 17, 18, 19, 20]. In particular, the recent work of [9] introduced an efficient leverage-based random sampling algorithm for Nyström approximation that is analyzed simultaneously for both the spectral and Frobenius norms, while other recent work requires separate algorithms depending on the choice of norm. CUR is also a natural extension of the CX decomposition, which selects either columns or rows, but not both, of the data matrix, and which has been studied in [3, 11].

In general, these works seek to obtain improved *multiplicative error bounds*, which are of the form

$$\|\mathbf{A} - \mathbf{CUR}\|_\xi \leq f\,(m,n,k,c,r)\,\|\mathbf{A} - \mathbf{A}_k\|_\xi,$$

where $\xi \in \{2, F\}$, and where f is a polynomial function and $\mathbf{A}_k$ is an optimal rank-k approximation to a given $\mathbf{A} \in \mathbb{R}^{m\times n}$. When f does not depend on m and n, these bounds are called *constant factor bounds* [12]. Recent works have also established *relative error bounds*, where $f \approx 1+\epsilon$ for a selection of roughly $O(k/\epsilon)$ rows and columns [4, 7, 9, 12, 16, 17].

Regardless of the form of the guarantee, there are two main drawbacks to the practical use of these existing approaches to CUR decompositions and the Nyström method: choosing $\ell \gtrsim O(k/\epsilon)$ columns/rows is often not practical, and thus one typically chooses $\ell = k + O(1)$, i.e., many fewer columns/rows than the sufficient conditions required by the worst-case theory; and, additionally, no known results adapt these methods specifically to matrices with rapidly decaying singular values. Because most data matrices to which CUR decompositions have been applied have decaying singular values, and because a decaying spectrum facilitates better approximations, CUR decompositions would greatly benefit from analysis connecting the quality of the approximation to the rate of spectral decay.

We introduce powerful *spectral gap error bounds* that solve these two related problems. We perform a more refined analysis based on the spectrum of the input data, and present bounds of the form

$$\|\mathbf{A} - \mathbf{CUR}\|_\xi^2 \leq \left(1 + O\left(\tau^2\right)\right)\|\mathbf{A} - \mathbf{A}_k\|_\xi^2,$$

for $\xi \in \{2, F\}$, where k is the target rank and τ is a quantity that depends on the singular value rate of decay of $\mathbf{A}$ and the amount of oversampling. For matrices with rapidly decaying singular values, and as a function of the amount of oversampling, $\tau \ll 1$. Thus, unlike previous work, our error bounds are near-optimal for matrices with rapidly decaying spectra, and the approximations achieve optimality in the limit as the rates of decay of the spectra increase. (Such a result is a natural requirement for a good approximation method, but none have proved this.) These bounds also help explain why it is acceptable to use a constant $O(1)$ amount of oversampling, i.e., why, given a desired rank k, one can sample $c = k + O(1)$ columns and/or $r = k + O(1)$ rows.

We also show that CUR can be unstable, and we develop a novel algorithm, *StableCUR*, that completely avoids this instability. This algorithm accepts any $\mathbf{C}$ and $\mathbf{R}$ matrices from any row and column selection algorithm, and avoids calculating $\mathbf{U}$, which we show can be ill-conditioned. We apply the column selection algorithm from [1] to determine $\mathbf{C}$ and $\mathbf{R}$, and then we apply our algorithm to compute a CUR decomposition in a stable form. Also, we compare the performance of the combination of these two algorithms to existing randomized CUR algorithms. For input matrices with rapidly decaying spectra, and when performing only a constant amount of oversampling relative to the rank k, our CUR algorithm combined with the algorithm from [1] achieves improved error bounds, improved computational complexity, and reduced storage compared to current methods. We also provide a brief empirical illustration of how deterministic and randomized CUR decompositions perform as a function of the oversampling parameter for matrices for which the spectrum decays quickly, as well as when it decays slowly.

2 Preliminaries

In this section we review previous results and important theorems to be used in our main results.

2.1 The CUR Decomposition

Given a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with rank ρ and a target rank k, we choose a subset of columns $\mathbf{C} \in \mathbb{R}^{m\times c}$, a subset of rows $\mathbf{R} \in \mathbb{R}^{r\times n}$ and compute a matrix $\mathbf{U} \in \mathbb{R}^{c\times r}$ so that $\tilde{\mathbf{A}} = \mathbf{CUR}$ approximates $\mathbf{A}$, where $k < c \ll n$ and $k < r \ll m$. Thus only $\mathbf{C}$, $\mathbf{U}$, and $\mathbf{R}$ need to be stored, which are much smaller than the original matrix $\mathbf{A}$. Additionally, $\mathbf{C}$ and $\mathbf{R}$ retain the sparsity of the original matrix.

2.2 Notation

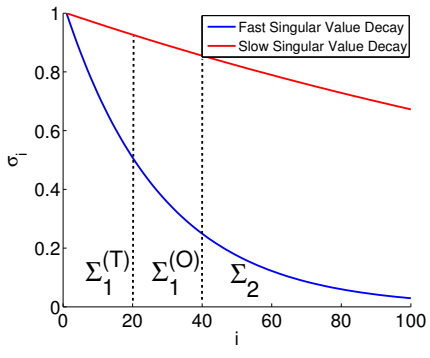

Figure 1: Illustration of Σ's when $k = 20$ and $p = 40$. Quick singular value decay implies we can choose $k + O(1)$ columns and rows for small residual errors.

We exploit the potential decay in the singular values of $\mathbf{A}$ for better computational efficiency and decomposition reliability. Consider a parameter p such that $k \leq p < \min(c, r)$. In the SVD of $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, we partition $\mathbf{U}$ and $\mathbf{V}$ as

$$\mathbf{U} = \begin{matrix} \\ m \end{matrix} \overset{\begin{matrix} p & \rho - p \end{matrix}}{\begin{pmatrix} \mathbf{U}_1 & \mathbf{U}_2 \end{pmatrix}}, \; \mathbf{V} = \begin{matrix} \\ n \end{matrix} \overset{\begin{matrix} p & \rho - p \end{matrix}}{\begin{pmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{pmatrix}}. \quad (1)$$

Let $\mathbf{\Sigma} = \mathbf{diag}(\sigma_1, \cdots, \sigma_\rho)$, $\sigma_1 \geq \cdots \geq \sigma_\rho > 0$ with

$$\mathbf{\Sigma} = \begin{matrix} p \\ \rho - p \end{matrix} \overset{\begin{matrix} p & \rho - p \end{matrix}}{\begin{pmatrix} \mathbf{\Sigma}_1 & \\ & \mathbf{\Sigma}_2 \end{pmatrix}}, \; \mathbf{\Sigma}_1 = \begin{matrix} k \\ p - k \end{matrix} \overset{\begin{matrix} k & p - k \end{matrix}}{\begin{pmatrix} \mathbf{\Sigma}_1^{(\mathbf{T})} & \\ & \mathbf{\Sigma}_1^{(\mathbf{O})} \end{pmatrix}}.$$

In equation (1), $\mathbf{U}_1$ and $\mathbf{V}_1$ comprise p orthonormal columns spanning leading p-dimensional row space and column space respectively. The largest k singular values of $\mathbf{A}$ are contained in the diagonal matrix $\mathbf{\Sigma}_1^{(\mathbf{T})}$, which in turn is contained in $\mathbf{\Sigma}_1$; the $(p+1)$-th through the ρ-th singular values of $\mathbf{A}$ are contained in $\mathbf{\Sigma}_2$. The value of p is chosen to create a "spectrum gap" between the kth and $(p+1)$th singular values of $\mathbf{A}$. To the best of our knowledge, such a partition was first introduced in [10]. Section 3 will show that if this gap is large, then the rank-k CUR approximation differs from the best possible rank-k approximation by a negligible amount.

Based on the SVD, the row statistical leverage scores and the row coherence relative to the best rank-p approximation to $\mathbf{A}$ are defined through the p leading left singular vectors in $\mathbf{U_1}$:

$$l_{rj} = ||\mathbf{U}_1(j,:)||^2, \quad \mu_r = \frac{m}{p} \times \max_{j \in \{1,\ldots m\}} l_i^r. \quad (2)$$

Similarly, the column statistical leverage scores and the column coherence relative to the best rank-p approximation to $\mathbf{A}$ are defined through the p leading right singular vectors in $\mathbf{V_1}$:

$$l_{cj} = ||\mathbf{V}_1(j,:)||^2, \quad \mu_c = \frac{n}{p} \times \max_{j \in \{1,\ldots n\}} l_i^c. \quad (3)$$

The *Moore-Penrose inverse* of $\mathbf{A}$ is $\mathbf{A}^\dagger = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$. For $\mathbf{A} \in \mathbb{R}^{m \times n}$(with $m \geq n$) it takes $O(mn^2)$ flops to compute the SVD and QR decomposition, $O(mnk)$ to compute the truncated SVD of rank-k, and $O(mn \ln n)$ flops to compute leverage scores [5].

2.3 The Sketching Model

Let $\mathbf{\Pi}_r \in \mathbb{R}^{m \times r}$ and $\mathbf{\Pi}_c \in \mathbb{R}^{n \times c}$ be row and column sketching matrices. Examples include sampling matrices that select a subset of columns and rows of $\mathbf{A}$ and Gaussian matrices which produce matrices $\mathbf{C}$ and $\mathbf{R}$ that are Gaussian mixtures of columns and rows of $\mathbf{A}$. Take $\mathbf{C} = \mathbf{A}\mathbf{\Pi}_c$, $\mathbf{R} = \mathbf{\Pi}_r^T\mathbf{A}$, and $\mathbf{U} = \mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger$. Then the CUR approximation is defined as $\widetilde{\mathbf{A}} = \mathbf{CUR}$, and $\widetilde{\mathbf{A}}_k = (\mathbf{CUR})_k$ is an approximation to $\mathbf{A}$ with rank at most k. Following [9], and for completeness, we formulate our main theoretical result in terms of arbitrary "sketching" matrices. Let

$$\mathbf{\Psi}_1 := \mathbf{U}_1^T\mathbf{\Pi}_r \quad \text{and} \quad \mathbf{\Psi}_2 := \mathbf{U}_2^T\mathbf{\Pi}_r.$$

Intuitively, $\mathbf{\Psi_2}\mathbf{\Psi_1^\dagger}$ defines the tangents of the angles between the spaces spanned by $\mathbf{U}_1$ and $\mathbf{\Pi}_r$ [9]. These angles should be sufficiently acute for $\mathbf{\Pi}_r$ to be a good sketch matrix. Similarly let,

$$\mathbf{\Omega}_1 := \mathbf{V}_1^T\mathbf{\Pi}_c \quad \text{and} \quad \mathbf{\Omega}_2 := \mathbf{V}_2^T\mathbf{\Pi}_c,$$

and $\mathbf{\Omega_2}\mathbf{\Omega_1^\dagger}$ defines the tangents of the angles between the spaces spanned by $\mathbf{V}_1$ and $\mathbf{\Pi}_c$.

When considering the modified Nyström method for positive semi-definite $\mathbf{A}$ instead of the CUR approximation, we will only use $\mathbf{\Pi}_c$ and $\mathbf{\Omega}$, and we set the other side by $\mathbf{R} = \mathbf{C}^T$.

2.4 Notion of Optimality

We employ the follow metric of approximation optimality due to Eckart and Young. Let

$$\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$$

be the rank-k truncated SVD of a data matrix $\mathbf{A}$.

Theorem 1. (Eckart-Young)

$$\mathbf{A}_k = \underset{rank(\mathbf{B}) \leq k}{\arg\min} \|\mathbf{A} - \mathbf{B}\|_2 = \underset{rank(\mathbf{B}) \leq k}{\arg\min} \|\mathbf{A} - \mathbf{B}\|_F,$$

with

$$\|\mathbf{A} - \mathbf{A}_k\|_2 = \sigma_{k+1}, \quad \|\mathbf{A} - \mathbf{A}_k\|_F = \sqrt{\sum_{j=k+1}^{\rho} \sigma_j^2}.$$

2.5 Deterministic Column-Selection

In this section we describe the deterministic Unweighted Column Selection (UCS) algorithm of [1], which will be used in our main results. Applied to a given a matrix $\mathbf{V}^T \in \mathbb{R}^{p \times n}$ with orthonormal rows, this greedy algorithm attempts to choose a subset π of columns to maximize $\sigma_{\min}\left(\mathbf{V}^T(:,\pi)\right)$. The previous column selection algorithm of [3] requires two

input matrices and outputs a weighted column selection, for which the weights could be arbitrary. The algorithm of [1] requires a single, relatively small input matrix and outputs an unweighted column selection, while also proving tighter error bounds. The fact that column selection algorithm of [3] requires two matrices to work on makes it less efficient than UCS in complexity and memory use. Consider the matrix $\mathbf{V}_1^T$ in equation (1). We refer to the i^{th} column as $\vec{u}_i \in \mathbb{R}^p$. Then the UCS algorithm is summarized as follows: starting with a p-by-p matrix $B = 0$ and a parameter $T > 0$, the UCS algorithm iteratively selects ℓ columns of $\mathbf{V}_1^T$ by iterating:

- solve for the unique $\lambda < \lambda_{\min}(B)$ such that
$$\mathbf{tr}\,(B - \lambda I)^{-1} = T, \tag{4}$$
- solve for the unique $\widehat{\lambda} < \lambda_k$ that satisfies
$$\left(\widehat{\lambda} - \lambda\right)\left(n - r + \sum_{j=1}^{p} \frac{1-\lambda_j}{\lambda_j - \lambda}\right) = \frac{\sum \frac{1-\lambda_j}{(\lambda_j-\lambda)(\lambda_j-\widehat{\lambda})}}{\sum \frac{1}{(\lambda_j-\lambda)(\lambda_j-\widehat{\lambda})}}, \tag{5}$$
where λ_j is the j^{th} eigenvalue of B,
- find an index i, not already selected, such that
$$\mathbf{tr}\left(B - \widehat{\lambda} I + \vec{u}_i \vec{u}_i^T\right) \leq \mathbf{tr}\,(B - \lambda I)^{-1} \tag{6}$$
- reset $B := B + \vec{u}_i \vec{u}_i^T$.

Theorem 2. *An index $i \notin \Pi$ can always be found to satisfy condition (6).*

Carried out efficiently, each i can be computed in $O(p^2 n)$ operations. We summarize the above procedure in Algorithm 1. It can be shown that

$$\lambda_{\min}\left(B_\ell\right) \geq \frac{(\sqrt{\ell} - \sqrt{p})^2}{(\sqrt{n-p} + \sqrt{\ell})^2 + (\sqrt{\ell} - \sqrt{p})^2}. \tag{7}$$

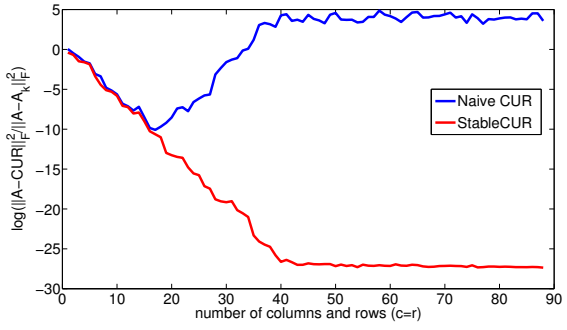

Figure 2: Stability comparison of the naive CUR algorithm and our proposed stable sketch algorithm.

Algorithm 1 Unweighted Column Selection (UCS)

Inputs: Row-orthonormal matrix $\mathbf{V}_1^T \in \mathbb{R}^{p\times n}$, $T \in \mathbb{R}^+$, $\ell, p \in \mathbb{N}$ s.t. $k \leq p < \ell$
Outputs: Index set Π and matrix B.
1: Set $B_0 = 0_{p\times p}$, $\Pi_0 = \phi$
2: **for** $t = 0, \cdots, \ell - 1$ **do**
3: Solve for λ using equation (4)
4: Calculate $\widehat{\lambda}$ using equation (5)
5: Find $i \notin \Pi$ such that inequality (6) is satisfied with $\vec{u}_i$
6: Update $B_{t+1} := B_t + \vec{u}_i \vec{u}_i^T$ and $\Pi := \Pi \cup \{i\}$.
7: **end for**

3 Theoretical Results

In this section, we present our *StableCUR* algorithm and our *spectral gap error bounds*.

3.1 The StableCUR Algorithm

Directly computing $\widetilde{\mathbf{A}}$ by multiplying $\mathbf{C}, \mathbf{U}, \mathbf{R}$ together is not numerically stable. Here, we present a new algorithm, *StableCUR*, to construct $\widetilde{\mathbf{A}}$. Each step of this procedure is numerically stable, and standard libraries exist for both QR and SVD.

Algorithm 2 StableCUR

Inputs: $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{R} \in \mathbb{R}^{r\times m}$, $\mathbf{C} \in \mathbb{R}^{n\times c}$, target rank k
Outputs: $\widetilde{\mathbf{A}} \in \mathbb{R}^{m\times n}$ and $\widetilde{\mathbf{A}}_k \in \mathbb{R}^{m\times n}$
1: Do QR factorization on $\mathbf{R}^T$ to obtain a basis of rows of $\mathbf{R}$, $\mathbf{R} = \mathbf{R}_r \mathbf{Q}_r$
2: Do QR factorization on $\mathbf{C}$ to obtain a basis of columns of $\mathbf{C}$, $\mathbf{C} = \mathbf{Q}_c \mathbf{R}_c$
3: $\mathbf{B} = \mathbf{Q}_c^T \mathbf{A} \mathbf{Q}_r^T$
4: $\widetilde{\mathbf{A}} = \mathbf{Q}_c \mathbf{B} \mathbf{Q}_r$
5: Do SVD on $\mathbf{B}$ to Compute $\mathbf{B}_k$.
6: $\widetilde{\mathbf{A}}_\mathbf{k} = \mathbf{Q}_c \mathbf{B}_k \mathbf{Q}_r$

In Figure 2 we compare the naive procedure and our stable procedure on a synthesized matrix whose ith singular value is 2^{-i}. The naive computations could lead to inaccurate results because as the number of columns and rows in $\mathbf{C}$ and $\mathbf{R}$ increase, these matrices capture a greater amount of the singular values of $\mathbf{A}$, and so $\mathbf{U} = \mathbf{C}^\dagger \mathbf{A} \mathbf{R}^\dagger$ can be ill-conditioned. Although the algorithm above performs QR on both $\mathbf{C}$ and $\mathbf{R}$, QR for either $\mathbf{C}$ or $\mathbf{R}$ is all that is necessary to make it stable.

3.2 Spectral Gap Error Bounds

Here, we introduce our *spectral gap error bounds*, theorems about accuracy in the individual singular values and error bounds in the spectral and Frobenius norms for the CUR sketching model. Theorems 3 and 4 below are stated in terms of the following upper bounds:

$$\mathcal{C}_\Omega \geq \left|\left|\mathbf{\Omega_2 \Omega_1^\dagger}\right|\right|_2, \quad \mathcal{C}_\Psi \geq \left|\left|\mathbf{\Psi_2 \Psi_1^\dagger}\right|\right|_2. \tag{8}$$

Singular Value Bound. We start with a bound on the individual singular values of the reconstructed matrix.

Theorem 3. *Let $\tau_j = \sigma_{p+1}/\sigma_j$. Then,*

$$\sigma_j(\widetilde{\mathbf{A}}) \geq \frac{\sigma_j\left(1-\tau_j^3\,\mathcal{C}_\Omega\,\mathcal{C}_\Psi\right)}{\sqrt{1+\tau_j^2\mathcal{C}_\Omega^2}\sqrt{1+\tau_j^2\mathcal{C}_\Psi^2}}, \quad \textit{for all } 1 \leq j \leq k.$$

Error Norm Bounds. Next, we present error bounds in the spectral and Frobenius norms.

Theorem 4.

$$\left|\left|\mathbf{A}-\widetilde{\mathbf{A}}_k\right|\right|_2^2 \leq \sigma_{k+1}^2 + k\left(\mathcal{C}_\Omega+\mathcal{C}_\Psi\right)^2\sigma_{p+1}^2,$$

$$\left|\left|\mathbf{A}-\widetilde{\mathbf{A}}_k\right|\right|_F^2 \leq \left(\sum_{j=k+1}^{\rho}\sigma_j^2\right) + k\left(\mathcal{C}_\Omega+\mathcal{C}_\Psi\right)^2\sigma_{p+1}^2.$$

Discussion

A good CUR decomposition heavily depends on how the sketch matrices are chosen; Theorems 3 and 4 point out the connection between sketch matrices and the quality of the CUR decomposition through quantities $\mathcal{C}_\Omega$ and $\mathcal{C}_\Psi$.

We call Theorems 3 and 4 *spectral gap error bounds* because they exhibit a surprisingly strong connection between the rate at which the singular values of matrix $\mathbf{A}$ might decay and the quality of the CUR decomposition. For the sake of argument assume for the moment that $\mathcal{C}_\Omega = O(1)$ and $\mathcal{C}_\Psi = O(1)$. When singular values of $\mathbf{A}$ decay rapidly, as they often do in many large data matrices, we can expect $\tau_j \ll 1$ for a choice of p that is somewhat larger than k. Theorem 3 suggests that the leading singular values of $\widetilde{\mathbf{A}}$, $\sigma_j(\widetilde{\mathbf{A}})$ for $1 \leq j \leq k$, differ from the corresponding singular values of $\mathbf{A}$ by a negligible relative amount. Due to our *spectral gap error bound* method, this holds even if all of the individual spectral gaps are small. Similarly, since

$$\left(\sum_{j=k+1}^{\rho}\sigma_j^2\right) \geq \sigma_{k+1}^2 \gg \sigma_{p+1}^2$$

when singular values rapidly decay, Theorems 1 and 4 suggest that the approximation error in $\widetilde{\mathbf{A}}$ differs from that in $\mathbf{A}_k$, the best rank-k approximation, by a negligible additional amount in both the Frobenius norm and spectral norm.

In the remainder of this section, we show that the UCS algorithm from [1] and two sampling algorithms are able to bring both $\mathcal{C}_\Omega$ and $\mathcal{C}_\Psi$ under effective control in their magnitude, leading to high quality CUR decompositions. It is important to note that when using the modified Nyström method, the above bounds still hold with $\mathcal{C}_\Psi := \mathcal{C}_\Omega$.

3.3 Bounds of the Deterministic Unweighted Column Selection

We apply Theorems 3 and 4 to bound the singular value errors and the low-rank approximation error in the spectral and Frobenius norms for the matrix constructed by Algorithm 1.

Theorem 5. (Unweighted Column Selection) *Let $\mathbf{\Pi}_r$ and $\mathbf{\Pi}_c$ be constructed with Algorithm 1, then Theorems 3 and 4 hold with*

$$\mathcal{C}_\Omega^{-1} = \frac{\sqrt{c}-\sqrt{p}}{\sqrt{(\sqrt{n-p}+\sqrt{c})^2+(\sqrt{c}-\sqrt{p})^2}},$$

$$\mathcal{C}_\Psi^{-1} = \frac{\sqrt{r}-\sqrt{p}}{\sqrt{(\sqrt{m-p}+\sqrt{r})^2+(\sqrt{r}-\sqrt{p})^2}}.$$

When applying the result above to the Nyström method (i.e. $\mathbf{R} := \mathbf{C}^T$), one simply needs to ignore the discussion of sampling rows. Simple algebra reveals as c and r increase, $\mathcal{C}_\Omega$ and $\mathcal{C}_\Psi$ will decrease as well. This suggests a tradeoff between controlling the $\mathcal{C}$ terms and improving the spectral gap τ_{k+1}.

3.4 Stochastic Bounds of Sampling Based Algorithms

We apply Theorems 3 and 4 to bound errors in the random sampling methods. μ_r and μ_c in Theorem 6 refer to the row coherence in equation (2) and column coherence in equation (3). The failure probabilities below are squared for the CUR because the rows and columns are sampled independently. When applying the two theorems below to the Nyström method (i.e. $\mathbf{R} := \mathbf{C}^T$), one needs to ignore the discussion of sampling rows and to take the square root of the failure probability by the point above.

Theorem 6. (Uniform Sampling) [8]. *Let $\mathbf{\Pi}_r \in \mathbb{R}^{r\times m}, \mathbf{\Pi}_c \in \mathbb{R}^{n\times c}$ be sketching matrices corresponding to sampling rows and columns uniformly at ran-*

dom, respectively. Fix a failure probability $0 < \delta \ll 1$ *and an accuracy factor* $\epsilon \in (0,1)$. *If*

$$r \geq 2\epsilon^{-2}\mu_r p \ln(p/\delta), \quad c \geq 2\epsilon^{-2}\mu_c p \ln(p/\delta),$$

then Theorems 3 and 4 hold with

$$\mathcal{C}_\Omega = \sqrt{\frac{n}{(1-\epsilon)c}}, \quad \mathcal{C}_\Psi = \sqrt{\frac{m}{(1-\epsilon)r}}$$

with probability at least $(1-\delta)^2$.

Theorem 7. (Leverage Score Sampling) [7] *Let* $\mathbf{\Pi}_r \in \mathbb{R}^{r\times m}, \mathbf{\Pi}_c \in \mathbb{R}^{n\times c}$ *be generated with probability distributions based on the row leverage scores* $\{l_{rj}\}$ *in equation (2) and column leverage scores* $\{l_{cj}\}$ *in equation (3):*

$$p_{rj} = \frac{l_{rj}}{p} \quad \textit{and} \quad p_{cj} = \frac{l_{cj}}{p}$$

for an accuracy factor $\epsilon \in (0,1)$. *If*

$$r \geq 400\epsilon^{-2} p \ln(p), \quad c \geq 400\epsilon^{-2} p \ln(p),$$

then Theorems 3 and 4 hold with

$$\mathcal{C}_\Omega = \sqrt{\frac{1}{1-\epsilon}}, \quad \mathcal{C}_\Psi = \sqrt{\frac{1}{1-\epsilon}}$$

with probability at least $0.9^2 = 0.81$.

4 Numerical Results

In this section, we provide a summary of our empirical evaluation. We start in Section 4.1 with a description of our data sets and our evaluation metrics; then, in Section 4.2, we show how oversampling affects reconstruction error for deterministic and randomized CUR on two data sets with different spectrum properties; and then, in Section 4.3, we compare our *StableCUR* algorithm using input matrices determined by the deterministic UCS algorithm with other related CUR decompositions.

4.1 Data Sets

We use data sets in [9], which include matrices constructed from the bag-of-words data (Dexter) and a Gaussian Radial Basis Function (RBF) Kernel (Abalone). The description of data sets is presented in Table 1. Here, m and n are numbers of columns and rows of the data matrix, %nnz is the percentage of number of non-zero entries, k is the target rank, $p = 2k$, and μ_c and μ_r are the coherence of the rows and columns of $\mathbf{A}$ respectively. Recall that, for a set of data points $\mathbf{x_1}, \ldots, \mathbf{x_n} \in \mathbb{R}^d$, the Gaussian RBF Kernel matrix $\mathbf{A}^\sigma$ is given by $\mathbf{A}^\sigma_{ij} = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\sigma^2}\right)$.

These data matrices are chosen because of their different spectral decay properties. In particular, by adjusting the σ in the Gaussian RBF Kernel, we can change the speed of the decay in a controlled manner. Observe that $\frac{\sigma_p(\mathbf{A})}{\sigma_k(\mathbf{A})}$ (which is the quantity that enters our *spectral gap error bounds*) increases from 0.156 to 0.801 as σ is decreased from 5 to 0.1. In more detail, Figure 3 shows that for the Abalone kernel matrix, decreasing values of σ from 5 to 0.1 slows down the singular value decay, reducing the domination by the top-k eigenspace. Table 1 shows that when $\sigma = 0.1$, the best rank-20 approximation is far from the original matrix (and thus low-rank approximation cannot be expected to yield good results), while for $\sigma = 5$, the matrix is very well approximated by a rank-20 matrix.

In our empirical evaluation, we consider the following measures to compare different CUR algorithms:

- $\sigma_k(\mathbf{CUR})/\sigma_k(\mathbf{A})$, k^{th} singular value ratio
- $\|\mathbf{A} - \mathbf{CUR}\|_F / \|\mathbf{A} - \mathbf{A}_k\|_F$, Frobenius norm error
- $\|\mathbf{A} - (\mathbf{CUR})_k\|_F / \|\mathbf{A} - \mathbf{A}_k\|_F$, rank-$k$ truncated Frobenius norm error
- $\|\mathbf{A} - \mathbf{CUR}\|_2 / \|\mathbf{A} - \mathbf{A}_k\|_2$, spectrum norm error
- $\|\mathbf{A} - (\mathbf{CUR})_k\|_2 / \|\mathbf{A} - \mathbf{A}_k\|_2$, rank-$k$ truncated spectrum norm error.

In addition, the legends in the following plots correspond to the four CUR algorithms we consider:

- RandLeverage: CUR Decomposition of [7] constructed from Leverage Score Sampling
- RandUniform: CUR Decomposition constructed from Uniform Sampling
- NearOptimal: CUR via the Near-Optimal Column Selection Algorithm of [3, 16]
- StableCUR: Our *StableCUR* Algorithm.

4.2 Oversampling Experiments

Here, we test how the spectrum gap can affect the performance of StableCUR and RandLeverage. Our main results are presented in Figures 4 and 5. We choose target rank $k = 20$, $c = r = 80$ and vary p from k to $2k$. Recall from our deterministic structural results from Section 3 that increasing p will decrease $\sigma_{p+1}(\mathbf{A})$ and thus improve the approximation accuracy. However, in Theorems 5 and 7, we showed that increasing p may increase $\mathcal{C}_\Omega$ and $\mathcal{C}_\Psi$. By our bounds, for matrices whose singular values decay rapidly, an increase in p could be beneficial.

Table 1: Dataset Summary.

Data Set	m	n	%nnz	k	$\frac{\|\mathbf{A}\|_F^2}{\|\mathbf{A}\|_2^2}$	$100\ \frac{\|\mathbf{A}-\mathbf{A_k}\|_F}{\|\mathbf{A}\|_F}$	μ_c	μ_r	$\frac{\sigma_p(\mathbf{A})}{\sigma_k(\mathbf{A})}$
Abalone($\sigma = 5$)	4177	4177	100	20	1.09	0.17	10.6	10.6	0.156
Abalone($\sigma = 2$)	4177	4177	100	20	1.88	4.39	2.67	2.67	0.285
Abalone($\sigma = 0.2$)	4177	4177	84.6	20	14.9	79.6	17.6	17.6	0.62
Abalone($\sigma = 0.1$)	4177	4177	40.74	20	174.7	97.47	59.9	59.9	0.801
Dexter	2000	20000	0.48	10	7.16	88.6	197.2	1945	0.806

Figure 4 shows the effects of different values of p on the Frobenius norm reconstruction error. For the Abalone kernel matrix with $\sigma = 5$, both StableCUR and RandLeverage behave better as p increases. On the other hand, when $\sigma = 0.1$, the reconstruction error is much larger and there is little performance gain as p increases.

Figure 5 shows the effects of different values of p on the spectral norm reconstruction error. These plots are qualitatively similar to the Frobenius norm error: for the Abalone kernel matrix with $\sigma = 5$, increasing p reduces reconstruction error for both algorithms, while, when $\sigma = 0.1$, increasing p would not improve StableCUR. Note that for RandLeverage, an increase in p could even *decrease* the reconstruction accuracy. The reason is likely that we do not have effective control on $\mathcal{C}_\Omega$ and $\mathcal{C}_\Psi$ as we increase p.

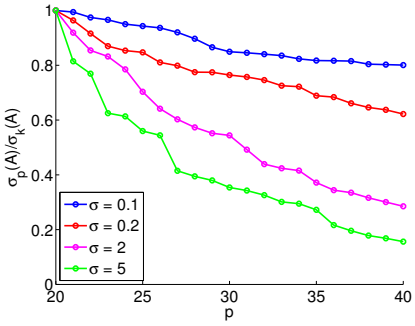

Figure 3: Singular value decay of the Abalone kernel matrices with different σ's. The reported value is the ratio between σ_p and σ_k, where $k = 20$ and p varies from 20 to 40.

4.3 Comparing Different CUR Methods

For completeness, we now compare the performance of different CUR algorithms (RandLeverage, RandUniform, NearOptimal, and StableCUR) on data matrices with the same number of columns and rows. To take advantage of the spectrum gap, we choose oversampling parameter $p = k + 10$ for a matrix (Figure 6) with rapid singular value decay. For matrices (Figures 7 and 8) with slow singular value decay, where there is little benefit to oversample, we choose $p = k$.

Figure 6 shows the performance of the algorithms on the Abalone matrix with $\sigma = 5$, whose singular values decay rapidly. Since $\mathbf{A}_k$ contains most of the information, low rank approximation is a reasonable model. The reconstructed matrix is able to capture most singular values, and the residual errors in both spectral and Frobenius norm decrease rapidly as more columns and rows are sampled. Since the leverage scores are fairly uniform, i.e., the coherence is fairly small, RandUniform performs well in this case, although it is still worse than other algorithms.

Figure 7 shows the performance of different algorithms on the Abalone matrix with $\sigma = 0.1$, whose singular values decay slowly. Since $\mathbf{A}_k$ only contains a small portion of information of $\mathbf{A}$, the curves are flatter in this case. Since the coherence is large, RandUniform performs poorly and RandLeverage performs best under most metrics. However, sampling with more columns and rows only increases approximation accuracy marginally, because the leverage score distribution is extremely imbalanced due to the high coherence of the matrix.

Figure 8 shows performance of different algorithms on the Dexter data matrix. This matrix is "worse" than the Abalone kernel matrix with $\sigma = 5$ because of its slow singular value decay and large coherence, and our empirical results are consistent with this.

5 Conclusion

We have introduced novel *spectral gap error bounds* and demonstrated both theoretically and empirically that only a constant amount of oversampling relative to the target rank is needed for matrices with rapidly decaying singular values. We presented a stable CUR algorithm, *StableCUR*, and combined it with the UCS algorithm to create a stable, efficient, and competitive CUR algorithm that takes advantage of these error bounds.

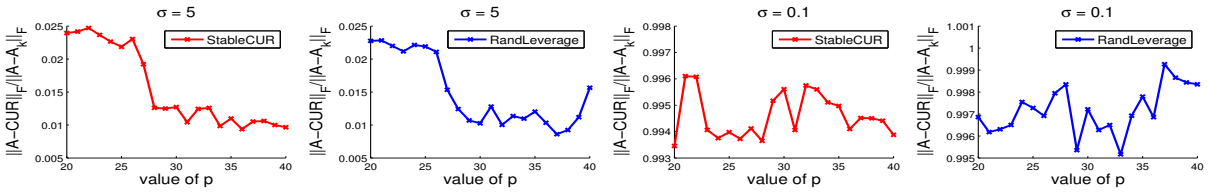

Figure 4: Reconstruction error in Frobenius norm for StableCUR and RandLeverage running on Abalone kernel matrix with $\sigma = 5$ and 0.1. When $\sigma = 5$, both algorithms perform better as we increase p. When $\sigma = 0.1$, the reconstruction errors are less consistent.

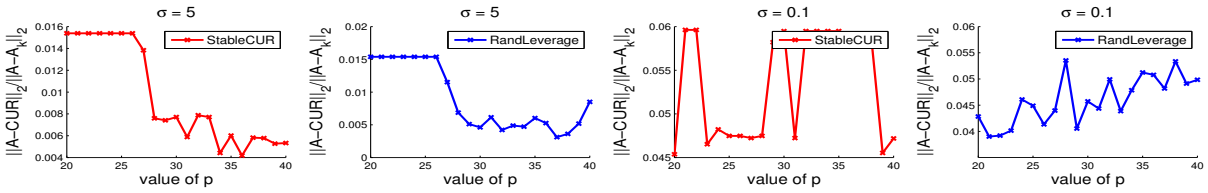

Figure 5: Reconstruction error in spectral norm for StableCUR and RandLeverage running on Abalone kernel matrix with $\sigma = 5$ and 0.1. The results are very similar to figure 5.

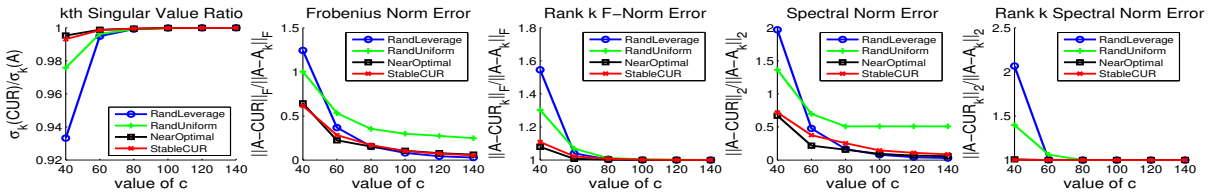

Figure 6: Results of algorithms comparison on RBF kernel($\sigma = 5$) of the Abalone data set. In this matrix, singular values decay very fast, which results in rapid decrease in residual errors and rapid increase in singular value ratio for all algorithms.

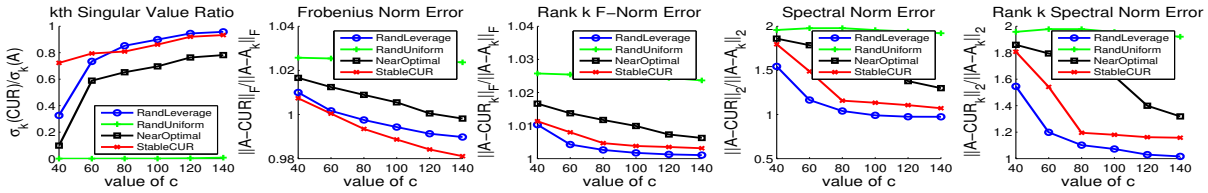

Figure 7: Results of algorithms comparison on RBF kernel($\sigma = 0.1$) of the Abalone data set. In this matrix, singular values decay very slowly. All curves are flatter than the ones in Figure 6.

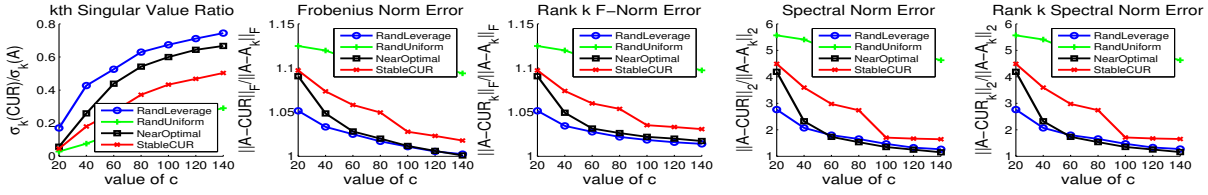

Figure 8: Results of algorithms comparison on Dexter data matrix. This is a non-symmetric matrix with slow decay in its singular values. The performance of algorithms are similar to the ones in Figure 7.

References

[1] D. G. Anderson, M. Gu, and C. Melgaard. An efficient algorithm for unweighted spectral graph sparsification. *arXiv preprint arXiv:1410.4273*, 2014.

[2] J. Bien, Y. Xu, and M. W. Mahoney. CUR from a sparse optimization viewpoint. In *Advances in Neural Information Processing Systems*, pages 217–225, 2010.

[3] C. Boutsidis, P. Drineas, and M. Magdon-Ismail. Near-optimal column-based matrix reconstruction. *SIAM Journal on Computing*, 43(2):687–717, 2014.

[4] C. Boutsidis and D. P. Woodruff. Optimal CUR matrix decompositions. *arXiv preprint arXiv:1405.7910*, 2014.

[5] P. Drineas, M. Magdon-Ismail, M. W. Mahoney, and D. P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(1):3475–3506, 2012.

[6] P. Drineas and M. W. Mahoney. On the Nyström method for approximating a gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6:2153–2175, 2005.

[7] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, 2008.

[8] A. Gittens. The spectral norm error of the naive Nyström extension. *arXiv preprint arXiv:1110.5305*, 2011.

[9] A. Gittens and M. W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. *arXiv preprint arXiv:1303.1849*, 2013.

[10] M. Gu. Subspace iteration randomization and singular value problems. *arXiv preprint arXiv:1408.2208*, 2014.

[11] V. Guruswami and A. K. Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1207–1214. SIAM, 2012.

[12] M. W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224, 2011.

[13] M. W. Mahoney and P. Drineas. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.

[14] M. W. Mahoney, M. Maggioni, and P. Drineas. Tensor-CUR decompositions for tensor-based data. *SIAM Journal on Matrix Analysis and Applications*, 30(3):957–987, 2008.

[15] C. Thurau, K. Kersting, and C. Bauckhage. Deterministic CUR for improved large-scale data analysis: An empirical study. In *SDM*, pages 684–695. SIAM, 2012.

[16] S. Wang and Z. Zhang. Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling. *Journal of Machine Learning Research*, 14(1):2729–2769, 2013.

[17] S. Wang and Z. Zhang. Efficient algorithms and error analysis for the modified Nyström method. *arXiv preprint arXiv:1404.0138*, 2014.

[18] C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, pages 682–688, 2001.

[19] K. Zhang and J. T. Kwok. Clustered Nyström method for large scale manifold learning and dimension reduction. *Neural Networks, IEEE Transactions on*, 21(10):1576–1587, 2010.

[20] K. Zhang, L. Lan, Z. Wang, and F. Moerchen. Scaling up kernel svm on limited resources: A low-rank linearization approach. In *International Conference on Artificial Intelligence and Statistics*, pages 1425–1434, 2012.