Journal of Machine Learning Research 17 (2016) 1-49 Submitted 5/15; Revised 4/16; Published 12/16

Towards More Efficient SPSD Matrix Approximation and CUR Matrix Decomposition

Shusen Wang SHUSEN@BERKELEY.EDU
Department of Statistics
University of California at Berkeley
Berkeley, CA 94720, USA

Zhihua Zhang ZHZHANG@MATH.PKU.EDU.CN
School of Mathematical Sciences
Peking University
Beijing 100871, China

Tong Zhang TZHANG@STAT.RUTGERS.EDU
Department of Statistics
Rutgers University
Piscataway, New Jersey 08854, USA

Editor: Gert Lanckriet

Abstract

Symmetric positive semi-definite (SPSD) matrix approximation methods have been extensively used to speed up large-scale eigenvalue computation and kernel learning methods. The standard sketch based method, which we call the prototype model, produces relatively accurate approximations, but is inefficient on large square matrices. The Nyström method is highly efficient, but can only achieve low accuracy. In this paper we propose a novel model that we call the *fast SPSD matrix approximation model*. The fast model is nearly as efficient as the Nyström method and as accurate as the prototype model. We show that the fast model can potentially solve eigenvalue problems and kernel learning problems in linear time with respect to the matrix size n to achieve $1 + \epsilon$ relative-error, whereas both the prototype model and the Nyström method cost at least quadratic time to attain comparable error bound. Empirical comparisons among the prototype model, the Nyström method, and our fast model demonstrate the superiority of the fast model. We also contribute new understandings of the Nyström method. The Nyström method is a special instance of our fast model and is approximation to the prototype model. Our technique can be straightforwardly applied to make the CUR matrix decomposition more efficiently computed without much affecting the accuracy.

Keywords: Kernel approximation, matrix factorization, the Nyström method, CUR matrix decomposition

1. Introduction

With limited computational and storage resource, machine-precision inversion and decompositions of large and dense matrix are prohibitive. In the past decade matrix approximation techniques have been extensively studied by the theoretical computer science community

©2016 Shusen Wang, Zhihua Zhang, and Tong Zhang.

(Woodruff, 2014), the machine learning community (Mahoney, 2011), and the numerical linear algebra community (Halko et al., 2011).

In machine learning, many graph analysis techniques and kernel methods require expensive matrix computations on symmetric matrices. The truncated eigenvalue decomposition (that is to find a few eigenvectors corresponding to the greatest eigenvalues) is widely used in graph analysis such as spectral clustering, link prediction in social networks (Shin et al., 2012), graph matching (Patro and Kingsford, 2012), etc. Kernel methods (Schölkopf and Smola, 2002) such as kernel PCA and manifold learning require the truncated eigenvalue decomposition. Some other kernel methods such as Gaussian process regression/classification require solving $n \times n$ matrix inversion, where n is the number of training samples. The rank k ($k \ll n$) truncated eigenvalue decomposition (k-eigenvalue decomposition for short) of an $n \times n$ matrix costs time $\tilde{\mathcal{O}}(n^2 k)$[1]; the matrix inversion costs time $\mathcal{O}(n^3)$. Thus, the standard matrix computation approaches are infeasible when n is large.

For kernel methods, we are typically given n data samples of dimension d, while the $n \times n$ kernel matrix $\mathbf{K}$ is unknown beforehand and should be computed. This adds to the additional $\mathcal{O}(n^2 d)$ time cost. When n and d are both large, computing the kernel matrix is prohibitively expensive. Thus, a good kernel approximation method should avoid the computation of the entire kernel matrix.

Typical SPSD matrix approximation methods speed up matrix computation by efficiently forming a low-rank decomposition $\mathbf{K} \approx \mathbf{C}\mathbf{U}\mathbf{C}^T$ where $\mathbf{C} \in \mathbb{R}^{n \times c}$ is a sketch of $\mathbf{K}$ (e.g., randomly sampled c columns of $\mathbf{K}$) and $\mathbf{U} \in \mathbb{R}^{c \times c}$ can be computed in different ways. With such a low-rank approximation at hand, it takes only $\mathcal{O}(nc^2)$ additional time to approximately compute the rank k ($k \leq c$) eigenvalue decomposition or the matrix inversion. Therefore, if $\mathbf{C}$ and $\mathbf{U}$ are obtained in linear time (w.r.t. n) and c is independent of n, then the aforementioned eigenvalue decomposition and matrix inversion can be approximately solved in linear time.

The Nyström method is perhaps the most widely used kernel approximation method. Let $\mathbf{P}$ be an $n \times c$ sketching matrix such as uniform sampling (Williams and Seeger, 2001; Gittens, 2011), adaptive sampling (Kumar et al., 2012), leverage score sampling (Gittens and Mahoney, 2016), etc. The Nyström method computes $\mathbf{C}$ by $\mathbf{C} = \mathbf{K}\mathbf{P} \in \mathbb{R}^{n \times c}$ and $\mathbf{U}$ by $\mathbf{U} = (\mathbf{P}^T \mathbf{C})^\dagger \in \mathbb{R}^{c \times c}$. This way of computing $\mathbf{U}$ is very efficient, but it incurs relatively large approximation error even if $\mathbf{C}$ is a good sketch of $\mathbf{K}$. As a result, the Nyström method is reported to have low approximation accuracy in real-world applications (Dai et al., 2014; Hsieh et al., 2014; Si et al., 2014b). In fact, the Nyström is impossible to attain $1+\epsilon$ bound relative to $\|\mathbf{K} - \mathbf{K}_k\|_F^2$ unless $c \geq \Omega\big(\sqrt{nk/\epsilon}\big)$ (Wang and Zhang, 2013). Here $\mathbf{K}_k$ denotes the best rank-k approximation of $\mathbf{K}$. The requirement that c grows at least linearly with $\sqrt{n}$ is a very pessimistic result. It implies that in order to attain $1+\epsilon$ relative-error bound, the time cost of the Nyström method is of order $nc^2 = \Omega(n^2 k/\epsilon)$ for solving the k-eigenvalue decomposition or matrix inversion, which is quadratic in n. Therefore, under the $1+\epsilon$ relative-error requirement, the Nyström method is not a linear time method.

The main reason for the low accuracy of the Nyström method is due to the way that the $\mathbf{U}$ matrix is calculated. In fact, much higher accuracy can be obtained if $\mathbf{U}$ is calculated

[1] The $\tilde{\mathcal{O}}$ notation hides the logarithm factors.

by solving the minimization problem $\min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2$, which is a standard way to approximate symmetric matrices (Halko et al., 2011; Gittens and Mahoney, 2016; Wang and Zhang, 2013; Wang et al., 2016). This is the randomized SVD for symmetric matrices (Halko et al., 2011). Wang et al. (2016) called this approach the prototype model and provided an algorithm that samples $c = \mathcal{O}(k/\epsilon)$ columns of $\mathbf{K}$ to form $\mathbf{C}$ such that $\min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2 \le (1+\epsilon)\|\mathbf{K} - \mathbf{K}_k\|_F^2$. Unlike the Nyström method, the prototype model does not require c to grow with n. The downside of the prototype model is the high computational cost. It requires the full observation of $\mathbf{K}$ and $\mathcal{O}(n^2c)$ time to compute $\mathbf{U}$. Therefore when applied to kernel approximation, the time cost cannot be less than $\mathcal{O}(n^2d + n^2c)$. To reduce the computational cost, this paper considers the problem of efficient calculation of $\mathbf{U}$ with fixed $\mathbf{C}$ while achieving an accuracy comparable to the prototype model.

More specifically, the key question we try to answer in this paper can be described as follows.

Question 1 *For any fixed $n \times n$ symmetric matrix $\mathbf{K}$, target rank k, and parameter γ, assume that*

A1 We are given a sketch matrix $\mathbf{C} \in \mathbb{R}^{n\times c}$ of $\mathbf{K}$, which is obtained in time Time$(\mathbf{C})$;

A2 The matrix $\mathbf{C}$ is a good sketch of $\mathbf{K}$ in that $\min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2 \le (1+\gamma)\|\mathbf{K} - \mathbf{K}_k\|_F^2$.

Then we would like to know whether for an arbitrary ϵ, it is possible to compute $\mathbf{C}$ and $\tilde{\mathbf{U}}$ such that the following two requirements are satisfied:

R1 The matrix $\tilde{\mathbf{U}}$ has the following error bound:

$$\|\mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T\|_F^2 \le (1+\epsilon)(1+\gamma)\|\mathbf{K} - \mathbf{K}_k\|_F^2.$$

R2 The procedure of computing $\mathbf{C}$ and $\tilde{\mathbf{U}}$ and approximately solving the aforementioned k-eigenvalue decomposition or the matrix inversion run in time $\mathcal{O}\big(n \cdot \mathrm{poly}(k, \gamma^{-1}, \epsilon^{-1})\big) +$ Time$(\mathbf{C})$.

Unfortunately, the following theorem shows that neither the Nyström method nor the prototype model enjoys such desirable properties. We prove the theorem in Appendix B.

Theorem 1 *Neither the Nyström method nor the prototype model satisfies the two requirements in Question 1. To make requirement R1 hold, both the Nyström method and the prototype model cost time no less than $\mathcal{O}\big(n^2 \cdot \mathrm{poly}(k, \gamma^{-1}, \epsilon^{-1})\big) +$ Time$(\mathbf{C})$ which is at least quadratic in n.*

In this paper we give an affirmative answer to the above question. In particular, it has the following consequences. First, the overall approximation has high accuracy in the sense that $\|\mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T\|_F^2$ is comparable to $\min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2$, and is thereby comparable to the best rank k approximation. Second, with $\mathbf{C}$ at hand, the matrix $\tilde{\mathbf{U}}$ is obtained efficiently (linear in n). Third, with $\mathbf{C}$ and $\tilde{\mathbf{U}}$ at hand, it takes extra time which is also linear in n to compute the aforementioned eigenvalue decomposition or linear system. Therefore, with a good $\mathbf{C}$, we can use linear time to obtain desired $\mathbf{U}$ matrix such that the accuracy is comparable to the best possible low-rank approximation.

The CUR matrix decomposition (Mahoney and Drineas, 2009) is closely related to the prototype model and troubled by the same computational problem. The CUR matrix decomposition is an extension of the prototype model from symmetric matrices to general matrices. Given any $m \times n$ fixed matrix $\mathbf{A}$, the CUR matrix decomposition selects c columns of $\mathbf{A}$ to form $\mathbf{C} \in \mathbb{R}^{m\times c}$ and r rows of $\mathbf{A}$ to form $\mathbf{R} \in \mathbb{R}^{r \times n}$, and computes matrix $\mathbf{U} \in \mathbb{R}^{c\times r}$ such that $\|\mathbf{A} - \mathbf{CUR}\|_F^2$ is small. Traditionally, it costs time

$$\mathcal{O}(mn \cdot \min\{c, r\})$$

to compute the optimal $\mathbf{U}^\star = \mathbf{C}^\dagger \mathbf{A} \mathbf{R}^\dagger$ (Stewart, 1999; Wang and Zhang, 2013; Boutsidis and Woodruff, 2014). How to efficiently compute a high-quality $\mathbf{U}$ matrix for CUR is unsolved.

1.1 Main Results

This work is motivated by an intrinsic connection between the Nyström method and the prototype model. Based on a generalization of this observation, we propose the *fast SPSD matrix approximation model* for approximating any symmetric matrix. We show that the fast model satisfies the requirements in Question 1. Given n data points of dimension d, the fast model computes $\mathbf{C}$ and $\mathbf{U}^{\text{fast}}$ and approximately solves the truncated eigenvalue decomposition or matrix inversion in time

$$\mathcal{O}\big(nc^3/\epsilon + nc^2 d/\epsilon\big) + \text{Time}(\mathbf{C}).$$

Here $\text{Time}(\mathbf{C})$ is defined in Question 1.

The fast SPSD matrix approximation model achieves the desired properties in Question 1 by solving $\min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F$ approximately rather than exactly while ensuring

$$\|\mathbf{K} - \mathbf{C}\mathbf{U}^{\text{fast}}\mathbf{C}^T\|_F^2 \le (1+\epsilon) \min_{\mathbf{U}} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2.$$

The time complexity for computing $\mathbf{U}^{\text{fast}}$ is linear in n, which is far less than the time complexity $\mathcal{O}(n^2 c)$ of the prototype model. Our method also avoids computing the entire kernel matrix $\mathbf{K}$; instead, it computes a block of $\mathbf{K}$ of size $\frac{\sqrt{n}c}{\epsilon} \times \frac{\sqrt{n}c}{\epsilon}$, which is substantially smaller than $n \times n$. The lower bound in Theorem 7 indicates that the $\sqrt{n}$ factor here is optimal, but the dependence on c and ϵ are suboptimal and can be potentially improved.

This paper provides a new perspective on the Nyström method. We show that, as well as our fast model, the Nyström method is approximate solution to the problem $\min_{\mathbf{U}} \|\mathbf{C}\mathbf{U}\mathbf{C}^T - \mathbf{K}\|_F^2$. Unfortunately, the approximation is so rough that the quality of the Nyström method is low.

Our method can also be applied to improve the CUR matrix decomposition of the general matrices which are not necessarily square. Given any matrices $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{C} \in \mathbb{R}^{m \times c}$, and $\mathbf{R} \in \mathbb{R}^{r\times n}$, it costs time $\mathcal{O}(mn \cdot \min\{c, r\})$ to compute the matrix $\mathbf{U} = \mathbf{C}^\dagger \mathbf{A} \mathbf{R}^\dagger$. Applying our technique, the time cost drops to only

$$\mathcal{O}\big(cr\epsilon^{-1} \cdot \min\{m, n\} \cdot \min\{c, r\}\big),$$

while the approximation quality is nearly the same.

1.2 Paper Organization

The remainder of this paper is organized as follows. Section 2 defines the notation used in this paper. Section 3 introduces the related work of matrix sketching and SPSD matrix

Table 1: A summary of the notation.

Notation	Description
n	number of data points
d	dimension of the data point
$\mathbf{K}$	$n \times n$ kernel matrix
$\mathbf{P}$, $\mathbf{S}$	sketching matrices
$\mathbf{C}$	$n \times c$ sketch computed by $\mathbf{C} = \mathbf{K}\mathbf{P}$
$\mathbf{U}^{\star}$	$\mathbf{C}^{\dagger}\mathbf{K}(\mathbf{C}^{\dagger})^T \in \mathbb{R}^{c \times c}$—the $\mathbf{U}$ matrix of the prototype model
$\mathbf{U}^{\text{nys}}$	$(\mathbf{P}^T\mathbf{K})^{\dagger} \in \mathbb{R}^{c \times c}$—the $\mathbf{U}$ matrix of the Nyström method
$\mathbf{U}^{\text{fast}}$	$(\mathbf{S}^T\mathbf{C})^{\dagger}(\mathbf{S}^T\mathbf{K}\mathbf{S})(\mathbf{C}^T\mathbf{S})^{\dagger} \in \mathbb{R}^{c \times c}$—the $\mathbf{U}$ matrix of the fast model

approximation. Section 4 describes our fast model and analyze the time complexity and error bound. Section 5 applies the technique of the fast model to compute the CUR matrix decomposition more efficiently. Section 6 conducts empirical comparisons to show the effect of the $\mathbf{U}$ matrix. The proofs of the theorems are in the appendix.

2. Notation

The notation used in this paper are defined as follows. Let $[n] = \{1, \ldots, n\}$, $\mathbf{I}_n$ be the $n \times n$ identity matrix, and $\mathbf{1}_n$ be the $n \times 1$ vector of all ones. We let $x \in y \pm z$ denote $y - z \leq x \leq y + z$. For an $m \times n$ matrix $\mathbf{A} = [A_{ij}]$, we let $\mathbf{a}_{i:}$ be its i-th row, $\mathbf{a}_{:j}$ be its j-th column, $\text{nnz}(\mathbf{A})$ be the number of nonzero entries of $\mathbf{A}$, $\|\mathbf{A}\|_F = (\sum_{i,j} A_{ij}^2)^{1/2}$ be its Frobenius norm, and $\|\mathbf{A}\|_2 = \max_{\mathbf{x} \neq \mathbf{0}} \|\mathbf{A}\mathbf{x}\|_2 / \|\mathbf{x}\|_2$ be its spectral norm.

Let $\rho = \text{rank}(\mathbf{A})$. The condensed singular value decomposition (SVD) of $\mathbf{A}$ is defined as

$$\mathbf{A} \; = \; \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \; = \; \sum_{i=1}^{\rho} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

where $\sigma_1, \cdots, \sigma_r$ are the positive singular values in the descending order. We also use $\sigma_i(\mathbf{A})$ to denote the i-th largest singular value of $\mathbf{A}$. Unless otherwise specified, in this paper "SVD" means the condensed SVD. Let $\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ be the top k principal components of $\mathbf{A}$ for any positive integer k less than ρ. In fact, $\mathbf{A}_k$ is the closest to $\mathbf{A}$ among all the rank k matrices. Let $\mathbf{A}^{\dagger} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$ be the *Moore-Penrose inverse* of $\mathbf{A}$.

Assume that $\rho = \text{rank}(\mathbf{A}) < n$. The column leverage scores of $\mathbf{A}$ are $l_i = \|\mathbf{v}_{i:}\|_2^2$ for $i = 1$ to n. Obviously, $l_1 + \cdots + l_n = \rho$. The column coherence is defined by $\nu(\mathbf{A}) = \frac{n}{\rho} \max_{j \in [n]} \|\mathbf{v}_{j:}\|_2^2$. If $\rho = \text{rank}(\mathbf{A}) < m$, the row leverage scores and coherence are similarly defined. The row leverage scores are $\|\mathbf{u}_{1:}\|_2^2, \cdots, \|\mathbf{u}_{m:}\|_2^2$ and the row coherence is $\mu(\mathbf{A}) = \frac{m}{\rho} \max_{i \in [m]} \|\mathbf{u}_{i:}\|_2^2$.

We also list some frequently used notation in Table 1. Given the decomposition $\tilde{\mathbf{K}} = \mathbf{C}\mathbf{U}\mathbf{C}^T \approx \mathbf{K}$ which has rank at most c, it takes $\mathcal{O}(nc^2)$ time to compute the eigenvalue decomposition of $\tilde{\mathbf{K}}$ and $\mathcal{O}(nc^2)$ time to solve the linear system $(\tilde{\mathbf{K}} + \alpha \mathbf{I}_n)\mathbf{w} = \mathbf{y}$ to obtain $\mathbf{w}$ (see Appendix A for more discussions). The truncated eigenvalue decomposition and linear system are the bottleneck of many kernel methods, and thus an accurate and efficient low-rank approximation can help to accelerate the computation of kernel learning.

3. Related Work

In Section 3.1 we introduce matrix sketching. In Section 3.2 we describe two SPSD matrix approximation methods.

3.1 Matrix Sketching

Popular matrix sketching methods include uniform sampling, leverage score sampling (Drineas et al., 2006, 2008; Woodruff, 2014), Gaussian projection (Johnson and Lindenstrauss, 1984), subsampled randomized Hadamard transform (SRHT) (Drineas et al., 2011; Lu et al., 2013; Tropp, 2011), count sketch (Charikar et al., 2004; Clarkson and Woodruff, 2013; Meng and Mahoney, 2013; Nelson and Nguyên, 2013; Pham and Pagh, 2013; Thorup and Zhang, 2012; Weinberger et al., 2009), etc.

3.1.1 Column Sampling

Let $p_1, \cdots, p_n \in (0,1)$ with $\sum_{i=1}^n p_i = 1$ be the sampling probabilities. Let each integer in $[n]$ be independently sampled with probabilities $sp_1, \cdots, sp_n$, where $s \in [n]$ is integer. Assume that $\tilde{s}$ integers are sampled from $[n]$. Let $i_1, \cdots, i_{\tilde{s}}$ denote the selected integers, and let $\mathbb{E}[\tilde{s}] = s$. We scale each selected column by $\frac{1}{\sqrt{sp_{i_1}}}, \cdots, \frac{1}{\sqrt{sp_{i_{\tilde{s}}}}}$, respectively. Uniform sampling means that the sampling probabilities are $p_1 = \cdots = p_n = \frac{1}{n}$. Leverage score sampling means that the sampling probabilities are proportional to the leverage scores $l_1, \cdots, l_n$ of a certain matrix.

We can equivalently characterize column selection by the matrix $\mathbf{S} \in \mathbb{R}^{n \times \tilde{s}}$. Each column of $\mathbf{S}$ has exactly one nonzero entry; let (i_j, j) be the position of the nonzero entry in the j-th column for $j \in [\tilde{s}]$. For $j = 1$ to $\tilde{s}$, we set

$$S_{i_j, j} = \frac{1}{\sqrt{sp_{i_j}}}. \qquad (1)$$

The expectation $\mathbb{E}[\tilde{s}]$ equals to s, and $\tilde{s} = \Theta(s)$ with high probability. For the sake of simplicity and clarity, in the rest of this paper we will not distinguish $\tilde{s}$ and s.

3.1.2 Random Projection

Let $\mathbf{G} \in \mathbb{R}^{n \times s}$ be a standard Gaussian matrix, namely each entry is sampled independently from $\mathcal{N}(0,1)$. The matrix $\mathbf{S} = \frac{1}{\sqrt{s}}\mathbf{G}$ is a Gaussian projection matrix. Gaussian projection is also well known as the Johnson-Lindenstrauss (JL) transform (Johnson and Lindenstrauss, 1984); its theoretical property is well established. It takes $\mathcal{O}(mns)$ time to apply $\mathbf{S} \in \mathbb{R}^{n \times s}$ to any $m \times n$ dense matrix, which makes Gaussian projection inefficient.

The subsampled randomized Hadamard transform (SRHT) is usually a more efficient alternative of Gaussian projection. Let $\mathbf{H}_n \in \mathbb{R}^{n \times n}$ be the Walsh-Hadamard matrix with $+1$ and -1 entries, $\mathbf{D} \in \mathbb{R}^{n \times n}$ be a diagonal matrix with diagonal entries sampled uniformly from $\{+1, -1\}$, and $\mathbf{P} \in \mathbb{R}^{n \times s}$ be the uniform sampling matrix defined above. The matrix $\mathbf{S} = \frac{1}{\sqrt{n}}\mathbf{D}\mathbf{H}_n\mathbf{P} \in \mathbb{R}^{n \times s}$ is an SRHT matrix, and it can be applied to any $m \times n$ matrix in $\mathcal{O}(mn \log s)$ time.

Count sketch stems from the data stream literature (Charikar et al., 2004; Thorup and Zhang, 2012) and has been applied to speedup matrix computation. The count sketch matrix $\mathbf{S} \in \mathbb{R}^{n \times s}$ can be applied to any matrix $\mathbf{A}$ in $\mathcal{O}(\mathrm{nnz}(\mathbf{A}))$ time where nnz denotes the number of non-zero entries. The readers can refer to (Woodruff, 2014) for detailed descriptions of count sketch.

3.1.3 Theories

The following lemma shows important properties of the matrix sketching methods. In the lemma, leverage score sampling means that the sampling probabilities are proportional to the row leverage scores of the column orthogonal matrix $\mathbf{U} \in \mathbb{R}^{n \times k}$. (Here $\mathbf{U}$ is different from the notation elsewhere in the paper.) We prove the lemma in Appendix C.

Lemma 2 *Let $\mathbf{U} \in \mathbb{R}^{n \times k}$ be any fixed matrix with orthonormal columns and $\mathbf{B} \in \mathbb{R}^{n \times d}$ be any fixed matrix. Let $\mathbf{S} \in \mathbb{R}^{n \times s}$ be any sketching matrix considered in this section; the order of s (with the $\mathcal{O}$-notation omitted) is listed in Table 2. Then*

$$\begin{aligned}
\mathbb{P}\Big\{ \big\| \mathbf{U}^T \mathbf{S} \mathbf{S}^T \mathbf{U} - \mathbf{I}_k \big\|_2 \geq \eta \Big\} \; &\leq \; \delta_1 \qquad \textit{(Property 1)}, \\
\mathbb{P}\Big\{ \big\| \mathbf{U}^T \mathbf{B} - \mathbf{U}^T \mathbf{S} \mathbf{S}^T \mathbf{B} \big\|_F^2 \geq \epsilon \| \mathbf{B} \|_F^2 \Big\} \; &\leq \; \delta_2 \qquad \textit{(Property 2)}, \\
\mathbb{P}\Big\{ \big\| \mathbf{U}^T \mathbf{B} - \mathbf{U}^T \mathbf{S} \mathbf{S}^T \mathbf{B} \big\|_2^2 \geq \epsilon' \| \mathbf{B} \|_2^2 + \frac{\epsilon'}{k} \| \mathbf{B} \|_F^2 \Big\} \; &\leq \; \delta_3 \qquad \textit{(Property 3)}.
\end{aligned}$$

Table 2: The leverage score sampling is w.r.t. the row leverage scores of $\mathbf{U}$. For uniform sampling, the notation $\mu(\mathbf{U}) \in [1, n]$ is the row coherence of $\mathbf{U}$.

Sketching	Property 1	Property 2	Property 3
Leverage Sampling	$\frac{k}{\eta^2} \log \frac{k}{\delta_1}$	$\frac{k}{\epsilon \delta_2}$	—
Uniform Sampling	$\frac{\mu(\mathbf{U}) k}{\eta^2} \log \frac{k}{\delta_1}$	$\frac{\mu(\mathbf{U}) k}{\epsilon \delta_2}$	—
Gaussian Projection	$\frac{k + \log(1/\delta_1)}{\eta^2}$	$\frac{k}{\epsilon \delta_2}$	$\frac{1}{\epsilon'} \big(k + \log \frac{d}{k \delta_3} \big)$
SRHT	$\frac{k + \log n}{\eta^2} \log \frac{k}{\delta_1}$	$\frac{k + \log n}{\epsilon \delta_2}$	$\frac{1}{\epsilon'} \big(k + \log \frac{nd}{k \delta_1} \big) \log \frac{d}{\delta_3}$
Count Sketch	$\frac{k^2}{\delta_1 \eta^2}$	$\frac{k}{\epsilon \delta_2}$	—

Property 1 is known as the subspace embedding property (Woodruff, 2014). It shows that all the singular values of $\mathbf{S}^T \mathbf{U}$ are close to one. Properties 2 and 3 show that sketching preserves the multiplication of a row orthogonal matrix and an arbitrary matrix.

For the SPSD/CUR matrix approximation problems, the three properties are all we need to capture the randomness in the sketching methods. Leverage score sampling, uniform sampling, and count sketch do not enjoy Property 3, but it is fine— Frobenius norm (Property 2) will be used as a loose upper bound on the spectral norm (Property 3). Gaussian projection and SRHT satisfy all the three properties; when applied to the SPSD/CUR problems, their error bounds are stronger than the leverage score sampling, uniform sampling, and count sketch.

3.2 SPSD Matrix Approximation Models

We first describe the prototype model and the Nyström method, which are most relevant to this work. We then introduce several other SPSD matrix approximation methods.

3.2.1 Most Relevant Work

Given an $n \times n$ matrix $\mathbf{K}$ and an $n \times c$ sketching matrix $\mathbf{P}$, we let $\mathbf{C} = \mathbf{K}\mathbf{P}$ and $\mathbf{W} = \mathbf{P}^T\mathbf{C} = \mathbf{P}^T\mathbf{K}\mathbf{P}$. *The prototype model* (Wang and Zhang, 2013) is defined by

$$\tilde{\mathbf{K}}_c^{\text{proto}} \triangleq \mathbf{C}\mathbf{U}^{\star}\mathbf{C}^T = \mathbf{C}\mathbf{C}^{\dagger}\mathbf{K}(\mathbf{C}^{\dagger})^T\mathbf{C}^T, \tag{2}$$

and *the Nyström method* is defined by

$$\begin{aligned}\tilde{\mathbf{K}}_c^{\text{nys}} &\triangleq \mathbf{C}\mathbf{U}^{\text{nys}}\mathbf{C}^T = \mathbf{C}\mathbf{W}^{\dagger}\mathbf{C}^T \\ &= \mathbf{C}\big(\mathbf{P}^T\mathbf{C}\big)^{\dagger}\big(\mathbf{P}^T\mathbf{K}\mathbf{P}\big)\big(\mathbf{C}^T\mathbf{P}\big)^{\dagger}\mathbf{C}^T. \end{aligned}\tag{3}$$

The only difference between the two models is their $\mathbf{U}$ matrices, and the difference leads to big difference in their approximation accuracies. Wang and Zhang (2013) provided a lower error bound of the Nyström method, which shows that no algorithm can select less than $\Omega(\sqrt{nk/\epsilon})$ columns of $\mathbf{K}$ to form $\mathbf{C}$ such that

$$\|\mathbf{K} - \mathbf{C}\mathbf{U}^{\text{nys}}\mathbf{C}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{K} - \mathbf{K}_k\|_F^2.$$

In contrast, the prototype model can attain the $1+\epsilon$ relative-error bound with $c = \mathcal{O}(k/\epsilon)$ (Wang et al., 2016), which is optimal up to a constant factor.

While we have mainly discussed the time complexity of kernel approximation in the previous sections, the memory cost is often a more important issue in large scale problems due to the limitation of computer memory. The Nyström method and the prototype model require $\mathcal{O}(nc)$ memory to hold $\mathbf{C}$ and $\mathbf{U}$ to approximately solve the aforementioned eigenvalue decomposition or the linear system.[2] Therefore, we hope to make c as small as possible while achieving a low approximation error. There are two elements: (1) a good sketch $\mathbf{C} = \mathbf{K}\mathbf{P}$, and (2) a high-quality $\mathbf{U}$ matrix. We focus on the latter in this paper.

3.2.2 Less Relevant Work

We note that there are many other kernel approximation approaches in the literature. However, these approaches do not directly address the issue we consider here, so they are complementary to our work. These studies are either less effective or inherently rely on the Nyström method.

The Nyström-like models such as MEKA (Si et al., 2014a) and the ensemble Nyström method (Kumar et al., 2012) are reported to significantly outperform the Nyström method in terms of approximation accuracy, but their key components are still the Nyström method and the component can be replaced by any other methods such as the method studied in this work. The spectral shifting Nyström method (Wang et al., 2014) also outperforms the

[2] The memory costs of the prototype model is $\mathcal{O}(nc+nd)$ rather than $\mathcal{O}(n^2)$. This is because we can hold the $n \times d$ data matrix and the $c \times n$ matrix $\mathbf{C}^{\dagger}$ in memory, compute a small block of $\mathbf{K}$ each time, and then compute $\mathbf{C}^{\dagger}\mathbf{K}$ block by block.

Nyström method in certain situations, but the spectral shifting strategy can be used for any other kernel approximation models beyond the prototype model. We do not compare with these methods in this paper because MEKA, the ensemble Nyström method, and the spectral shifting Nyström method can all be improved if we replace the underlying Nyström method or the prototype model by the new method developed here.

The column-based low-rank approximation model (Kumar et al., 2009) is another SPSD matrix approximation approach different from the Nyström-like methods. Let $\mathbf{P} \in \mathbb{R}^{n\times c}$ be any sketching matrix and $\mathbf{C} = \mathbf{K}\mathbf{P}$. The column-based model approximates $\mathbf{K}$ by $\mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1/2}\mathbf{C}^T = (\mathbf{C}\mathbf{C}^T)^{1/2}$. Equivalently, it approximates $\mathbf{K}^2$ by

$$\mathbf{K}^T\mathbf{K} \;\approx\; \mathbf{C}\mathbf{C}^T \;=\; \mathbf{K}^T\mathbf{P}\mathbf{P}^T\mathbf{K}.$$

From Lemma 2 we can see that it is a typical sketch based approximation to the matrix multiplication. Unfortunately, the approximate matrix multiplication is effective only when $\mathbf{K}$ has much more rows than columns, which is not true for the kernel matrix. The column-based model does not have good error bound and is not empirically as good as the Nyström method (Kumar et al., 2009).

The random feature mapping (Rahimi and Recht, 2007) is a family of kernel approximation methods. Each random feature mapping method is applicable to certain kernel rather than arbitrary SPSD matrix. Furthermore, they are known to be noticeably less effective than the Nyström method (Yang et al., 2012).

4. The Fast SPSD Matrix Approximation Model

In Section 4.1 we present the motivation behind the fast model. In Section 4.2 we provide an alternative perspective on our fast model and the Nyström method by formulating them as approximate solutions to an optimization problem. In Section 4.3 we analyze the error bound of the fast model. Theorem 3 is the main theorem, which shows that in terms of the Frobenius norm approximation, the fast model is almost as good as the prototype model. In Section 4.4 we describe the implementation of the fast model and analyze the time complexity. In Section 4.5 we give some implementation details that help to improve the approximation quality. In Section 4.6 we show that our fast model exactly recovers $\mathbf{K}$ under certain conditions, and we provide a lower error bound of the fast model.

4.1 Motivation

Let $\mathbf{P} \in \mathbb{R}^{n\times c}$ be sketching matrix and $\mathbf{C} = \mathbf{K}\mathbf{P} \in \mathbb{R}^{n\times c}$. The fast SPSD matrix approximation model is defined by

$$\tilde{\mathbf{K}}_{c,s}^{\textrm{fast}} \;\triangleq\; \mathbf{C}\big(\mathbf{S}^T\mathbf{C}\big)^{\dagger}\big(\mathbf{S}^T\mathbf{K}\mathbf{S}\big)\big(\mathbf{C}^T\mathbf{S}\big)^{\dagger}\mathbf{C}^T,$$

where $\mathbf{S}$ is $n \times s$ sketching matrix.

From (2) and (3) we can see that the Nyström method is a special case of the fast model where $\mathbf{S}$ is defined as $\mathbf{P}$ and that the prototype model is a special case where $\mathbf{S}$ is defined as $\mathbf{I}_n$.

The fast model allows us to trade off the accuracy and the computational cost—larger s leads to higher accuracy and higher time cost, and vice versa. Setting s as small as c

Table 3: Summary of the time cost of the models for computing the $\mathbf{U}$ matrices and the number of entries of $\mathbf{K}$ required to be observed in order to compute the $\mathbf{U}$ matrices. As for the fast model, assume that $\mathbf{S}$ is column selection matrix. The notation is defined previously in Table 1.

	Time	#Entries
Nyström	$\mathcal{O}(c^3)$	nc
Prototype	$\mathcal{O}\big(\mathrm{nnz}(\mathbf{K})c + nc^2\big)$	n^2
Fast	$\mathcal{O}(nc^2 + s^2c)$	$nc + (s-c)^2$

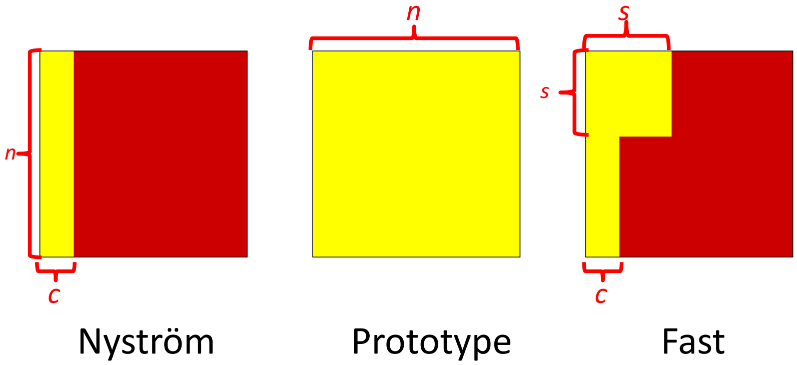


Figure 1: The yellow blocks denote the submatrices of $\mathbf{K}$ that must be seen by the kernel approximation models. The Nyström method computes an $n \times c$ block of $\mathbf{K}$, provided that $\mathbf{P}$ is column selection matrix; the prototype model computes the entire $n \times n$ matrix $\mathbf{K}$; the fast model computes an $n \times c$ block and an $(s-c) \times (s-c)$ block of $\mathbf{K}$ (due to the symmetry of $\mathbf{K}$), provided that $\mathbf{P}$ and $\mathbf{S}$ are column selection matrices.

sacrifices too much accuracy, whereas setting s as large as n is unnecessarily expensive. Later on, we will show that $s = \mathcal{O}(c\sqrt{n/\epsilon}) \ll n$ is a good choice. The setting $s \ll n$ makes the fast model much cheaper to compute than the prototype model. When applied to kernel methods, the fast model avoids computing the entire kernel matrix. We summarize the time complexities of the three matrix approximation methods in Table 3; the middle column lists the time cost for computing the $\mathbf{U}$ matrices given $\mathbf{C}$ and $\mathbf{K}$; the right column lists the number of entry of $\mathbf{K}$ which much be observed. We show a very intuitive comparison in Figure 1.

4.2 Optimization Perspective

With the sketch $\mathbf{C} = \mathbf{KP} \in \mathbb{R}^{n \times c}$ at hand, we want to find the $\mathbf{U}$ matrix such that $\mathbf{CUC}^T \approx \mathbf{K}$. It is very intuitive to solve the following problem to make the approximation tight:

$$\mathbf{U}^{\star} \;=\; \operatorname*{argmin}_{\mathbf{U}} \big\| \mathbf{C}\mathbf{U}\mathbf{C}^T - \mathbf{K} \big\|_F^2 \;=\; \mathbf{C}^{\dagger}\mathbf{K}(\mathbf{C}^{\dagger})^T. \tag{4}$$

This is the prototype model. Since solving this system is time expensive, we propose to draw a sketching matrix $\mathbf{S} \in \mathbb{R}^{n\times s}$ and solve the following problem instead:

$$\begin{aligned} \mathbf{U}^{\text{fast}} &= \operatorname*{argmin}_{\mathbf{U}} \big\| \mathbf{S}^T(\mathbf{C}\mathbf{U}\mathbf{C}^T - \mathbf{K})\mathbf{S} \big\|_F^2 \\ &= \operatorname*{argmin}_{\mathbf{U}} \big\| (\mathbf{S}^T\mathbf{C})\mathbf{U}(\mathbf{S}^T\mathbf{C})^T - \mathbf{S}^T\mathbf{K}\mathbf{S} \big\|_F^2 \\ &= (\mathbf{S}^T\mathbf{C})^{\dagger}(\mathbf{S}^T\mathbf{K}\mathbf{S})(\mathbf{C}^T\mathbf{S})^{\dagger}, \end{aligned} \tag{5}$$

which results in the fast model. Similar ideas have been exploited to efficiently solve the least squares regression problem (Drineas et al., 2006, 2011; Clarkson and Woodruff, 2013), but their analysis can not be directly applied to the more complicated system (5).

This approximate linear system interpretation offers a new perspective on the Nyström method. The $\mathbf{U}$ matrix of the Nyström method is in fact an approximate solution to the problem $\min_{\mathbf{U}} \|\mathbf{C}\mathbf{U}\mathbf{C}^T - \mathbf{K}\|_F^2$. The Nyström method uses $\mathbf{S} = \mathbf{P}$ as the sketching matrix, which leads to the solution

$$\mathbf{U}^{\text{nys}} \;=\; \operatorname*{argmin}_{\mathbf{U}} \big\| \mathbf{P}^T(\mathbf{C}\mathbf{U}\mathbf{C}^T - \mathbf{K})\mathbf{P} \big\|_F^2 \;=\; (\mathbf{P}^T\mathbf{K}\mathbf{P})^{\dagger} \;=\; \mathbf{W}^{\dagger}.$$

4.3 Error Analysis

Let $\mathbf{U}^{\text{fast}}$ correspond to the fast model (5). Any of the five sketching methods in Lemma 2 can be used to compute $\mathbf{U}^{\text{fast}}$, although column selection is more useful than random projection in this application. In the following we show that $\mathbf{U}^{\text{fast}}$ is nearly as good as $\mathbf{U}^{\star}$ in terms of the objective function value. The proof is in Appendix D.

Theorem 3 (Main Result) *Let $\mathbf{K}$ be any $n \times n$ fixed symmetric matrix, $\mathbf{C}$ be any $n \times c$ fixed matrix, $k_c = \operatorname{rank}(\mathbf{C})$, and $\mathbf{U}^{fast}$ be the $c\times c$ matrix defined in (5). Let $\mathbf{S} \in \mathbb{R}^{n\times s}$ be any of the five sketching matrices defined in Table 4. Assume that $\epsilon^{-1} = o(n)$ or $\epsilon^{-1} = o(n/c)$. The inequality*

$$\big\| \mathbf{K} - \mathbf{C}\mathbf{U}^{fast}\mathbf{C}^T \big\|_F^2 \quad\leq\quad (1+\epsilon)\, \min_{\mathbf{U}} \big\| \mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T \big\|_F^2 \tag{6}$$

holds with probability at least 0.8.

In the theorem, Gaussian projection and SRHT require smaller sketch size than the other three methods. It is because Gaussian projection and SRHT enjoys all of Properties 1, 2, 3 in Lemma 2, whereas leverage score sampling, uniform sampling, and count sketch does not enjoy Property 3.

Remark 4 *Wang et al. (2016) showed that there exists an algorithm (though not linear-time algorithm) attaining the error bound*

$$\big\| \mathbf{K} - \mathbf{C}\mathbf{C}^{\dagger}\mathbf{K}(\mathbf{C}^{\dagger})^T\mathbf{C}^T \big\|_F^2 \;\leq\; (1+\epsilon)\big\| \mathbf{K} - \mathbf{K}_k \big\|_F^2$$

Table 4: Leverage score sampling means sampling according to the row leverage scores of $\mathbf{C}$. For uniform sampling, the parameter $\mu(\mathbf{C}) \in [1, n]$ is the row coherence of $\mathbf{C}$.

Sketching	Order of s	Assumption	T_{sketch}	#Entries
Leverage Score Sampling	$c\sqrt{n/\epsilon}$	$\epsilon = o(n)$	$\mathcal{O}(nc^2 + s^2)$	$nc + (s-c)^2$
Uniform Sampling	$\mu(\mathbf{C})c\sqrt{n/\epsilon}$	$\epsilon = o(n)$	$\mathcal{O}(s^2)$	$nc + (s-c)^2$
Gaussian Projection	$\sqrt{\frac{n}{c\epsilon}}\big(c + \log\frac{n}{c}\big)$	$\epsilon = o(n/c)$	$\mathcal{O}\big(\mathrm{nnz}(\mathbf{K})s\big)$	n^2
SRHT	$\sqrt{\frac{n}{c\epsilon}}(c + \log n)\log(n)$	$\epsilon = o(n/c)$	$\mathcal{O}(n^2 \log s)$	n^2
Count Sketch	$c\sqrt{n/\epsilon}$	$\epsilon = o(n)$	$\mathcal{O}\big(\mathrm{nnz}(\mathbf{K})\big)$	n^2

Algorithm 1 The Fast SPSD Matrix Approximation Model.

```
Input: an $n \times n$ symmetric matrix $\mathbf{K}$ and the number of selected columns or target dimension of projection $c$ $(< n)$.
Sketching: $\mathbf{C} = \mathbf{KP}$ using an arbitrary $n \times c$ sketching matrix $\mathbf{P}$ (not studied in this work);
Optional: replace $\mathbf{C}$ by any orthonormal bases of the columns of $\mathbf{C}$;
Compute another $n \times s$ sketching matrix $\mathbf{S}$, e.g. the leverage score sampling in Algorithm 2;
Compute the sketches $\mathbf{S}^T\mathbf{C} \in \mathbb{R}^{s \times c}$ and $\mathbf{S}^T\mathbf{KS} \in \mathbb{R}^{s \times s}$;
Compute $\mathbf{U}^{\text{fast}} = (\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{KS})(\mathbf{C}^T\mathbf{S})^\dagger \in \mathbb{R}^{c \times c}$;
Output: $\mathbf{C}$ and $\mathbf{U}^{\text{fast}}$ such that $\mathbf{K} \approx \mathbf{C}\mathbf{U}^{\text{fast}}\mathbf{C}^T$.
```

with high probability by sampling $c = \mathcal{O}(k/\epsilon)$ columns of $\mathbf{K}$ to form $\mathbf{C}$. Let $\mathbf{C} \in \mathbb{R}^{n \times c}$ be formed by this algorithm and $\mathbf{S} \in \mathbb{R}^{n \times s}$ be the leverage score sampling matrix. With $c = \mathcal{O}(k/\epsilon)$ and $s = \tilde{\mathcal{O}}(n^{1/2}k\epsilon^{-3/2})$, the fast model satisfies

$$\big\|\mathbf{K} - \mathbf{C}\mathbf{U}^{fast}\mathbf{C}^T\big\|_F^2 \;\leq\; (1+\epsilon)\big\|\mathbf{K} - \mathbf{K}_k\big\|_F^2$$

with high probability.

4.4 Algorithm and Time Complexity

We describe the whole procedure of the fast model in Algorithm 1, where $\mathbf{S} \in \mathbb{R}^{n \times s}$ can be one of the five sketching matrices described in Table 4. Given $\mathbf{C}$ and (the whole or a part of) $\mathbf{K}$, it takes time

$$\mathcal{O}\big(s^2 c\big) + T_{\text{sketch}}$$

to compute $\mathbf{U}^{\text{fast}}$, where T_{sketch} is the time cost of forming the sketches $\mathbf{S}^T\mathbf{C}$ and $\mathbf{S}^T\mathbf{KS}$ and is described in Table 4. In Table 4 we also show the number of entries of $\mathbf{K}$ that must be observed. From Table 4 we can see that column selection is much more efficient than random projection, and column selection does not require the full observation of $\mathbf{K}$.

We are particularly interested in the column selection matrix $\mathbf{S}$ corresponding to the row leverage scores of $\mathbf{C}$. The leverage score sampling described in Algorithm 2 can be efficiently performed. Using the leverage score sampling, it takes time $\mathcal{O}(nc^3/\epsilon)$ (excluding the time of computing $\mathbf{C} = \mathbf{KP}$) to compute $\mathbf{U}^{\text{fast}}$. For the kernel approximation problem, suppose that we are given n data points of dimension d and that the kernel matrix $\mathbf{K}$ is unknown beforehand. Then it takes $\mathcal{O}(nc^2 d/\epsilon)$ additional time to evaluate the kernel function values.

Algorithm 2 The Leverage Score Sampling Algorithm.

```
Input: an $n \times c$ matrix $\mathbf{C}$, an integer $s$.
Compute the condensed SVD of $\mathbf{C}$ (by discarding the zero singular values) to obtain the orthonormal bases $\mathbf{U}_\mathbf{C} \in \mathbb{R}^{n \times \rho}$, where $\rho = \mathrm{rank}(\mathbf{C}) \leq c$;
Compute the sampling probabilities $p_i = s\ell_i/\rho$, where $\ell_i = \|\mathbf{e}_i^T \mathbf{U}_\mathbf{C}\|_2^2$ is the $i$-th leverage score;
Initialize $\mathbf{S}$ to be an matrices of size $n \times 0$;
for $i = 1$ to $n$ do
    With probability $p_i$, add $\sqrt{\frac{c}{s\ell_i}}\mathbf{e}_i$ to be a new column of $\mathbf{S}$, where $\mathbf{e}_i$ is the $i$-th standard basis;
end for
Output: $\mathbf{S}$, whose expected number of columns is $s$.
```

4.5 Implementation Details

In practice, the approximation accuracy and numerical stability can be significantly improved by the following techniques and tricks.

If $\mathbf{P}$ and $\mathbf{S}$ are both random sampling matrices, then empirically speaking, enforcing $\mathcal{P} \subset \mathcal{S}$ significantly improves the approximation accuracy. Here $\mathcal{P}$ and $\mathcal{S}$ are the subsets of $[n]$ selected by $\mathbf{P}$ and $\mathbf{S}$, respectively. Instead of directly sampling s indices from $[n]$ by Algorithm 2, it is better to sample s indices from $[n] \setminus \mathcal{P}$ to form $\mathcal{S}'$ and let $\mathcal{S} = \mathcal{S}' \cup \mathcal{P}$. In this way, $s + c$ columns are sampled. Whether the requirement $\mathcal{P} \subset \mathcal{S}$ improves the accuracy is unknown to us.

Corollary 5 *Theorem 3 still holds when we restrict $\mathcal{P} \subset \mathcal{S}$.*

Proof Let $p_1, \cdots, p_n$ be the original sampling probabilities without the restriction $\mathcal{P} \subset \mathcal{S}$. We define the modified sampling probabilities by

$$\tilde{p}_i \; = \; \begin{cases} 1 & \text{if } i \in \mathcal{P}; \\ p_i & \text{otherwise} . \end{cases}$$

The column sampling with restriction $\mathcal{P} \subset \mathcal{S}$ amounts to sampling columns according to $\tilde{p}_1, \cdots, \tilde{p}_n$. Since $\tilde{p}_i \geq p_i$ for all $i \in [n]$, it follows from Remark 14 that the error bound will not get worse if p_i is replaced by $\tilde{p}_i$. ■

If $\mathbf{S}$ is the leverage score sampling matrix, we find it better not to scale the entries of $\mathbf{S}$, although the scaling is necessary for theoretical analysis. According to our observation, the scaling sometimes makes the approximation numerically unstable.

4.6 Additional Properties

When $\mathbf{K}$ is a low-rank matrix, the Nyström method and the prototype model are guaranteed to exactly recover $\mathbf{K}$ (Kumar et al., 2009; Talwalkar and Rostamizadeh, 2010; Wang et al., 2016). We show in the following theorem that the fast model has the same property. We prove the theorem in Appendix E.

Theorem 6 (Exact Recovery) *Let $\mathbf{K}$ be any $n \times n$ symmetric matrix, $\mathbf{P} \in \mathbb{R}^{n \times c}$ and $\mathbf{S} \in \mathbb{R}^{n \times s}$ be any sketching matrices, $\mathbf{C} = \mathbf{K}\mathbf{P}$, and $\mathbf{W} = \mathbf{P}^T\mathbf{C}$. Assume that $\mathrm{rank}(\mathbf{S}^T\mathbf{C}) \geq \mathrm{rank}(\mathbf{W})$. Then $\mathbf{K} = \mathbf{C}(\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{K}\mathbf{S})(\mathbf{C}^T\mathbf{S})^\dagger\mathbf{C}^T$ if and only if $\mathrm{rank}(\mathbf{K}) = \mathrm{rank}(\mathbf{C})$.*

In the following we establish a lower error bound of the fast model, which implies that to attain the $1+\epsilon$ Frobenius norm bound relative to the best rank k approximation, the fast model must satisfy

$$c \geq \Omega\big(k/\epsilon\big) \quad \text{and} \quad s \geq \Omega\big(\sqrt{nk/\epsilon}\big).$$

Notice that the theorem only holds for column selection matrices $\mathbf{P}$ and $\mathbf{S}$. We prove the theorem in Appendix F.

Theorem 7 (Lower Bound) *Let $\mathbf{P} \in \mathbb{R}^{n\times c}$ and $\mathbf{S} \in \mathbb{R}^{n \times s}$ be any two column selection matrices such that $\mathcal{P} \subset \mathcal{S} \subset [n]$, where $\mathcal{P}$ and $\mathcal{S}$ are the index sets formed by $\mathbf{P}$ and $\mathbf{S}$, respectively. There exists an $n\times n$ symmetric matrix $\mathbf{K}$ such that*

$$\frac{\|\mathbf{K} - \tilde{\mathbf{K}}_{c,s}^{fast}\|_F^2}{\|\mathbf{K} - \mathbf{K}_k\|_F^2} \quad \geq \quad \frac{n-c}{n-k}\Big(1+\frac{2k}{c}\Big) + \frac{n-s}{n-k}\frac{k(n-s)}{s^2}, \tag{7}$$

where k is arbitrary positive integer smaller than n, $\mathbf{C} = \mathbf{K}\mathbf{P} \in \mathbb{R}^{n\times c}$, and

$$\tilde{\mathbf{K}}_{c,s}^{fast} \;=\; \mathbf{C}(\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{K}\mathbf{S})(\mathbf{C}^T\mathbf{S})^\dagger\mathbf{C}^T$$

is the fast model.

Interestingly, Theorem 7 matches the lower bounds of the Nyström method and the prototype model. When $s = c$, the right-hand side of (7) becomes $\Omega(1 + kn/c^2)$, which is the lower error bound of the Nyström method given by Wang and Zhang (2013). When $s = n$, the right-hand side of (7) becomes $\Omega(1 + k/c)$, which is the lower error bound of the prototype model given by Wang et al. (2016).

5. Extension to CUR Matrix Decomposition

In Section 5.1 we describe the CUR matrix decomposition and establish an improved error bound of CUR in Theorem 8. In Section 5.2 we use sketching to more efficiently compute the $\mathbf{U}$ matrix of CUR. Theorem 8 and Theorem 9 together show that our fast CUR method satisfies $1 + \epsilon$ error bound relative to the best rank k approximation. In Section 5.3 we provide empirical results to intuitively illustrate the effectiveness of our fast CUR. In Section 5.4 we discuss the application of our results beyond the CUR decomposition.

5.1 The CUR Matrix Decomposition

Given any $m \times n$ matrix $\mathbf{A}$, the CUR matrix decomposition is computed by selecting c columns of $\mathbf{A}$ to form $\mathbf{C} \in \mathbb{R}^{m\times c}$ and r rows of $\mathbf{A}$ to form $\mathbf{R} \in \mathbb{R}^{r\times n}$ and computing the $\mathbf{U}$ matrix such that $\|\mathbf{A} - \mathbf{CUR}\|_F^2$ is small. CUR preserves the sparsity and non-negativity properties of $\mathbf{A}$; it is thus more attractive than SVD in certain applications (Mahoney and Drineas, 2009). In addition, with the CUR of $\mathbf{A}$ at hand, the truncated SVD of $\mathbf{A}$ can be very efficiently computed.

A standard way to finding the $\mathbf{U}$ matrix is by minimizing $\|\mathbf{A} - \mathbf{CUR}\|_F^2$ to obtain the optimal $\mathbf{U}$ matrix

$$\mathbf{U}^\star \;=\; \operatorname*{argmin}_{\mathbf{U}} \|\mathbf{A} - \mathbf{CUR}\|_F^2 \;=\; \mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger, \tag{8}$$

which has been used by Stewart (1999); Wang and Zhang (2013); Boutsidis and Woodruff (2014). This approach costs time $\mathcal{O}(mc^2 + nr^2)$ to compute the Moore-Penrose inverse and $\mathcal{O}(mn \cdot \min\{c, r\})$ to compute the matrix product. Therefore, even if $\mathbf{C}$ and $\mathbf{R}$ are uniformly sampled from $\mathbf{A}$, the time cost of CUR is $\mathcal{O}(mn \cdot \min\{c, r\})$.

At present the strongest theoretical guarantee is by Boutsidis and Woodruff (2014). They use the adaptive sampling algorithm to select $c = \mathcal{O}(k/\epsilon)$ column and $r = \mathcal{O}(k/\epsilon)$ rows to form $\mathbf{C}$ and $\mathbf{R}$, respectively, and form $\mathbf{U}^\star = \mathbf{C}^\dagger \mathbf{A} \mathbf{R}^\dagger$. The approximation error is bounded by

$$\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 \; \leq \; (1+\epsilon)\|\mathbf{A} - \mathbf{A}_k\|_F^2.$$

This result matches the theoretical lower bound up to a constant factor. Therefore this CUR algorithm is near optimal. We establish in Theorem 8 an improved error bound of the adaptive sampling based CUR algorithm, and the constants in the theorem are better than the those in (Boutsidis and Woodruff, 2014). Theorem 8 is obtained by following the idea of Boutsidis and Woodruff (2014) and slightly changing the proof of Wang and Zhang (2013). The proof is in Appendix G.

Theorem 8 *Let $\mathbf{A}$ be any given $m \times n$ matrix, k be any positive integer less than m and n, and $\epsilon \in (0, 1)$ be an arbitrary error parameter. Let $\mathbf{C} \in \mathbb{R}^{m \times c}$ and $\mathbf{R} \in \mathbb{R}^{r \times n}$ be columns and rows of $\mathbf{A}$ selected by the near-optimal column selection algorithm of Boutsidis et al. (2014). When c and r are both greater than $4k\epsilon^{-1}\big(1 + o(1)\big)$, the following inequality holds:*

$$\mathbb{E}\big\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \; \leq \; (1+\epsilon)\|\mathbf{A} - \mathbf{A}_k\|_F^2,$$

where the expectation is taken w.r.t. the random column and row selection.

5.2 Fast CUR Decomposition

Analogous to the fast SPSD matrix approximation model, the CUR decomposition can be sped up while preserving its accuracy. Let $\mathbf{S}_C \in \mathbb{R}^{m \times s_c}$ and $\mathbf{S}_R \in \mathbb{R}^{n \times s_r}$ be any sketching matrices satisfying the approximate matrix multiplication properties. We propose to compute $\mathbf{U}$ more efficiently by

$$\begin{aligned} \tilde{\mathbf{U}} \; &= \; \operatorname*{argmin}_{\mathbf{U}} \|\mathbf{S}_C^T \mathbf{A} \mathbf{S}_R - (\mathbf{S}_C^T \mathbf{C}) \mathbf{U} (\mathbf{R} \mathbf{S}_R)\|_F^2 \\ &= \; \underbrace{(\mathbf{S}_C^T \mathbf{C})^\dagger}_{c \times s_c} \underbrace{(\mathbf{S}_C^T \mathbf{A} \mathbf{S}_R)}_{s_c \times s_r} \underbrace{(\mathbf{R} \mathbf{S}_R)^\dagger}_{s_r \times r}, \end{aligned} \tag{9}$$

which costs time

$$\mathcal{O}(s_r r^2 + s_c c^2 + s_c s_r \cdot \min\{c, r\}) + T_{\mathrm{sketch}},$$

where T_{sketch} denotes the time for forming the sketches $\mathbf{S}_C^T \mathbf{A} \mathbf{S}_R$, $\mathbf{S}_C^T \mathbf{C}$, and $\mathbf{R}\mathbf{S}_R$. As for Gaussian projection, SRHT, and count sketch, T_{sketch} are respectively $\mathcal{O}\big(\mathrm{nnz}(\mathbf{A}) \min\{s_c, s_r\}\big)$, $\mathcal{O}\big(mn \log(\min\{s_c, s_r\})\big)$, and $\mathcal{O}\big(\mathrm{nnz}(\mathbf{A})\big)$. As for leverage score sampling and uniform sampling, T_{sketch} are respectively $\mathcal{O}(mc^2 + nr^2 + s_c s_r)$ and $\mathcal{O}(s_c s_r)$. Forming the sketches by column selection is more efficient than by random projection.

The following theorem shows that when s_c and s_r are sufficiently large, $\tilde{\mathbf{U}}$ is nearly as good as the best possible $\mathbf{U}$ matrix. In the theorem, leverage score sampling means that $\mathbf{S}_C$ and $\mathbf{S}_R$ sample columns according to the row leverage scores of $\mathbf{C}$ and $\mathbf{R}^T$, respectively. The proof is in Appendix H.

Theorem 9 *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{C} \in \mathbb{R}^{m\times c}$, $\mathbf{R} \in \mathbb{R}^{r\times n}$ be any fixed matrices with $c \ll n$ and $r \ll m$. Let $q = \min\{m, n\}$ and $\tilde{q} = \min\{m/c, n/r\}$. The sketching matrices $\mathbf{S}_C \in \mathbb{R}^{m\times s_c}$ and $\mathbf{S}_R \in \mathbb{R}^{n\times s_r}$ are described in Table 5. Assume that $\epsilon^{-1} = o(q)$ or $\epsilon^{-1} = o(\tilde{q})$, as shown in the table. The matrix $\tilde{\mathbf{U}}$ is defined in (9). Then the inequality*

$$\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 \;\leq\; (1+\epsilon) \min_{\mathbf{U}} \|\mathbf{A} - \mathbf{C}\mathbf{U}\mathbf{R}\|_F^2$$

holds with probability at least 0.7.

Table 5: Leverage score sampling means sampling according to the row leverage scores of $\mathbf{C}$ and the column leverage scores of $\mathbf{R}$, respectively. For uniform sampling, the parameter $\mu(\mathbf{C})$ is the row coherence of $\mathbf{C}$ and $\nu(\mathbf{R})$ is the column coherence of $\mathbf{R}$.

Sketching	Order of s_c	Order of s_r	Assumption
Leverage Score Sampling	$c\sqrt{q/\epsilon}$	$r\sqrt{q/\epsilon}$	$\epsilon^{-1} = o(q)$
Uniform Sampling	$\mu(\mathbf{C})c\sqrt{q/\epsilon}$	$\nu(\mathbf{R})r\sqrt{q/\epsilon}$	$\epsilon^{-1} = o(q)$
Gaussian Projection	$\sqrt{\frac{m}{c\epsilon}}\big(c + \log\frac{n}{c}\big)$	$\sqrt{\frac{n}{r\epsilon}}\big(r + \log\frac{m}{r}\big)$	$\epsilon^{-1} = o(\tilde{q})$
SRHT	$\sqrt{\frac{m}{c\epsilon}}\big(c + \log\frac{mn}{c}\big)\log(m)$	$\sqrt{\frac{n}{r\epsilon}}\big(r + \log\frac{mn}{r}\big)\log(n)$	$\epsilon^{-1} = o(\tilde{q})$
Count Sketch	$c\sqrt{q/\epsilon}$	$r\sqrt{q/\epsilon}$	$\epsilon^{-1} = o(q)$

As for leverage score sampling, uniform sampling, and count sketch, the sketch sizes $s_c = \mathcal{O}(c\sqrt{q/\epsilon})$ and $s_r = \mathcal{O}(r\sqrt{q/\epsilon})$ suffice, where $q = \min\{m, n\}$. As for Gaussian projection and SRHT, much smaller sketch sizes are required: $s_c = \tilde{\mathcal{O}}(\sqrt{mc/\epsilon})$ and $s_r = \tilde{\mathcal{O}}(\sqrt{nr/\epsilon})$ suffice. However, these random projection methods are inefficient choices in this application and only have theoretical interest. Only column sampling methods have linear time complexities. If $\mathbf{S}_C$ and $\mathbf{S}_R$ are leverage score sampling matrices (according to the row leverage scores of $\mathbf{C}$ and $\mathbf{R}^T$, respectively), it follows from Theorem 9 that $\tilde{\mathbf{U}}$ with $1 + \epsilon$ bound can be computed in time

$$\mathcal{O}\big(s_r r^2 + s_c c^2 + s_c s_r \cdot \min\{c, r\}\big) + T_{\text{sketch}} \;=\; \mathcal{O}\big(cr\epsilon^{-1} \cdot \min\{m, n\} \cdot \min\{c, r\}\big),$$

which is linear in $\mathcal{O}(\min\{m, n\})$.

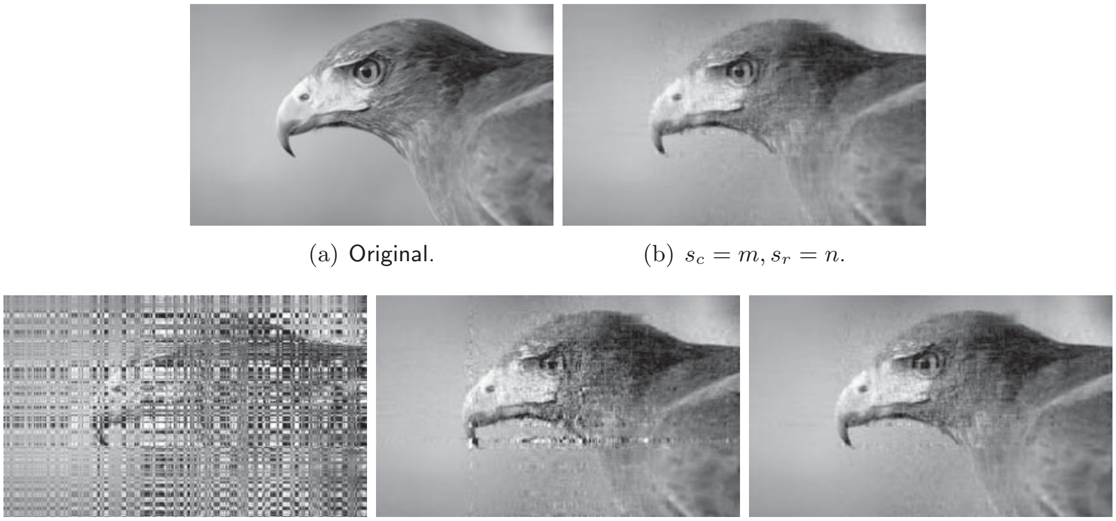
5.3 Empirical Comparisons

To intuitively demonstrate the effectiveness of our method, we conduct a simple experiment on a 1920 × 1168 natural image obtained from the internet. We first uniformly sample $c = 100$ columns to form $\mathbf{C}$ and $r = 100$ rows to form $\mathbf{R}$, and then compute the $\mathbf{U}$ matrix by varying s_c and s_r. We show the image $\tilde{\mathbf{A}} = \mathbf{CUR}$ in Figure 2.

Figure 2(b) is obtained by computing the $\mathbf{U}$ matrix according to (8), which is the best possible result when $\mathbf{C}$ and $\mathbf{R}$ are fixed. The $\mathbf{U}$ matrix of Figure 2(c) is computed according to Drineas et al. (2008):

$$\mathbf{U} \;=\; (\mathbf{P}_R^T \mathbf{A} \mathbf{P}_C)^\dagger,$$

where $\mathbf{P}_C$ and $\mathbf{P}_R$ are column selection matrices such that $\mathbf{C} = \mathbf{A}\mathbf{P}_C$ and $\mathbf{R} = \mathbf{P}_R^T\mathbf{A}$. This is equivalently to (9) by setting $\mathbf{S}_C = \mathbf{P}_R$ and $\mathbf{S}_R = \mathbf{P}_C$. Obviously, this setting leads to very poor quality. In Figures 2(c) and (d) the sketching matrices $\mathbf{S}_C$ and $\mathbf{S}_R$ are uniform sampling matrices. The figures show that when s_c and s_r are moderately greater than r and c, respectively, the approximation quality is significantly improved. Especially, when $s_c = 4r$ and $s_r = 4c$, the approximation quality is nearly as good as using the optimal $\mathbf{U}$ matrix defined in (8).



(a) Original. (b) $s_c = m, s_r = n$.

(c) $s_c = r, s_r = c$. (d) $s_c = 2r, s_r = 2c$. (e) $s_c = 4r, s_r = 4c$.

Figure 2: (a): the original 1920 × 1168 image. (b) to (e): CUR decomposition with $c = r = 100$ and different settings of s_c and s_r.

5.4 Discussions

We note that we are not the first to use row and column sampling to solve the CUR problem more efficiently, though we are the first to provide rigorous error analysis. Previous work has exploited similar ideas as heuristics to speed up computation and to avoid visiting every entry of $\mathbf{A}$. For example, the MEKA method (Si et al., 2014a) partitions the kernel matrix $\mathbf{K}$ into b^2 blocks $\mathbf{K}^{(i,j)}$ ($i = 1, \cdots, b$ and $j = 1, \cdots, b$), and requires solving

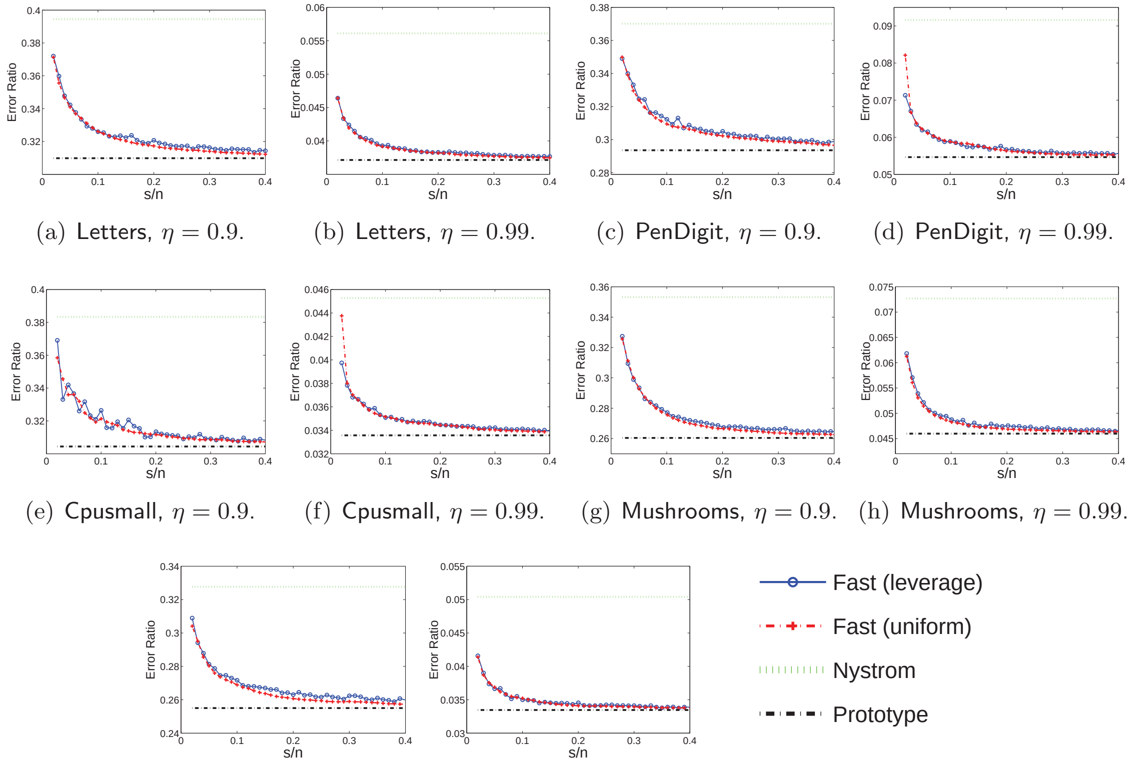
$$\mathbf{L}^{(i,j)} \; = \; \operatorname*{argmin}_{\mathbf{L}} \big\| \mathbf{W}^{(i)} \mathbf{L} \mathbf{W}^{(j)^T} - \mathbf{K}^{(i,j)} \big\|_F^2$$

for all $i \in [b]$, $j \in [b]$, and $i \neq j$. Since $\mathbf{W}^{(i)}$ and $\mathbf{W}^{(j)}$ have much more rows than columns, Si et al. (2014a) proposed to approximately solve the linear system by uniformly sampling rows from $\mathbf{W}^{(i)}$ and $\mathbf{K}^{(i,j)}$ and columns from $(\mathbf{W}^{(j)})^T$ and $\mathbf{K}^{(i,j)}$, and they noted that this heuristic works pretty well. The basic ideas of our fast CUR and their MEKA are the same; their experiments demonstrate the effectiveness and efficiency of this approach, and

Table 6: A summary of the datasets for kernel approximation.

Dataset	Letters	PenDigit	Cpusmall	Mushrooms	WineQuality
#Instance	$15,000$	10,992	$8,192$	$8,124$	$4,898$
#Attribute	16	16	12	112	12
σ (when $\eta = 0.90$)	0.400	0.101	0.075	1.141	0.314
σ (when $\eta = 0.99$)	0.590	0.178	0.180	1.960	0.486

our analysis answers why this approach is correct. This also implies that our algorithms and analysis may have broad applications and impacts beyond the CUR decomposition and SPSD matrix approximation.

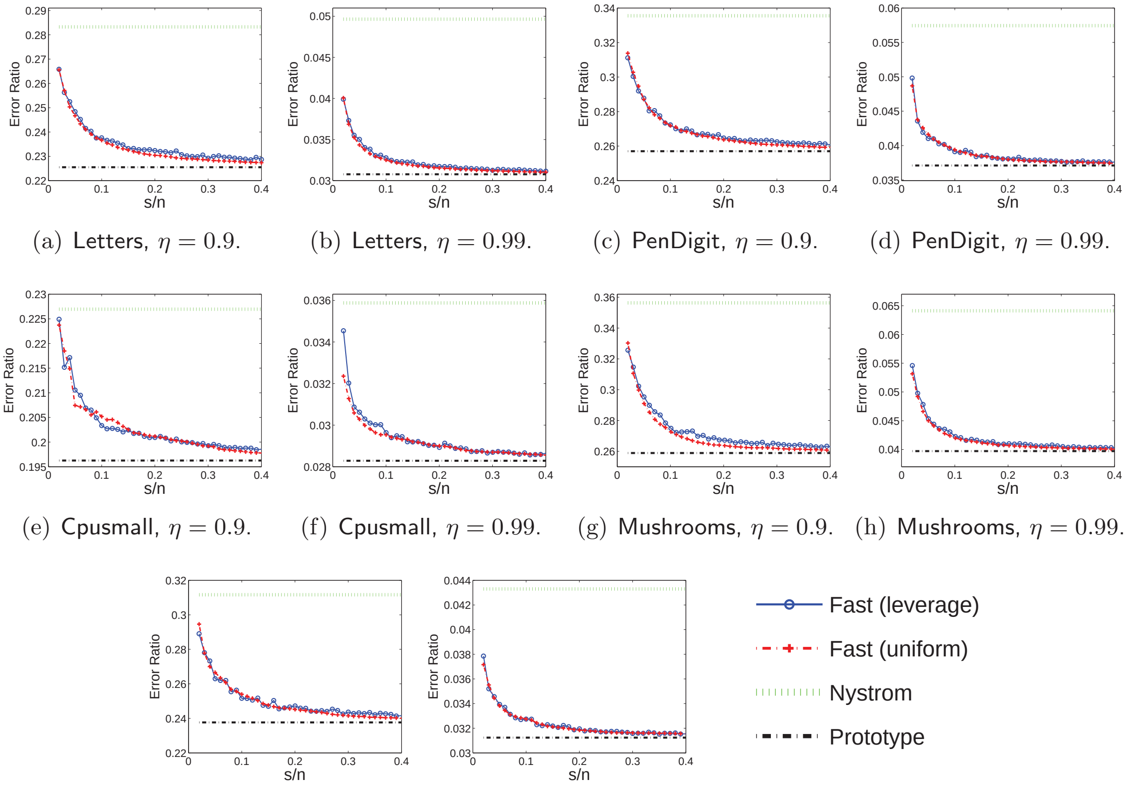


(a) Letters, $\eta = 0.9$. (b) Letters, $\eta = 0.99$. (c) PenDigit, $\eta = 0.9$. (d) PenDigit, $\eta = 0.99$.

(e) Cpusmall, $\eta = 0.9$. (f) Cpusmall, $\eta = 0.99$. (g) Mushrooms, $\eta = 0.9$. (h) Mushrooms, $\eta = 0.99$.

(i) Wine, $\eta = 0.9$. (j) Wine, $\eta = 0.99$. (k) Legend.

Figure 3: The plot of $\frac{s}{n}$ against the approximation error $\|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2 / \|\mathbf{K}\|_F^2$, where $\mathbf{C}$ contains $c = \lceil n/100 \rceil$ column of $\mathbf{K} \in \mathbb{R}^{n \times n}$ selected by uniform sampling.



(a) Letters, $\eta = 0.9$.
(b) Letters, $\eta = 0.99$.
(c) PenDigit, $\eta = 0.9$.
(d) PenDigit, $\eta = 0.99$.
(e) Cpusmall, $\eta = 0.9$.
(f) Cpusmall, $\eta = 0.99$.
(g) Mushrooms, $\eta = 0.9$.
(h) Mushrooms, $\eta = 0.99$.
(i) Wine, $\eta = 0.9$.
(j) Wine, $\eta = 0.99$.
(k) Legend.

Figure 4: The plot of $\frac{s}{n}$ against the approximation error $\|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2 / \|\mathbf{K}\|_F^2$, where $\mathbf{C}$ contains $c = \lceil n/100 \rceil$ column of $\mathbf{K} \in \mathbb{R}^{n \times n}$ selected by the uniform+adaptive^2 sampling algorithm (Wang et al., 2016).

6. Experiments

In this section we conduct several sets of illustrative experiments to show the effect of the $\mathbf{U}$ matrix. We compare the three methods with different settings of c and s. We do not compare with other kernel approximation methods for the reasons stated in Section 3.2.2.

6.1 Setup

Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ be the $d \times n$ data matrix, and $\mathbf{K}$ be the RBF kernel matrix with each entry computed by $K_{ij} = \exp\big(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\big)$ where σ is the scaling parameter.

When comparing the kernel approximation error $\|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2$, we set the scaling parameter σ in the following way. We let $k = \lceil n/100 \rceil$ and define

$$\eta \;=\; \frac{\|\mathbf{K}_k\|_F^2}{\|\mathbf{K}\|_F^2} \;=\; \frac{\sum_{i=1}^k \sigma_i^2(\mathbf{K})}{\sum_{i=1}^n \sigma_i^2(\mathbf{K})},$$

which indicate the importance of the top one percent singular values of $\mathbf{K}$. In general η grows with σ. We set σ such that $\eta = 0.9$ or 0.99.

All the methods are implemented in MATLAB and run on a laptop with Intel i5 2.5GHz CUP and 8GB RAM. To compare the running time, we set MATLAB in the single thread mode.

6.2 Kernel Approximation Accuracy

We conduct experiments on several datasets available at the LIBSVM site. The datasets are summarized in Table 6. In this set of experiments, we study the effect of the $\mathbf{U}$ matrices. We use two methods to form $\mathbf{C} \in \mathbb{R}^{n \times c}$: uniform sampling and the uniform+adaptive^2 sampling (Wang et al., 2016); we fix $c = \lceil n/100 \rceil$. For our fast model, we use two kinds of sketching matrices $\mathbf{S} \in \mathbb{R}^{n \times s}$: uniform sampling and leverage score sampling; we vary s from $2c$ to $40c$. We plot $\frac{s}{n}$ against the approximation error $\|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2 / \|\mathbf{K}\|_F^2$ in Figures 3 and 4. The Nyström method and the prototype model are included for comparison.

Figures 3 and 4 show that the fast SPSD matrix approximation model is significantly better than the Nyström method when s is slightly larger than c, e.g., $s = 2c$. Recall that the prototype model is a special case of the fast model where $s = n$. We can see that the fast model is nearly as accurate as the prototype model when s is far smaller than n, e.g., $s = 0.2n$.

The results also show that using uniform sampling and leverage score sampling to generate $\mathbf{S}$ does not make much difference. Thus, in practice, one can simply compute $\mathbf{S}$ by uniform sampling.

By comparing the results in Figures 3 and 4, we can see that computing $\mathbf{C}$ by uniform+adaptive^2 sampling is substantially better than uniform sampling. However, adaptive sampling requires the full observation of $\mathbf{K}$; thus with uniform+adaptive^2 sampling, our fast model does not have much advantage over the prototype model in terms of time efficiency. Our main focus of this work is the $\mathbf{U}$ matrix, so in the rest of the experiments we simply use uniform sampling to compute $\mathbf{C}$.

6.3 Approximate Kernel Principal Component Analysis

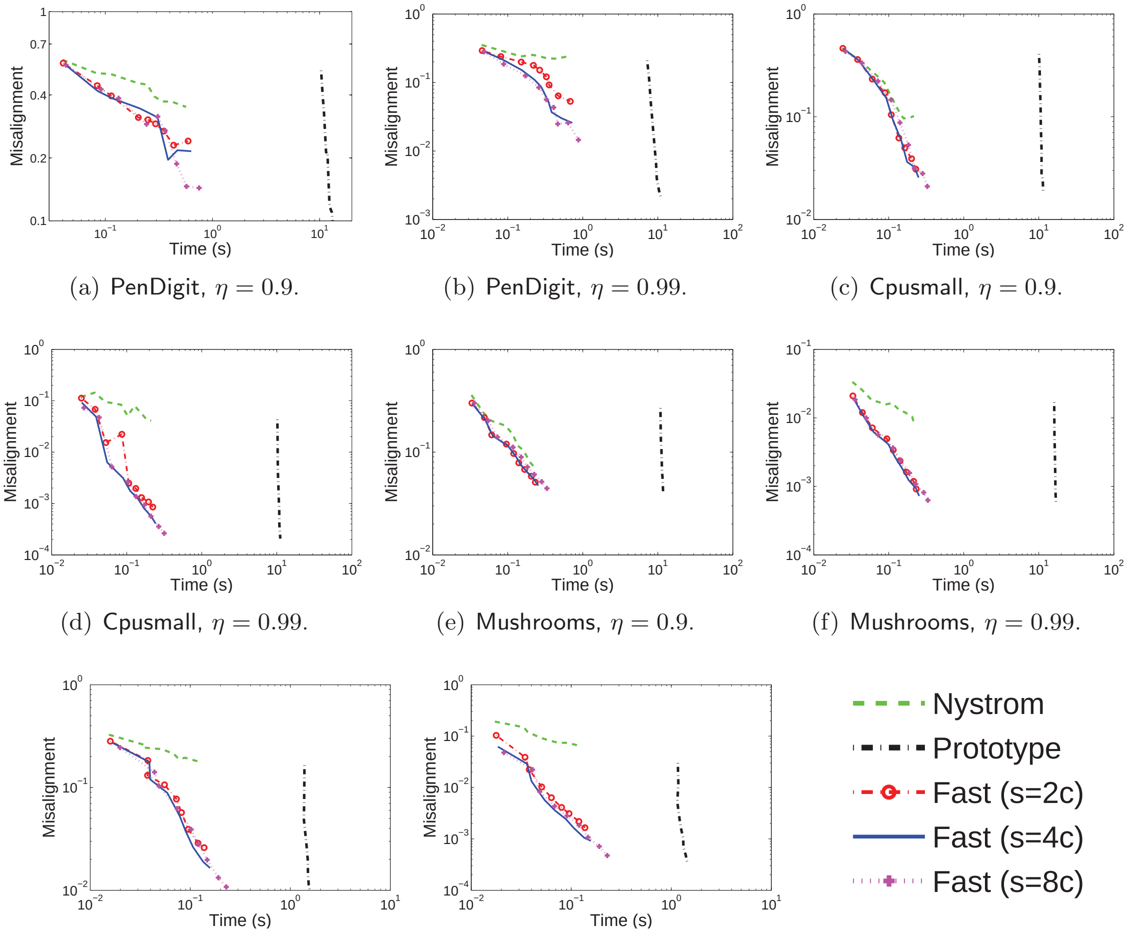
We apply the three methods to approximately compute kernel principal component analysis (KPCA), and contrast with the exact solution. The experiment setting follows Zhang and Kwok (2010). We fix k and vary c. For our fast model, we set $s = 2c$, $4c$, or $8c$. Since computing $\mathbf{S}$ by uniform sampling or leverage score sampling yields the same empirical performance, we use only uniform sampling. Let $\mathbf{C}\mathbf{U}\mathbf{C}^T$ be the low-rank approximation formed by the three methods. Let $\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}}\tilde{\mathbf{V}}^T$ be the k-eigenvalue decomposition of $\mathbf{C}\mathbf{U}\mathbf{C}^T$.

6.3.1 Quality of the Approximate Eigenvectors

Let $\mathbf{U}_{\mathbf{K},k} \in \mathbb{R}^{n \times k}$ contain the top k eigenvectors of $\mathbf{K}$. In the first set of experiments, we measure the distance between $\mathbf{U}_{\mathbf{K},k}$ and the approximate eigenvectors $\tilde{\mathbf{V}}$ by

$$\text{Misalignment} \;=\; \frac{1}{k}\big\|\mathbf{U}_{\mathbf{K},k} - \tilde{\mathbf{V}}\tilde{\mathbf{V}}^T\mathbf{U}_{\mathbf{K},k}\big\|_F^2 \;\; (\in\; [0,1]). \tag{10}$$

Small misalignment indicates high approximation quality. We fix $k = 3$.



(a) PenDigit, $\eta = 0.9$.

(b) PenDigit, $\eta = 0.99$.

(c) Cpusmall, $\eta = 0.9$.

(d) Cpusmall, $\eta = 0.99$.

(e) Mushrooms, $\eta = 0.9$.

(f) Mushrooms, $\eta = 0.99$.

(g) Wine, $\eta = 0.9$.

(h) Wine, $\eta = 0.99$.

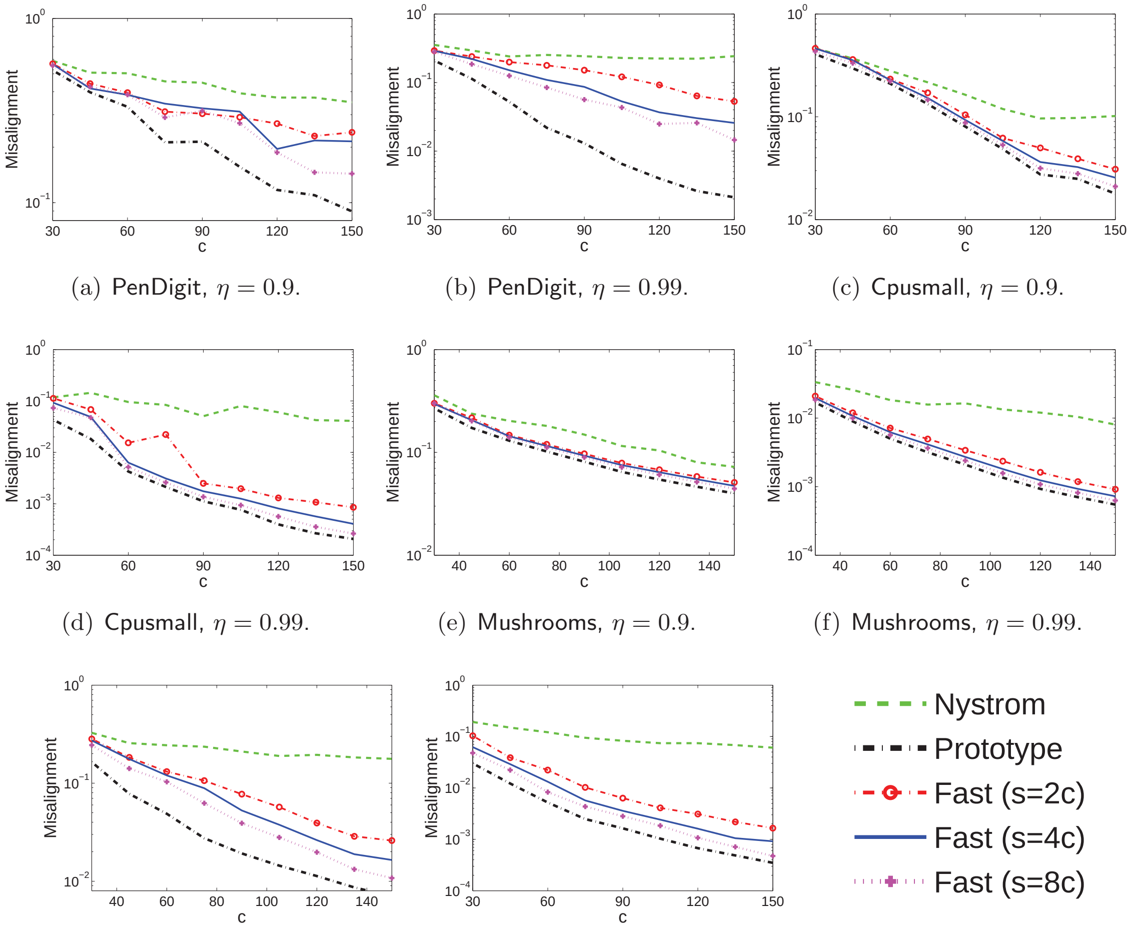
(i) Legend.

Figure 5: The plot of (log-scale) elapsed time against the (log-scale) misalignment defined in (10).

We conduct experiments on the datasets summarized in Table 6. We record the elapsed time of the entire procedure—computing (part of) the kernel matrix, computing $\mathbf{C}$ and $\mathbf{U}$ by the kernel approximation methods, computing the k-eigenvalue decomposition of $\mathbf{CUC}^T$. We plot the elapsed time against the misalignment defined in Figure 5. Results on the Letters dataset are not reported because the exact k-eigenvalue decomposition on MATLAB ran out of memory, making it impossible to calculate the misalignment.

At the end of Section 3.2.1 we have mentioned the importance of memory cost of the kernel approximation methods and that all three compared methods cost $\mathcal{O}(nc + nd)$ memory. Since n and d are fixed, we plot c against the misalignment in Figure 6 to show the memory efficiency.

The results show that using the same amount of time or memory, the misalignment incurred by the Nyström method is usually tens of times higher than our fast model. The



(a) PenDigit, $\eta = 0.9$.

(b) PenDigit, $\eta = 0.99$.

(c) Cpusmall, $\eta = 0.9$.

(d) Cpusmall, $\eta = 0.99$.

(e) Mushrooms, $\eta = 0.9$.

(f) Mushrooms, $\eta = 0.99$.

(g) Wine, $\eta = 0.9$.

(h) Wine, $\eta = 0.99$.

(i) Legend.

Figure 6: The plot of c against the (log-scale) misalignment defined in (10).

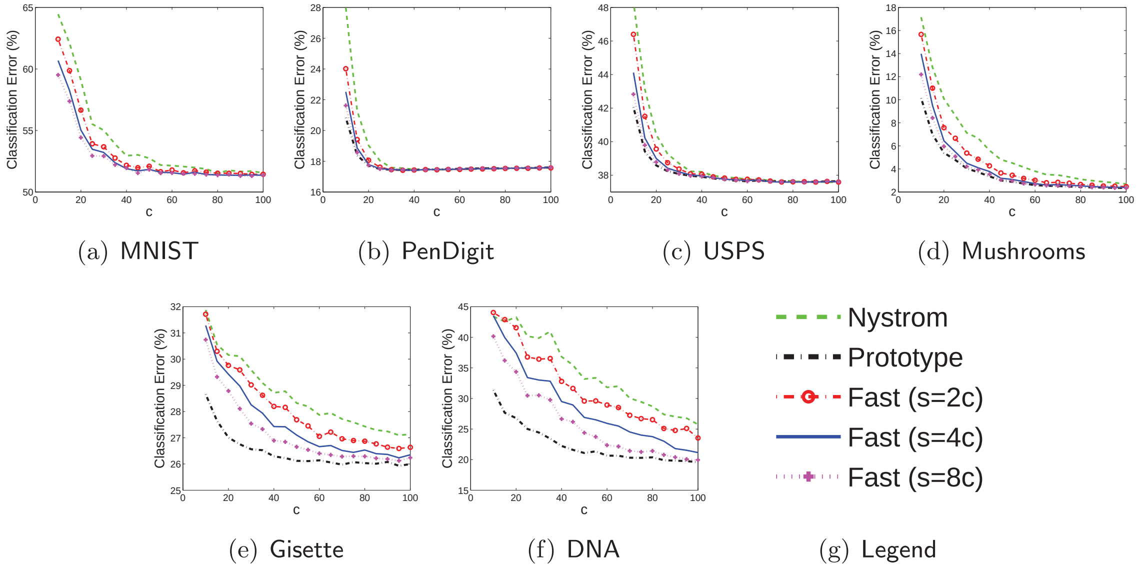
Table 7: A summary of the datasets for clustering and classification.

Dataset	MNIST	Pendigit	USPS	Mushrooms	Gisette	DNA
#Instance	$60,000$	$10,992$	$9,298$	$8,124$	$7,000$	$2,000$
#Attribute	780	16	256	112	$5,000$	180
#Class	10	10	10	2	2	3
Scaling Parameter σ	10	0.7	15	3	50	4

experiment also shows that with fixed c, the fast model is nearly as accuracy as the prototype model when $s = 8c \ll n$.

6.3.2 Quality of the Generalization

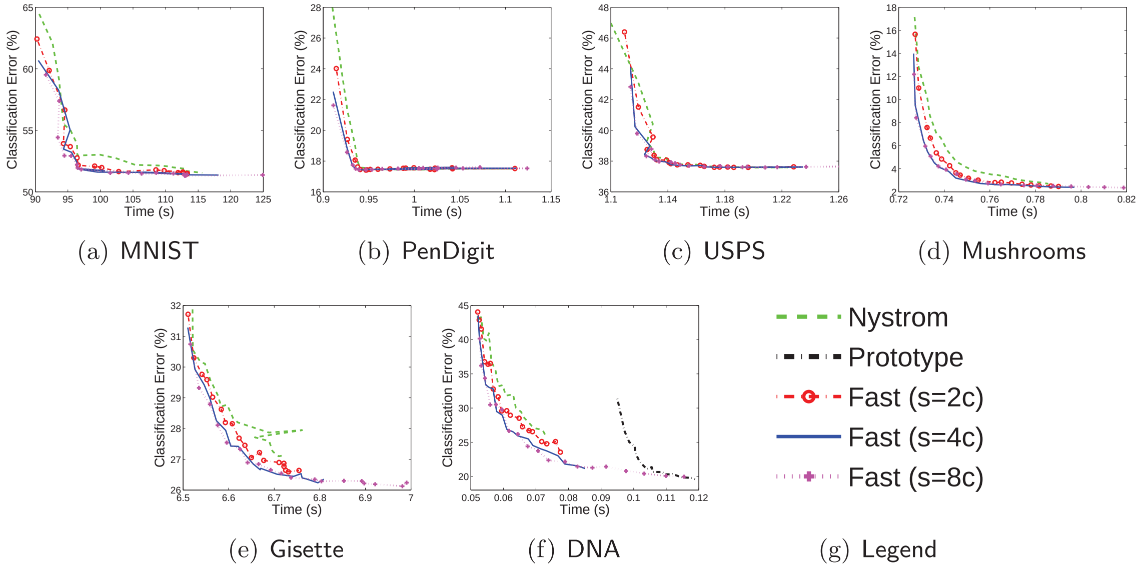
In the second set of experiments, we test the generalization performance of the kernel approximation methods on classification tasks. The classification datasets are described in



(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

(e) Gisette (f) DNA (g) Legend

Figure 7: The plot of c against the classification error. Here $k = 3$.



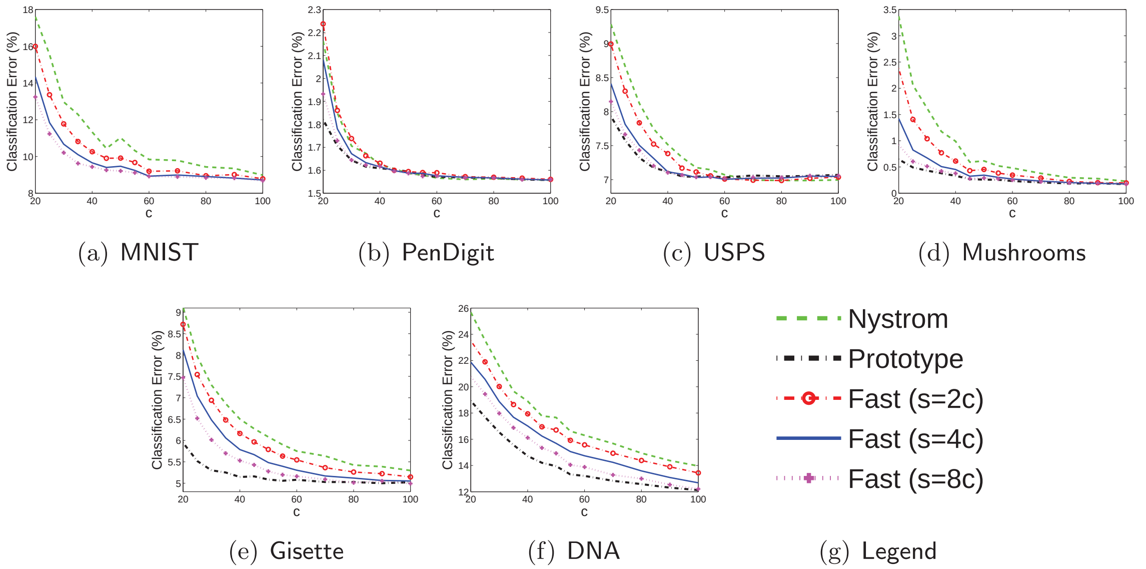
(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

(e) Gisette (f) DNA (g) Legend

Figure 8: The plot of elapsed time against the classification error. Here $k = 3$.

Table 7. For each dataset, we randomly sample $n_1 = 50\%n$ data points for training and the rest $50\%n$ for test. In this set of experiments, we set $k = 3$ and $k = 10$.

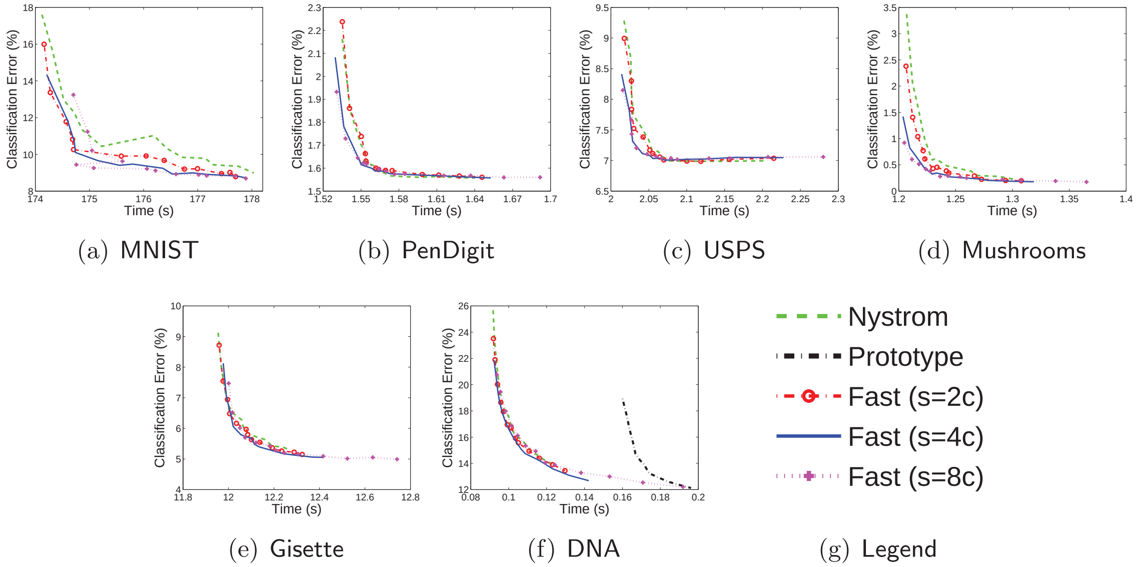
We let $\mathbf{K} \in \mathbb{R}^{n_1 \times n_1}$ be the RBF kernel matrix of the training data and $\mathbf{k}(\mathbf{x}) \in \mathbb{R}^{n_1}$ be defined by $[\mathbf{k}(\mathbf{x})]_i = \exp\big(-\frac{\|\mathbf{x}-\mathbf{x}_i\|_2^2}{2\sigma^2}\big)$, where $\mathbf{x}_i$ is the i-th training data point. In the training step, we approximately compute the top k eigenvalues and eigenvectors, and denote



(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

(e) Gisette (f) DNA (g) Legend

Figure 9: The plot of c against the classification error. Here $k = 10$.



(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

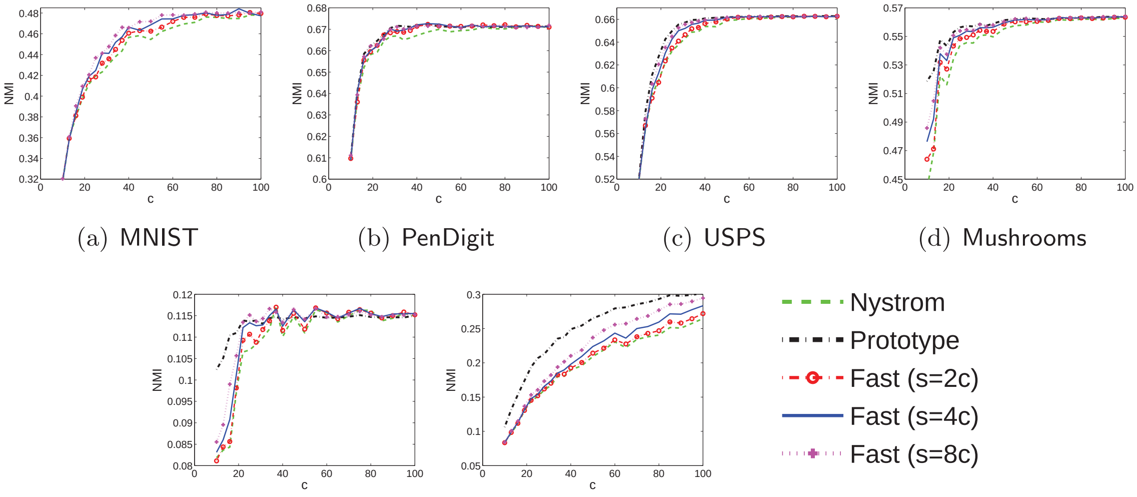
(e) Gisette (f) DNA (g) Legend

Figure 10: The plot of elapsed time against the classification error. Here $k = 10$.

$\tilde{\mathbf{\Lambda}} \in \mathbb{R}^{k \times k}$ and $\tilde{\mathbf{V}} \in \mathbb{R}^{n_1 \times k}$. The feature vector (extracted by KPCA) of the i-th training data point is the i-th column of $\tilde{\mathbf{\Lambda}}^{0.5} \tilde{\mathbf{V}}^T$. In the test step, the feature vector of test data $\mathbf{x}$ is $\tilde{\mathbf{\Lambda}}^{-0.5} \tilde{\mathbf{V}}^T \mathbf{k}(\mathbf{x})$. Then we put the training labels and training and test features into the MATLAB K-nearest-neighbor classifier `knnclassify` to classify the test data. We fix the number of nearest neighbors to be 10. The scaling parameters of each dataset are listed in

Table 7. Since the kernel approximation methods are randomized, we repeat the training and test procedure 20 times and record the average elapsed time and average classification error.

We plot c against the classification error in Figures 7 and 9, and plot the elapsed time (excluding the time cost of KNN) against the classification error in Figures 8 and 10. Using the same amount of memory, the fast model is significantly better than the Nyström method, especially when c is small. Using the same amount of time, the fast model outperforms the Nyström method by one to two percent of classification error in many cases, and it is at least as good as the Nyström method in the rest cases. This set of experiments also indicate that the fast model with $s = 4c$ or $8c$ has the best empirical performance.



(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

(e) Gisette (f) DNA (g) Legend

Figure 11: The plot of c against NMI.

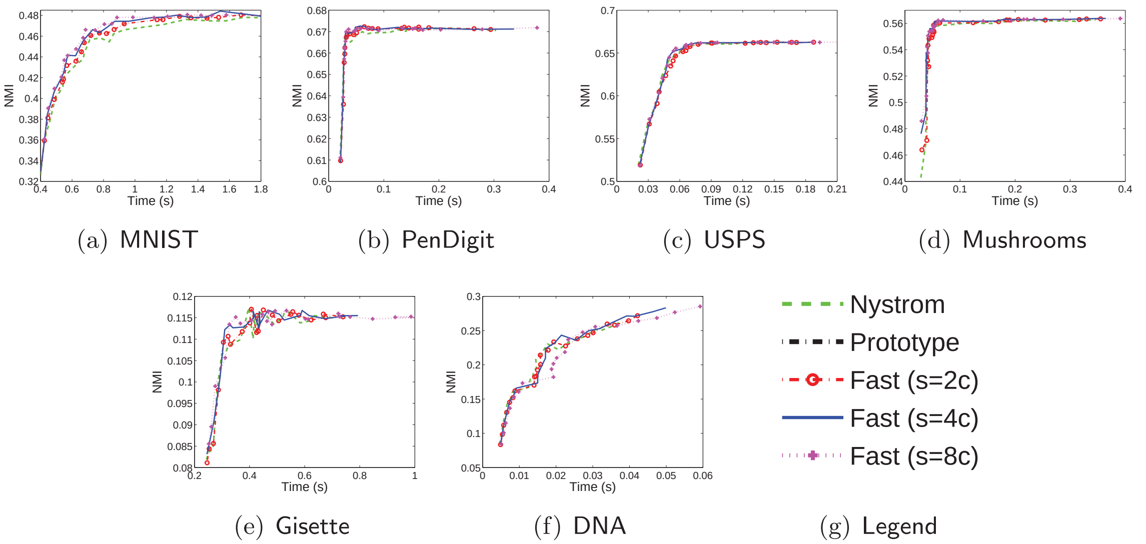
6.4 Approximate Spectral Clustering

Following the work of Fowlkes et al. (2004), we evaluate the performance of the kernel approximation methods on the spectral clustering task. We conduct experiments on the datasets summarized in Table 7.

We describe the approximate spectral clustering in the following. The target is to cluster n data points into k classes. We use the RBF kernel matrix $\mathbf{K}$ as the weigh matrix and let $\mathbf{C}\mathbf{U}\mathbf{C}^T \approx \mathbf{K}$ be the low-rank approximation. The degree matrix $\mathbf{D} = \mathsf{diag}(\mathbf{d})$ is a diagonal matrix with $\mathbf{d} = \mathbf{C}\mathbf{U}\mathbf{C}^T\mathbf{1}_n$, and the normalized graph Laplacian is $\mathbf{L} = \mathbf{I}_n - \mathbf{D}^{-1/2}(\mathbf{C}\mathbf{U}\mathbf{C}^T)\mathbf{D}^{-1/2}$. The bottom k eigenvectors of $\mathbf{L}$ are the top k eigenvectors of

$$\underbrace{(\mathbf{D}^{-1/2}\mathbf{C})}_{n\times c}\underbrace{\mathbf{U}}_{c\times c}\underbrace{(\mathbf{D}^{-1/2}\mathbf{C})^T}_{c\times n},$$

which can be efficiently computed according to Appendix A. We denote the top k eigenvectors by $\tilde{\mathbf{V}} \in \mathbb{R}^{n\times k}$. We normalize the rows of $\tilde{\mathbf{V}}$ and take the normalized rows



(a) MNIST (b) PenDigit (c) USPS (d) Mushrooms

(e) Gisette (f) DNA (g) Legend

Figure 12: The plot of elapsed time against NMI.

of $\tilde{\mathbf{V}}$ as the input of the k-means clustering. Since the matrix approximation methods are randomized, we repeat this procedure 20 times and record the average elapsed time and the average normalized mutual information (NMI)[3] of clustering.

We plot c against NMI in Figure 11 and the elapsed time (excluding the time cost of k-means) against NMI in Figure 12. Figure 11 shows that using the same amount of memory, the performance of the fast model is better than the Nyström method. Using the same amount of time, the fast model and the Nyström method have almost the same performance, and they are both better than the prototype model.

7. Concluding Remarks

In this paper we have studied the fast SPSD matrix approximation model for approximating large-scale SPSD matrix. We have shown that our fast model potentially costs time linear in n, while it is nearly as accurate as the best possible approximation. The fast model is theoretically better than the Nyström method and the prototype model because the latter two methods cost time quadratic in n to attain the same theoretical guarantee. Experiments show that our fast model is nearly as accurate as the prototype model and nearly as efficient as the Nyström method.

The technique of the fast model can be straightforwardly applied to speed up the CUR matrix decomposition, and theoretical analysis shows that the accuracy is almost unaffected. In this way, for any $m \times n$ large-scale matrix, the time cost of computing the $\mathbf{U}$ matrix drops from $\mathcal{O}(mn)$ to $\mathcal{O}(\min\{m, n\})$.

[3] NMI is a standard metric of clustering. NMI is between 0 and 1. Big NMI indicates good clustering performance.

Acknowledgments

We thank the anonymous reviewer for their helpful feedbacks. Shusen Wang acknowledges the support of Cray Inc., the Defense Advanced Research Projects Agency, the National Science Foundation, and the Baidu Scholarship. Zhihua Zhang acknowledges the support of National Natural Science Foundation of China (No. 61572017) and MSRA Collaborative Research Grant awards. Tong Zhang acknowledges NSF IIS-1250985, NSF IIS-1407939, and NIH R01AI116744.

Appendix A. Approximately Solving the Eigenvalue Decomposition and Matrix Inversion

In this section we show how to use the SPSD matrix approximation methods to speed up eigenvalue decomposition and linear system. The two lemmas are well known results. We show them here for the sake of self-containing.

Lemma 10 (Approximate Eigenvalue Decomposition) *Given* $\mathbf{C} \in \mathbb{R}^{n\times c}$ *and* $\mathbf{U} \in \mathbb{R}^{c\times c}$. *Then the eigenvalue decomposition of* $\tilde{\mathbf{K}} = \mathbf{C}\mathbf{U}\mathbf{C}^T$ *can be computed in time* $\mathcal{O}(nc^2)$.

Proof It cost $\mathcal{O}(nc^2)$ time to compute the SVD

$$\mathbf{C} = \underbrace{\mathbf{U}_{\mathbf{C}}}_{n\times c}\underbrace{\mathbf{\Sigma}_{\mathbf{C}}}_{c\times c}\underbrace{\mathbf{V}_{\mathbf{C}}^T}_{c\times c}$$

and $\mathcal{O}(c^3)$ time to compute $\mathbf{Z} = (\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^T)\mathbf{U}(\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^T)^T \in \mathbb{R}^{c\times c}$. It costs $\mathcal{O}(c^3)$ time to compute the eigenvalue decomposition $\mathbf{Z} = \mathbf{V}_{\mathbf{Z}}\mathbf{\Lambda}_{\mathbf{Z}}\mathbf{V}_{\mathbf{Z}}^T$. Combining the results above, we obtain

$$\begin{aligned} \mathbf{C}\mathbf{U}\mathbf{C}^T &= (\mathbf{U}_{\mathbf{C}}\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^T)\mathbf{U}(\mathbf{U}_{\mathbf{C}}\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^T)^T \\ &= \mathbf{U}_{\mathbf{C}}\mathbf{Z}\mathbf{U}_{\mathbf{C}}^T \;=\; (\mathbf{U}_{\mathbf{C}}\mathbf{V}_{\mathbf{Z}})\mathbf{\Lambda}_{\mathbf{Z}}(\mathbf{U}_{\mathbf{C}}\mathbf{V}_{\mathbf{Z}})^T. \end{aligned}$$

It then cost time $\mathcal{O}(nc^2)$ to compute the matrix product $\mathbf{U}_{\mathbf{C}}\mathbf{V}_{\mathbf{Z}}$. Since $(\mathbf{U}_{\mathbf{C}}\mathbf{V}_{\mathbf{Z}})$ has orthonormal columns and $\mathbf{\Lambda}_{\mathbf{Z}}$ is diagonal matrix, the eigenvalue decomposition of $\mathbf{C}\mathbf{U}\mathbf{C}^T$ is solved. The total time cost is $\mathcal{O}(nc^2) + \mathcal{O}(c^3) = \mathcal{O}(nc^2)$. ■

Lemma 11 (Approximately Solving Matrix Inversion) *Given* $\mathbf{C} \in \mathbb{R}^{n\times c}$, *SPDS matrix* $\mathbf{U} \in \mathbb{R}^{c\times c}$, *vector* $\mathbf{y} \in \mathbb{R}^n$, *and arbitrary positive real number* α. *Then it costs time* $\mathcal{O}(nc^2)$ *to solve the* $n \times n$ *linear system* $(\mathbf{C}\mathbf{U}\mathbf{C}^T + \alpha\mathbf{I}_n)\mathbf{w} = \mathbf{y}$ *to obtain* $\mathbf{w} \in \mathbb{R}^n$.

In addition, if the SVD of $\mathbf{C}$ *is given, then it takes only* $\mathcal{O}(c^3 + nc)$ *time to solve the linear system.*

Proof Since the matrix $(\mathbf{C}\mathbf{U}\mathbf{C}^T + \alpha\mathbf{I}_n)$ is nonsingular when $\alpha > 0$ and $\mathbf{U}$ is SPSD, the solution is $\mathbf{w}^\star = (\mathbf{C}\mathbf{U}\mathbf{C}^T + \alpha\mathbf{I}_n)^{-1}\mathbf{y}$. Instead of directly computing the matrix inversion, we can expand the matrix inversion by the Sherman-Morrison-Woodbury matrix identity and obtain

$$(\mathbf{C}\mathbf{U}\mathbf{C}^T + \alpha\mathbf{I}_n)^{-1} = \alpha^{-1}\mathbf{I}_n - \alpha^{-1}\mathbf{C}(\alpha\mathbf{U}^{-1} + \mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T.$$

Thus the solution to the linear system is

$$\mathbf{w}^\star = \alpha^{-1}\mathbf{y} - \alpha^{-1} \underbrace{\mathbf{C}}_{n\times c} \underbrace{(\alpha\mathbf{U}^{-1} + \mathbf{C}^T\mathbf{C})^{-1}}_{c\times c} \underbrace{\mathbf{C}^T}_{c\times n} \mathbf{y}.$$

Suppose we are given only $\mathbf{C}$ and $\mathbf{U}$. The matrix multiplication $\mathbf{C}^T\mathbf{C}$ costs time $\mathcal{O}(nc^2)$, the matrix inversions cost time $\mathcal{O}(c^3)$, and multiplying matrix with vector costs time $\mathcal{O}(nc)$. Thus the total time cost is $\mathcal{O}(nc^2) + \mathcal{O}(c^3) + \mathcal{O}(nc) = \mathcal{O}(nc^2)$.

Suppose we are given $\mathbf{U}$ and the SVD $\mathbf{C} = \mathbf{U}_\mathbf{C}\mathbf{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C}^T$. The matrix product

$$\mathbf{C}^T\mathbf{C} = \mathbf{V}_\mathbf{C}\mathbf{\Sigma}_\mathbf{C}\mathbf{U}_\mathbf{C}^T\mathbf{U}_\mathbf{C}\mathbf{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C} = \mathbf{V}_\mathbf{C}\mathbf{\Sigma}_\mathbf{C}^2\mathbf{V}_\mathbf{C}$$

can be computed in time $\mathcal{O}(c^3)$. Thus the total time cost is merely $\mathcal{O}(c^3 + nc)$. ■

Appendix B. Proof of Theorem 1

The prototype model trivially satisfies requirement R1 with $\epsilon = 0$. However, it violates requirement R2 because computing the $\mathbf{U}$ matrix by solving $\min_\mathbf{U} \|\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T\|_F^2$ costs time $\mathcal{O}(n^2c)$.

For the Nyström method, we provide such an adversarial case that assumptions A1 and A2 can both be satisfied and that requirements R1 and R2 cannot hold simultaneously. The adversarial case is the block diagonal matrix

$$\mathbf{K} = \mathsf{diag}(\underbrace{\mathbf{B}, \cdots, \mathbf{B}}_{k \text{ blocks}}),$$

where

$$\mathbf{B} = (1-a)\mathbf{I}_p + a\mathbf{1}_p\mathbf{1}_p^T, \qquad a < 1, \qquad \text{and } p = \frac{n}{k},$$

and let $a \to 1$. Wang et al. (2016) showed that sampling $c = 3k\gamma^{-1}\big(1 + o(1)\big)$ columns of $\mathbf{K}$ to form $\mathbf{C}$ makes assumptions A1 and A2 in Question 1 be satisfied. This indicates that $\mathbf{C}$ is a good sketch of $\mathbf{K}$. The problem is caused by the way the $\mathbf{U}^{\text{nys}}$ matrix is computed. Wang and Zhang (2013, Theorem 12) showed that to make requirement R1 in Question 1 satisfied, c must be greater than $\Omega(\sqrt{nk/(\epsilon+\gamma)})$. Thus it takes time $\mathcal{O}(nc^2) = \Omega(n^2k/(\epsilon+\gamma))$ to compute the rank-k eigenvalue decomposition of $\mathbf{C}\mathbf{U}^{\text{nys}}\mathbf{C}^T$ or the linear system $(\mathbf{C}\mathbf{U}^{\text{nys}}\mathbf{C}^T + \alpha\mathbf{I}_n)\mathbf{w} = \mathbf{y}$. Thus, requirement R2 is violated.

Appendix C. Proof of Lemma 2

Lemma 2 is a simplified version of Lemma 12. We prove Lemma 12 in the subsequent subsections. In the lemma, leverage score sampling means that the sampling probabilities are proportional to the row leverage scores of $\mathbf{U} \in \mathbb{R}^{n\times k}$. For uniform sampling, $\mu(\mathbf{U})$ is the row coherence of $\mathbf{U}$.

Lemma 12 *Let* $\mathbf{U} \in \mathbb{R}^{n\times k}$ *be any fixed matrix with orthonormal columns and* $\mathbf{B} \in \mathbb{R}^{n\times d}$ *be any fixed matrix. Let* $\mathbf{S} \in \mathbb{R}^{n\times s}$ *be any sketching matrix described in Table 8. Then*

$$\mathbb{P}\Big\{\big\|\mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U} - \mathbf{I}_k\big\|_2 \geq \eta\Big\} \;\leq\; \delta_1 \qquad \textit{(Property 1)},$$

$$\mathbb{P}\Big\{\big\|\mathbf{U}^T\mathbf{B} - \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{B}\big\|_F^2 \geq \epsilon\|\mathbf{B}\|_F^2\Big\} \;\leq\; \delta_2 \qquad \textit{(Property 2)},$$

$$\mathbb{P}\Big\{\big\|\mathbf{U}^T\mathbf{B} - \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{B}\big\|_2^2 \geq \epsilon'\|\mathbf{B}\|_2^2 + \frac{\epsilon'}{k}\|\mathbf{B}\|_F^2\Big\} \;\leq\; \delta_3 \qquad \textit{(Property 3)}.$$

Table 8: The sketch size s for satisfying the three properties. For SRHT, we define $\lambda = \big(1 + \sqrt{8k^{-1}\log(100n)}\big)^2$ and $\lambda' = \Big(1 + \sqrt{4k^{-1}\log\frac{nd}{k\delta_1}}\Big)^2$.

Sketching	Property 1	Property 2	Property 3
Leverage Sampling	$k\frac{6+2\eta}{3\eta^2}\log\frac{k}{\delta_1}$	$\frac{k}{\epsilon\delta_2}$	—
Uniform Sampling	$\mu(\mathbf{U})k\frac{6+2\eta}{3\eta^2}\log\frac{k}{\delta_1}$	$\frac{\mu(\mathbf{U})k}{\epsilon\delta_2}$	—
SRHT	$\lambda k\frac{6+2\eta}{3\eta^2}\log\frac{k}{\delta_1-0.01}$	$\frac{\lambda k}{\epsilon(\delta_2-0.01)}$	$\lambda' k\frac{24+4\sqrt{2\epsilon'}}{3\epsilon'}\log\frac{2d}{\delta_3-0.01}$
Gaussian Projection	$\frac{9\left(\sqrt{k}+\sqrt{2\log(2/\delta_1)}\right)^2}{\eta^2}$	$\frac{18k}{\epsilon\delta_2}$	$\frac{36k}{\epsilon'}\Big(1+\sqrt{k^{-1}\log\frac{2d}{k\delta_3}}\Big)^2$
Count Sketch	$\frac{k^2+k}{\delta_1\eta^2}$	$\frac{2k}{\epsilon\delta_2}$	—

C.1 Column Selection

In this subsection we prove Property 1 and Property 2 of leverage score sampling and uniform sampling. We cite the following lemma from (Wang et al., 2016); the lemma was firstly proved by the work Drineas et al. (2008); Gittens (2011); Woodruff (2014).

Lemma 13 *Let* $\mathbf{U} \in \mathbb{R}^{n\times k}$ *be any fixed matrix with orthonormal columns. The column selection matrix* $\mathbf{S} \in \mathbb{R}^{n\times s}$ *samples* s *columns according to arbitrary probabilities* $p_1, p_2, \cdots, p_n$*. Assume* $\alpha \geq k$ *and*

$$\max_{i\in[n]} \frac{\|\mathbf{u}_{i:}\|_2^2}{p_i} \;\leq\; \alpha.$$

If $s \;\geq\; \alpha\frac{6+2\eta}{3\eta^2}\log(k/\delta_1)$*, it holds that*

$$\mathbb{P}\Big\{\big\|\mathbf{I}_k - \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U}\big\|_2 \;\geq\; \eta\Big\} \;\leq\; \delta_1.$$

If $s \;\geq\; \frac{\alpha}{\epsilon\delta_2}$*, it holds that*

$$\mathbb{P}\Big\{\big\|\mathbf{U}\mathbf{B} - \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{B}\big\|_F^2 \;\geq\; \epsilon\|\mathbf{B}\|_F^2\Big\} \;\leq\; \delta_2.$$

Leverage score sampling satisfies $\max_{i\in[n]}\frac{\|\mathbf{u}_{i:}\|_2^2}{p_i} \;\leq\; k$. Uniform sampling satisfies $\max_{i\in[n]}\frac{\|\mathbf{u}_{i:}\|_2^2}{p_i} \leq \mu(\mathbf{U})k$, where $\mu(\mathbf{U})$ is the row coherence of $\mathbf{U}$. Then Property 1 and Property 2 of the two column sampling methods follow from Lemma 13.

Remark 14 *Let $p_1, \cdots, p_n$ be the sampling probabilities corresponding to the leverage score sampling or uniform sampling, and let $\tilde{p}_i \in [p_i, 1]$ for all $i \in [n]$ be arbitrary. For all $i \in [n]$, if the i-th column is sampled with probability $s\tilde{p_i}$ and scaled by $\frac{1}{\sqrt{s\tilde{p_i}}}$ if it gets sampled, then Lemma 2 still holds. This can be easily seen from the proof of the above lemma (in (Wang et al., 2016)). Intuitively, it indicates that if we increase the sampling probabilities, the resulting error bound will not get worse.*

C.2 Count Sketch

Count sketch stems from the data stream literature (Charikar et al., 2004; Thorup and Zhang, 2012). Theoretical guarantees were first shown by Weinberger et al. (2009); Pham and Pagh (2013); Clarkson and Woodruff (2013). Meng and Mahoney (2013); Nelson and Nguyên (2013) strengthened and simplified the proofs. Because the proof is involved, we will not show the proof here. The readers can refer to (Meng and Mahoney, 2013; Nelson and Nguyên, 2013; Woodruff, 2014) for the proof.

C.3 Property 1 and Property 2 of SRHT

The properties of SRHT were established in the previous work (Drineas et al., 2011; Lu et al., 2013; Tropp, 2011). Following (Tropp, 2011), we show a simple proof of the properties of SRHT. Our analysis is based on the following two key observations.

- The scaled Walsh-Hadamard matrix $\frac{1}{\sqrt{n}}\mathbf{H}_n$ and the diagonal matrix $\mathbf{D}$ are both orthogonal, so $\frac{1}{\sqrt{n}}\mathbf{D}\mathbf{H}_n$ is also orthogonal. If $\mathbf{U}$ has orthonormal columns, the matrix $\frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{U}$ has orthonormal columns.

- For any fixed matrix $\mathbf{U} \in \mathbb{R}^{n\times k}$ ($k \ll n$) with orthonormal columns, the matrix $\frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{U} \in \mathbb{R}^{n\times k}$ has low row coherence with high probability. Tropp (2011) showed that the row coherence of $\frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{U}$ satisfies

$$\mu \;\triangleq\; \frac{n}{k}\max_{i\in[n]}\Big\|\big(\frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{U}\big)_{i:}\Big\|_2^2 \;\leq\; \bigg(1+\sqrt{\frac{8\log(n/\delta)}{k}}\bigg)^2$$

  with probability at least $1-\delta$. In other words, the randomized Hadamard transform flats out the leverage scores. Consequently uniform sampling can be safely applied to form a sketch.

In the following, we use the properties of uniform sampling and the bound on the coherence μ to analyze SRHT. Let $\mathbf{V} \triangleq \frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{U} \in \mathbb{R}^{n\times k}$, $\bar{\mathbf{B}} \triangleq \frac{1}{\sqrt{n}}(\mathbf{D}\mathbf{H}_n)^T\mathbf{B} \in \mathbb{R}^{n\times d}$, and μ be the row coherence of $\mathbf{V}$. It holds that

$$\begin{aligned}
&\mathbf{V}^T\mathbf{V} = \mathbf{U}^T\mathbf{U} = \mathbf{I}_k, \qquad \mathbf{V}^T\mathbf{P}\mathbf{P}^T\mathbf{V} = \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U},\\
&\mathbf{V}^T\bar{\mathbf{B}} = \mathbf{U}^T\mathbf{B}, \quad \mathbf{V}^T\mathbf{P}\mathbf{P}^T\bar{\mathbf{B}} = \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{B}, \quad \|\bar{\mathbf{B}}\|_F = \|\mathbf{B}\|_F,\\
&\mathbb{P}\Big\{\mu \;>\; \big(1+\sqrt{8k^{-1}\log(100n)}\big)^2\Big\} \;\leq\; 0.01.
\end{aligned}$$

Therefore it suffices to prove that

$$\mathbb{P}\Big\{\big\|\mathbf{I}_k - \mathbf{V}^T\mathbf{P}\mathbf{P}^T\mathbf{V}\big\|_2 \;\geq\; \eta\Big\} \;\leq\; \delta_1 - 0.01,$$
$$\mathbb{P}\Big\{\big\|\mathbf{V}\bar{\mathbf{B}} - \mathbf{V}^T\mathbf{P}\mathbf{P}^T\bar{\mathbf{B}}\big\|_F^2 \;\geq\; \epsilon\|\bar{\mathbf{B}}\|_F^2\Big\} \;\leq\; \delta_2 - 0.01.$$

The above inequalities follows from the two properties of uniform sampling.

C.4 Property 1 and Property 2 of Gaussian Projection

The two properties of Gaussian projection can be found in (Woodruff, 2014). In the following we prove Property 1 in a much simpler way than (Woodruff, 2014).

The concentration of the singular values of standard Gaussian matrix is very well known. Let $\mathbf{G}$ be an $n \times s$ $(n > s)$ standard Gaussian matrix. For any fixed matrix $\mathbf{U} \in \mathbb{R}^{n\times k}$ with orthonormal columns, the matrix $\mathbf{N} = \mathbf{G}^T\mathbf{U} \in \mathbb{R}^{s\times k}$ is also standard Gaussian matrix. Vershynin (2010) showed that for every $t \geq 0$, the following holds with probability at least $1 - 2e^{-t^2/2}$:

$$\sqrt{s} - \sqrt{k} - t \leq \sigma_k(\mathbf{N}) \leq \sigma_1(\mathbf{N}) \leq \sqrt{s} + \sqrt{k} + t.$$

Therefore, for any $\eta \in (0,1)$, if $s = 9\eta^{-2}\big(\sqrt{k} + \sqrt{2\log(2/\delta_1)}\big)^2$, then

$$\sigma_i(\mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U}) \;=\; \sigma_i^2(\mathbf{S}^T\mathbf{U}) \;\in\; 1 \pm \eta \qquad \text{for all } i \in [n]$$

hold simultaneously with probability at least $1 - \delta_1$. Hence

$$\mathbb{P}\Big\{\big\|\mathbf{I}_k - \mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U}\big\|_2 \;\geq\; \eta\Big\} \;\leq\; \delta_1.$$

This concludes Property 1 of Gaussian projection.

C.5 Property 3 of SRHT and Gaussian Projection

The following lemma is the main result of (Cohen et al., 2015). If a sketching method satisfies Property 1 for arbitrary column orthogonal matrix $\mathbf{U}$, then it satisfies Property 3 due to the following lemma. Notice that the lemma does not apply to the leverage score and uniform sampling because they depends on the leverage scores or matrix coherence of specific column orthogonal matrix $\mathbf{U}$. The lemma is inappropriate for count sketch because Property 1 of count sketch holds with constant probability rather than arbitrary high probability.

Lemma 15 *Let $\mathbf{A} \in \mathbb{R}^{n\times k}$ and $\mathbf{B} \in \mathbb{R}^{n\times d}$ be any fixed matrices and r be any fixed integer. Let $\tilde{k} \geq k$ and $\tilde{d} \geq d$ be the least integer divisible by r. Let $\mathbf{S} \in \mathbb{R}^{n\times s}$ be a certain data-independent sketching matrix satisfying*

$$\mathbb{P}\Big\{\big\|\mathbf{U}^T\mathbf{S}\mathbf{S}^T\mathbf{U} - \mathbf{I}\big\|_2^2 \;\geq\; \eta\Big\} \;\leq\; \frac{r^2\delta_3}{\tilde{k}\tilde{d}}$$

for any fixed matrix $\mathbf{U} \in \mathbb{R}^{n\times 2r}$ with orthonormal columns. Then

$$\big\|\mathbf{A}^T\mathbf{S}\mathbf{S}^T\mathbf{B} - \mathbf{A}^T\mathbf{B}\big\|_2^2 \;\leq\; \eta\bigg(\|\mathbf{A}\|_2^2 + \frac{\|\mathbf{A}\|_F^2 - \|\mathbf{A}\|_2^2}{r}\bigg)\bigg(\|\mathbf{B}\|_2^2 + \frac{\|\mathbf{B}\|_F^2 - \|\mathbf{B}\|_2^2}{r}\bigg)$$

holds with probability at least $1 - \delta_3$.

SRHT and Gaussian projection enjoys Property 1 with high probability for arbitrary column orthogonal matrix $\mathbf{U}$. Thus Property 3 can be immediately obtained by applying the above lemma with the setting $r = k$.

Appendix D. Proof of Theorem 3

Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ be any fixed SPSD matrix, $\mathbf{C} \in \mathbb{R}^{n \times c}$ be any fixed matrix, $\mathbf{S} \in \mathbb{R}^{n \times s}$ be a sketching matrix, and

$$\begin{aligned} \mathbf{U}^\star &= \mathop{\mathrm{argmin}}_{\mathbf{U}} \big\| \mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T \big\|_F^2 \;=\; \mathbf{C}^\dagger \mathbf{K} (\mathbf{C}^T)^\dagger, \\ \tilde{\mathbf{U}} &= \mathop{\mathrm{argmin}}_{\mathbf{U}} \big\| \mathbf{S}^T (\mathbf{K} - \mathbf{C}\mathbf{U}\mathbf{C}^T) \mathbf{S} \big\|_F^2 \;=\; (\mathbf{S}^T\mathbf{C})^\dagger (\mathbf{S}^T\mathbf{K}\mathbf{S}) (\mathbf{C}^T\mathbf{S})^\dagger. \end{aligned}$$

Lemma 16 is a direct consequence of Lemma 24.

Lemma 16 *Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ be any fixed SPSD matrix, $\mathbf{C} \in \mathbb{R}^{n \times c}$ be any fixed matrix, and $\mathbf{C} = \mathbf{U}_\mathbf{C} \mathbf{\Sigma}_\mathbf{C} \mathbf{V}_\mathbf{C}^T$ be the SVD. Assume that $\mathbf{S}^T \mathbf{U}_\mathbf{C}$ has full column rank. Let $\mathbf{U}^\star$ and $\tilde{\mathbf{U}}$ be defined in the above. Then the following inequality holds:*

$$\|\mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T\|_F^2 \;\;\leq\;\; \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T\|_F^2 + \Big(2f\sqrt{h} + f^2 \sqrt{g_2 g_F} \Big)^2,$$

where $\alpha \in [0, 1]$ is arbitrary and

$$\begin{aligned} & f = \sigma_{\min}^{-1}(\mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T \mathbf{U}_\mathbf{C}), \qquad h = \big\| \mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T (\mathbf{K} - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T \mathbf{K}) \big\|_F^2, \\ & g_2 = \big\| \mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T) \mathbf{K}^\alpha \big\|_2^2, \qquad g_F = \big\| \mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T) \mathbf{K}^{1-\alpha} \big\|_F^2. \end{aligned}$$

The following lemma shows that $\tilde{\mathbf{X}}$ is nearly as good as $\mathbf{X}^\star$ in terms of objective function value if $\mathbf{S}$ satisfies Assumption 1.

Assumption 1 *Let $\mathbf{B}$ be any fixed matrix. Let $\mathbf{C} \in \mathbb{R}^{m \times c}$ and $\mathbf{C} = \mathbf{U}_\mathbf{C} \mathbf{\Sigma}_\mathbf{C} \mathbf{V}_\mathbf{C}^T$ be the SVD. Assume that the sketching matrix $\mathbf{S} \in \mathbb{R}^{m \times s}$ satisfies*

$$\begin{aligned} \mathbb{P}\Big\{ \big\| \mathbf{U}_\mathbf{C} \mathbf{S}\mathbf{S}^T \mathbf{U}_\mathbf{C} - \mathbf{I} \big\|_2 \;\geq\; \frac{1}{10} \Big\} &\;\;\leq\;\; \delta_1 \\ \mathbb{P}\Big\{ \big\| \mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T \mathbf{B} - \mathbf{U}_\mathbf{C}^T \mathbf{B} \big\|_F^2 \;\geq\; \epsilon \|\mathbf{B}\|_F^2 \Big\} &\;\;\leq\;\; \delta_2 \end{aligned}$$

for any $\delta_1, \delta_2 \in (0, 1/3)$.

Lemma 17 *Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ be any fixed SPSD matrix, $\mathbf{C} \in \mathbb{R}^{n \times c}$ be any fixed matrix, and $\mathbf{C} = \mathbf{U}_\mathbf{C} \mathbf{\Sigma}_\mathbf{C} \mathbf{V}_\mathbf{C}^T$ be the SVD. Let $\mathbf{U}^\star$ and $\tilde{\mathbf{U}}$ be defined in the above, respectively. Let $\mathbf{S} \in \mathbb{R}^{n \times s}$ be certain sketching matrix satisfying Assumption 1. Assume that $\epsilon^{-1} = o(n)$. Then*

$$\begin{aligned} & \big\| \mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T \big\|_F^2 - \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2 \\ \leq\;& \Big(\frac{20\sqrt{\epsilon}}{9} \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F + \frac{100\epsilon}{81} \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K} \big\|_* \Big)^2 \\ \leq\;& 4\epsilon^2 n \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2. \end{aligned}$$

holds with probability at least $1 - \delta_1 - 2\delta_2$.

Proof Let f, h, g_2, g_F, α be defined in Lemma 16 and fix $\alpha = 1/2$. Under Assumption 1 it holds simultaneously with probability at least $1 - \delta_1 - 2\delta_2$ that

$$f \leq \frac{10}{9}, \qquad h \leq \epsilon \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K} \big\|_F^2, \qquad g_2 \leq g_F \leq \epsilon \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K}^{1/2} \big\|_F^2.$$

It follows that

$$\begin{aligned} g_2 \leq g_F & \leq \epsilon \cdot \mathrm{tr}\Big((\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K}^{1/2}\mathbf{K}^{1/2}(\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T) \Big) \\ & \leq \epsilon \cdot \mathrm{tr}\Big((\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K}(\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T) \Big) \\ & = \epsilon \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K}(\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T) \big\|_* \\ & \leq \epsilon \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K} \big\|_*. \end{aligned}$$

It follows from Lemma 16 and the assumption $\epsilon^{-1} = o(n)$ that

$$\begin{aligned} & \big\| \mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T \big\|_F^2 - \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2 \\ \leq & \Big(\frac{20\sqrt{\epsilon}}{9} \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F + \frac{10^2 \epsilon}{9^2} \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K} \big\|_* \Big)^2 \\ \leq & \Big(\frac{20\sqrt{\epsilon}}{9} \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F + \frac{10^2 \epsilon\sqrt{n}}{9^2} \big\| (\mathbf{I}_n - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{K} \big\|_F \Big)^2 \\ = & \frac{10^4 \epsilon^2 n}{9^4} \big(1 + o(1)\big) \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2, \end{aligned}$$

by which the lemma follows. ■

Under both Assumption 1 and Assumption 2, the error bound can be further improved. We show the improved bound in Lemma 18.

Assumption 2 *Let $\mathbf{B}$ be any fixed matrix. Let $\mathbf{C} \in \mathbb{R}^{m \times c}$, $k_c = \mathrm{rank}(\mathbf{C})$, and $\mathbf{C} = \mathbf{U}_\mathbf{C}\boldsymbol{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C}^T$ be the SVD. Assume that the sketching matrix $\mathbf{S} \in \mathbb{R}^{m \times s}$ satisfies*

$$\mathbb{P}\Big\{ \big\| \mathbf{U}_\mathbf{C}^T \mathbf{S}\mathbf{S}^T \mathbf{B} - \mathbf{U}_\mathbf{C}^T \mathbf{B} \big\|_2^2 \ \geq \ \epsilon \|\mathbf{B}\|_2^2 + \frac{\epsilon}{k_c} \|\mathbf{B}\|_F^2 \Big\} \ \leq \ \delta_3$$

for any $\delta_3 \in (0, 1/3)$.

Lemma 18 *Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ be any fixed SPSD matrix, $\mathbf{C} \in \mathbb{R}^{n \times c}$ be any fixed matrix, $k_c = \mathrm{rank}(\mathbf{C})$, and $\mathbf{C} = \mathbf{U}_\mathbf{C}\boldsymbol{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C}^T$ be the SVD. Let $\mathbf{U}^\star$ and $\tilde{\mathbf{U}}$ be defined in the beginning of this section. Let $\mathbf{S} \in \mathbb{R}^{n \times s}$ be certain sketching matrix satisfying both Assumption 1 and Assumption 2. Assume that $\epsilon = o(n/k_c)$. Then*

$$\big\| \mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T \big\|_F^2 \ \leq \ \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2 + 4\epsilon^2 n / k_c \, \big\| \mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T \big\|_F^2$$

holds with probability at least $1 - \delta_1 - \delta_2 - \delta_3$.

Proof Let f, h, g_2, g_F, α be defined in Lemma 16 and fix $\alpha = 0$. Under Assumption 1 it holds simultaneously with probability at least $1 - \delta_1 - \delta_2$ that

$$f \leq \frac{10}{9}, \qquad h = g_F \leq \epsilon \big\|(\mathbf{I}_n - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{K}\big\|_F^2.$$

Under Assumption 2, it holds with probability at least $1 - \delta_3$ that

$$\begin{aligned} g_2 &= \big\|\mathbf{U}_{\mathbf{C}}^T\mathbf{S}\mathbf{S}^T(\mathbf{I}_n - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T) + \underbrace{\mathbf{U}_{\mathbf{C}}^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)}_{=\mathbf{0}}\big\|_2^2 \\ &\leq \epsilon\big\|\mathbf{I}_n - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\big\|_2^2 + \frac{\epsilon}{k_c}\big\|\mathbf{I}_n - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\big\|_F^2 \;\leq\; \epsilon + \frac{\epsilon}{k_c}(n - k_c) \;=\; \frac{\epsilon n}{k_c}. \end{aligned}$$

It follows from Lemma 16 and the assumption $\epsilon^{-1} = o(n/k_c)$ that

$$\begin{aligned} & \big\|\mathbf{K} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{C}^T\big\|_F^2 - \big\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T\big\|_F^2 \\ \leq\; & \Big(\frac{20\sqrt{\epsilon}}{9}\big\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T\big\|_F + \frac{10^2\epsilon}{9^2}\sqrt{n/k_c}\,\big\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T\big\|_F\Big)^2 \\ \leq\; & 4\epsilon^2 n/k_c\,\big\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{C}^T\big\|_F^2, \end{aligned}$$

by which the lemma follows. ■

Finally, we prove Theorem 3 using Lemma 17 and Lemma 18. Leverage score sampling, uniform sampling, and count sketch satisfy Assumption 1, and the bounds follow by setting $\epsilon = 0.5\sqrt{\epsilon'/n}$ and applying Lemma 17. For the three sketching methods, we set $\delta_1 = 0.01$ and $\delta_2 = 0.095$.

Gaussian projection and SRHT satisfy Assumption 1 and Assumption 2, and their bounds follow by setting $\epsilon = 0.5\sqrt{\epsilon' k_c/n}$ and applying Lemma 18. For Gaussian projection, we set $\delta_1 = 0.01$, $\delta_2 = 0.09$, and $\delta_3 = 0.1$. For SRHT, we set $\delta_1 = 0.02$, $\delta_2 = 0.08$, and $\delta_3 = 0.1$.

Appendix E. Proof of Theorem 6

Since $\mathbf{C} = \mathbf{K}\mathbf{P} \in \mathbb{R}^{n\times c}$, $\mathbf{W} = \mathbf{P}^T\mathbf{C} \in \mathbb{R}^{c\times c}$, and $\mathrm{rank}(\mathbf{S}^T\mathbf{C}) \geq \mathrm{rank}(\mathbf{W})$, we have that

$$\mathrm{rank}(\mathbf{K}) \;\geq\; \mathrm{rank}(\mathbf{C}) \;\geq\; \mathrm{rank}(\mathbf{S}^T\mathbf{C}) \;\geq\; \mathrm{rank}(\mathbf{W}). \tag{11}$$

If $\mathrm{rank}(\mathbf{C}) = \mathrm{rank}(\mathbf{K})$, there exists a matrix $\mathbf{X}$ such that $\mathbf{K} = \mathbf{C}\mathbf{X}$. By left multiplying both sides by $\mathbf{P}^T$, it follows that

$$\mathbf{C}^T \;=\; \mathbf{P}^T\mathbf{K} \;=\; \mathbf{P}^T\mathbf{C}\mathbf{X} \;=\; \mathbf{W}\mathbf{X},$$

and thus $\mathrm{rank}(\mathbf{W}) = \mathrm{rank}(\mathbf{S}^T\mathbf{C}) = \mathrm{rank}(\mathbf{C}) = \mathrm{rank}(\mathbf{K})$. It follows from $\mathbf{K} = \mathbf{C}\mathbf{X}$ and $\mathbf{C} = \mathbf{X}^T\mathbf{W}$ that

$$\mathbf{K} \;=\; \mathbf{X}^T\mathbf{W}\mathbf{X}.$$

We let $\mathbf{\Phi} = \mathbf{XS}$, and it holds that

$$\begin{aligned}\tilde{\mathbf{K}}_{c,s}^{\text{fast}} &= \mathbf{C}(\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{KS})(\mathbf{C}^T\mathbf{S})^\dagger\mathbf{C}^T \\ &= \mathbf{X}^T\mathbf{W}(\mathbf{S}^T\mathbf{X}^T\mathbf{W})^\dagger(\mathbf{S}^T\mathbf{X}^T\mathbf{WXS})(\mathbf{WXS})^\dagger\mathbf{WX} \\ &= \mathbf{X}^T\mathbf{W}(\mathbf{\Phi}^T\mathbf{W})^\dagger(\mathbf{\Phi}^T\mathbf{W}\mathbf{\Phi})(\mathbf{W}\mathbf{\Phi})^\dagger\mathbf{WX}.\end{aligned}$$

Let $\text{rank}(\mathbf{W}) = \text{rank}(\mathbf{C}) = \text{rank}(\mathbf{S}^T\mathbf{C}) = \text{rank}(\mathbf{K}) = \rho$. Since $\mathbf{W}$ is symmetric, we denote the rank-ρ eigenvalue decomposition of $\mathbf{W}$ by

$$\mathbf{W} = \underbrace{\mathbf{U_W}}_{c\times\rho}\underbrace{\mathbf{\Lambda_W}}_{\rho\times\rho}\underbrace{\mathbf{U_W}^T}_{\rho\times c}.$$

Since $\mathbf{S}^T\mathbf{C} = \mathbf{\Phi}^T\mathbf{W}$ and $\text{rank}(\mathbf{S}^T\mathbf{C}) = \text{rank}(\mathbf{W}) = \rho$, we have that $\text{rank}(\mathbf{\Phi}^T\mathbf{W}) = \text{rank}(\mathbf{W}) = \rho$. The $n \times \rho$ matrix $\mathbf{\Phi}^T\mathbf{U_W}$ must have full column rank, otherwise $\text{rank}(\mathbf{\Phi}^T\mathbf{W}) < \rho$. Thus we have

$$(\mathbf{\Phi}^T\mathbf{W})^\dagger = (\mathbf{\Phi}^T\mathbf{U_W}\mathbf{\Lambda_W}\mathbf{U_W}^T)^\dagger = (\mathbf{\Lambda_W}\mathbf{U_W}^T)^\dagger(\mathbf{\Phi}^T\mathbf{U_W})^\dagger.$$

It follows that

$$\begin{aligned}\tilde{\mathbf{K}}_{c,s}^{\text{fast}} &= \mathbf{X}^T\mathbf{W}\underbrace{(\mathbf{\Lambda_W}\mathbf{U_W}^T)^\dagger}_{c\times\rho}\underbrace{(\mathbf{\Phi}^T\mathbf{U_W})^\dagger}_{\rho\times n}\underbrace{(\mathbf{\Phi}^T\mathbf{U_W})}_{n\times\rho}\mathbf{\Lambda_W}(\mathbf{U_W}^T\mathbf{\Phi})(\mathbf{U_W}^T\mathbf{\Phi})^\dagger(\mathbf{U_W}\mathbf{\Lambda_W})^\dagger\mathbf{WX} \\ &= \mathbf{X}^T\mathbf{U_W}\mathbf{\Lambda_W}\mathbf{U_W}\mathbf{X} = \mathbf{X}^T\mathbf{WX} = \mathbf{K}.\end{aligned}$$

This shows that the fast model is exact. To this end, we have shown that if $\text{rank}(\mathbf{C}) = \text{rank}(\mathbf{K})$, then the fast model is exact.

Conversely, if the fast model is exact, that is, $\mathbf{K} = \mathbf{C}(\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{KS})(\mathbf{C}^T\mathbf{S})^\dagger\mathbf{C}^T$, we have that $\text{rank}(\mathbf{K}) \leq \text{rank}(\mathbf{C})$. It follows from (11) that $\text{rank}(\mathbf{K}) = \text{rank}(\mathbf{C})$.

Appendix F. Proof of Theorem 7

We prove Theorem 7 by constructing an adversarial case. Theorem 7 is a direct consequence of the following theorem.

Theorem 19 *Let $\mathbf{A}$ be the $n \times n$ symmetric matrix defined in Lemma 21 with $\alpha \to 1$ and k be any positive integer smaller than n. Let $\mathcal{P}$ be any subset of $[n]$ with cardinality c and $\mathbf{C} \in \mathbb{R}^{n\times c}$ contain c columns of $\mathbf{A}$ indexed by $\mathcal{P}$. Let $\mathbf{S}$ be any $n \times s$ column selection matrix satisfying $\mathcal{P} \subset \mathcal{S}$, where $\mathcal{S} \subset [n]$ is the index set formed by $\mathbf{S}$. Then the following inequality holds:*

$$\frac{\|\mathbf{A} - \mathbf{C}(\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{AS})(\mathbf{C}^T\mathbf{S})^\dagger\mathbf{C}^T\|_F^2}{\|\mathbf{A} - \mathbf{A}_k\|_F^2} \geq \frac{n-c}{n-k}\Big(1 + \frac{2k}{c}\Big) + \frac{n-s}{n-k}\frac{k(n-s)}{s^2}.$$

Proof Let $\mathbf{A}$ and $\mathbf{B}$ be defined in Lemma 21. We prove the theorem using Lemma 21 and Lemma 23. Let $n = pk$. Let $\mathbf{C}$ consist of c column sampled from $\mathbf{A}$ and $\hat{\mathbf{C}}_i$ consist of c_i columns sampled from the i-th diagonal block of $\mathbf{A}$. Thus $\mathbf{C} = \mathsf{diag}(\hat{\mathbf{C}}_1, \cdots, \hat{\mathbf{C}}_k)$. Without loss of generality, we assume $\hat{\mathbf{C}}_i$ consists of the first c_i columns of $\mathbf{B}$. Let $\hat{\mathbf{S}} =$

$\mathsf{diag}\big(\hat{\mathbf{S}}_1, \cdots, \hat{\mathbf{S}}_k\big)$ be an $n \times s$ column selection matrix, where $\hat{\mathbf{S}}_i$ is a $p \times s_i$ column selection matrix and $s_1 + \cdots s_k = s$. Then the $\mathbf{U}$ matrix is computed by

$$\begin{aligned}\mathbf{U} &= \big(\mathbf{S}^T\mathbf{C}\big)^\dagger\big(\mathbf{S}^T\mathbf{A}\mathbf{S}\big)\big(\mathbf{C}^T\mathbf{S}\big)^\dagger \\ &= \big[\mathsf{diag}\big(\hat{\mathbf{S}}_1^T\hat{\mathbf{C}}_1, \cdots, \hat{\mathbf{S}}_k^T\hat{\mathbf{C}}_k\big)\big]^\dagger \mathsf{diag}\big(\hat{\mathbf{S}}_1^T\mathbf{B}\hat{\mathbf{S}}_1, \cdots, \hat{\mathbf{S}}_k^T\mathbf{B}\hat{\mathbf{S}}_k\big)\big[\mathsf{diag}\big(\hat{\mathbf{C}}_1^T\hat{\mathbf{S}}_1, \cdots, \hat{\mathbf{C}}_k^T\hat{\mathbf{S}}_k\big)\big]^\dagger \\ &= \mathsf{diag}\Big(\big(\hat{\mathbf{S}}_1^T\hat{\mathbf{C}}_1\big)^\dagger\big(\hat{\mathbf{S}}_1^T\mathbf{B}\hat{\mathbf{S}}_1\big)\big(\hat{\mathbf{C}}_1^T\hat{\mathbf{S}}_1\big)^\dagger, \cdots, \big(\hat{\mathbf{S}}_k^T\hat{\mathbf{C}}_k\big)^\dagger\big(\hat{\mathbf{S}}_k^T\mathbf{B}\hat{\mathbf{S}}_k\big)\big(\hat{\mathbf{C}}_k^T\hat{\mathbf{S}}_k\big)^\dagger\Big).\end{aligned}$$

The approximation formed by the fast model is the block-diagonal matrix whose the i-th ($i \in [k]$) diagonal block is the $p \times p$ matrix

$$\big[\tilde{\mathbf{A}}_{c,s}^{\textrm{fast}}\big]_{ii} \;=\; \hat{\mathbf{C}}_i\big(\hat{\mathbf{S}}_i^T\hat{\mathbf{C}}_i\big)^\dagger\big(\hat{\mathbf{S}}_i^T\mathbf{B}\hat{\mathbf{S}}_i\big)\big(\hat{\mathbf{C}}_i^T\hat{\mathbf{S}}_i\big)^\dagger\hat{\mathbf{C}}_i^T.$$

It follows from Lemma 23 that for any $i \in [k]$,

$$\lim_{\alpha\to1}\frac{\|\mathbf{B} - \big[\tilde{\mathbf{A}}_{c,s}^{\textrm{fast}}\big]_{ii}\|_F^2}{(1-\alpha)^2} \;=\; (p-c_i)\Big(1+\frac{2}{c_i}\Big) + \frac{(p-s_i)^2}{s_i^2}.$$

Thus

$$\begin{aligned}\lim_{\alpha\to1}\frac{\|\mathbf{A} - \tilde{\mathbf{A}}_{c,s}^{\textrm{fast}}\|_F^2}{(1-\alpha)^2} &= \lim_{\alpha\to1}\sum_{i=1}^k\frac{\|\mathbf{B} - \big[\tilde{\mathbf{A}}_{c,s}^{\textrm{fast}}\big]_{ii}\|_F^2}{(1-\alpha)^2} \\ &= \sum_{i=1}^k(p-c_i)\Big(1+\frac{2}{c_i}\Big) + \frac{(p-s_i)^2}{s_i^2} \\ &= \Big(\sum_{i=1}^k p - c_i - 2\Big) + \Big(2p\sum_{i=1}^k\frac{1}{c_i}\Big) + \Big(p^2\sum_{i=1}^k\frac{1}{s_i^2}\Big) - \Big(2p\sum_{i=1}^k\frac{1}{s_i}\Big) + k \\ &\geq n - c - 2k + \frac{2nk}{c} + \frac{kn^2}{s^2} - \frac{2nk}{s} + k \\ &= (n-c)\Big(1+\frac{2k}{c}\Big) + \frac{k(n-s)^2}{s^2}.\end{aligned}$$

Here the inequality follows by minimizing over $c_1, \cdots, c_k$ and $s_1, \cdots, s_k$ with constraints $\sum_i c_i = c$ and $\sum_i s_i = s$. Finally, it follows from Lemma 21 that

$$\lim_{\alpha\to1}\frac{\|\mathbf{A} - \tilde{\mathbf{A}}_{c,s}^{\textrm{fast}}\|_F^2}{\|\mathbf{A} - \mathbf{A}_k\|_F^2} \;\geq\; \frac{n-c}{n-k}\Big(1+\frac{2k}{c}\Big) + \frac{n-s}{n-k}\frac{k(n-s)}{s^2}.$$

■

F.1 Key Lemmas

Lemma 20 provides a useful tool for expanding the Moore-Penrose inverse of partitioned matrices.

Lemma 20 (Page 179 of Ben-Israel and Greville (2003)) *Given a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ of rank c which has a nonsingular $c \times c$ submatrix $\mathbf{X}_{11}$. By rearrangement of columns and rows by permutation matrices $\mathbf{P}$ and $\mathbf{Q}$, the submatrix $\mathbf{X}_{11}$ can be bought to the top left corner of $\mathbf{X}$, that is,*

$$\mathbf{P}\mathbf{X}\mathbf{Q} \;=\; \left[\begin{array}{cc} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{array} \right].$$

Then the Moore-Penrose inverse of $\mathbf{X}$ is

$$\mathbf{X}^{\dagger} \;=\; \mathbf{Q} \left[\begin{array}{c} \mathbf{I}_c \\ \mathbf{T}^T \end{array} \right] \big(\mathbf{I}_c + \mathbf{T}\mathbf{T}^T\big)^{-1} \mathbf{X}_{11}^{-1} \big(\mathbf{I}_c + \mathbf{H}^T\mathbf{H}\big)^{-1} \left[\begin{array}{cc} \mathbf{I}_c & \mathbf{H}^T \end{array} \right] \mathbf{P},$$

where $\mathbf{T} = \mathbf{X}_{11}^{-1}\mathbf{X}_{12}$ and $\mathbf{H} = \mathbf{X}_{21}\mathbf{X}_{11}^{-1}$.

Lemmas 21 and 23 will be used to prove Theorem 19.

Lemma 21 (Lemma 19 of Wang and Zhang (2013)) *Given n and k, we let $\mathbf{B}$ be an $\frac{n}{k} \times \frac{n}{k}$ matrix whose diagonal entries equal to one and off-diagonal entries equal to $\alpha \in [0, 1)$. We let $\mathbf{A}$ be an $n \times n$ block-diagonal matrix*

$$\mathbf{A} \;=\; \mathsf{diag}(\underbrace{\mathbf{B}, \cdots, \mathbf{B}}_{k \text{ blocks}}). \tag{12}$$

Let $\mathbf{A}_k$ be the best rank-k approximation to the matrix $\mathbf{A}$, then we have that

$$\|\mathbf{A} - \mathbf{A}_k\|_F^2 \;=\; (1-\alpha)^2 (n-k).$$

Lemma 22 *The following equality holds for any nonzero real number a:*

$$\big(a\mathbf{I}_c + b\mathbf{1}_c\mathbf{1}_c^T\big)^{-1} \;=\; a^{-1}\mathbf{I}_c - \frac{b}{a(a+bc)}\mathbf{1}_c\mathbf{1}_c^T.$$

Proof The lemma directly follows from the Sherman-Morrison-Woodbury matrix identity

$$(\mathbf{X} + \mathbf{Y}\mathbf{Z}\mathbf{R})^{-1} = \mathbf{X}^{-1} - \mathbf{X}^{-1}\mathbf{Y}(\mathbf{Z}^{-1} + \mathbf{R}\mathbf{X}^{-1}\mathbf{Y})^{-1}\mathbf{R}\mathbf{X}^{-1}.$$

■

Lemma 23 *Let $\mathbf{B}$ be any $n \times n$ matrix with diagonal entries equal to one and off-diagonal entries equal to α. Let $\mathbf{C} = \mathbf{B}\mathbf{P} \in \mathbb{R}^{n \times c}$; let $\tilde{\mathbf{B}} = \mathbf{C}(\mathbf{S}^T\mathbf{C})^{\dagger}(\mathbf{S}^T\mathbf{K}\mathbf{S})(\mathbf{C}^T\mathbf{S})^{\dagger}\mathbf{C}^T$ be the fast SPSD matrix approximation model of $\mathbf{B}$. Let $\mathcal{P}$ and $\mathcal{S}$ be the index sets formed by $\mathbf{P}$ and $\mathbf{S}$, respectively. If $\mathcal{P} \subset \mathcal{S}$, the error incurred by the fast model satisfies*

$$\lim_{\alpha \to 1} \frac{\|\mathbf{B} - \tilde{\mathbf{B}}\|_F^2}{(1-\alpha)^2} \;\geq\; (n-c)\Big(1 + \frac{2}{c}\Big) + \frac{(n-s)^2}{s^2}.$$

Proof Let $\mathbf{B}_1 = \mathbf{S}^T\mathbf{B}\mathbf{S} \in \mathbb{R}^{s\times s}$ and $\mathbf{C}_1 = \mathbf{S}^T\mathbf{C} = \mathbf{S}^T\mathbf{B}\mathbf{P} \in \mathbb{R}^{s\times c}$. Without loss of generality, we assume that $\mathbf{P}$ selects the first c columns and $\mathbf{S}$ selects the first s columns. We partition $\mathbf{B}$ and $\mathbf{C}$ by:

$$\mathbf{B} = \left[\begin{array}{cc} \mathbf{B}_1 & \mathbf{B}_3^T \\ \mathbf{B}_3 & \mathbf{B}_2 \end{array}\right] \qquad \text{and} \qquad \mathbf{C} = \left[\begin{array}{c} \mathbf{C}_1 \\ \mathbf{C}_2 \end{array}\right] = \left[\begin{array}{c} \mathbf{W} \\ \mathbf{C}_{12} \\ \mathbf{C}_2 \end{array}\right].$$

We further partition $\mathbf{B}_1 \in \mathbb{R}^{s\times s}$ by

$$\mathbf{B}_1 = \left[\begin{array}{cc} \mathbf{W} & \mathbf{C}_{12}^T \\ \mathbf{C}_{12} & \mathbf{B}_{12} \end{array}\right],$$

where

$$\mathbf{C}_{12} = \alpha \mathbf{1}_{s-c}\mathbf{1}_c^T \quad \text{and} \quad \mathbf{B}_{12} = (1-\alpha)\mathbf{I}_{s-c} + \alpha\mathbf{1}_{s-c}\mathbf{1}_{s-c}^T.$$

The $\mathbf{U}$ matrix is computed by

$$\mathbf{U} = (\mathbf{S}^T\mathbf{C})^\dagger(\mathbf{S}^T\mathbf{B}\mathbf{S})(\mathbf{C}^T\mathbf{S})^\dagger = \mathbf{C}_1^\dagger\mathbf{B}_1(\mathbf{C}_1^\dagger)^T.$$

It is not hard to see that $\mathbf{C}_1$ contains the first c rows of $\mathbf{B}_1$.

We expand the Moore-Penrose inverse of $\mathbf{C}_1$ by Lemma 20 and obtain

$$\mathbf{C}_1^\dagger = \mathbf{W}^{-1}\big(\mathbf{I}_c + \mathbf{H}^T\mathbf{H}\big)^{-1}\left[\begin{array}{cc} \mathbf{I}_c & \mathbf{H}^T \end{array}\right],$$

where

$$\mathbf{W}^{-1} \;=\; \Big((1-\alpha)\mathbf{I}_c + \alpha\mathbf{1}_c\mathbf{1}_c^T\Big)^{-1} \;=\; \frac{1}{1-\alpha}\mathbf{I}_c - \frac{\alpha}{(1-\alpha)(1-\alpha+c\alpha)}\mathbf{1}_c\mathbf{1}_c^T$$

and

$$\mathbf{H} = \mathbf{C}_{12}\mathbf{W}^{-1} = \frac{\alpha}{1-\alpha+c\alpha}\mathbf{1}_{s-c}\mathbf{1}_c^T.$$

It is easily verified that $\mathbf{H}^T\mathbf{H} = \big(\frac{\alpha}{1-\alpha+c\alpha}\big)^2(s-c)\mathbf{1}_c\mathbf{1}_c^T$. It follows from Lemma 22 that

$$(\mathbf{I}_c + \mathbf{H}^T\mathbf{H})^{-1} \;=\; \mathbf{I}_c - \frac{(s-c)\alpha^2}{c(s-c)\alpha^2 + (1-\alpha+c\alpha)^2}\mathbf{1}_c\mathbf{1}_c^T.$$

Then we obtain

$$\begin{aligned} \mathbf{C}_1^\dagger &= \mathbf{W}^{-1}\big(\mathbf{I}_c + \mathbf{H}^T\mathbf{H}\big)^{-1}\left[\begin{array}{cc} \mathbf{I}_c & \mathbf{H}^T \end{array}\right] \\ &= \Big(\frac{1}{1-\alpha}\mathbf{I}_c + \gamma_1\mathbf{1}_c\mathbf{1}_c^T\Big)\left[\begin{array}{cc} \mathbf{I}_c & \mathbf{H}^T \end{array}\right], \end{aligned} \tag{13}$$

where

$$\begin{aligned} \gamma_1 &= c\gamma_2\gamma_3 - \gamma_2 - \frac{\gamma_3}{1-\alpha}, \\ \gamma_2 &= \frac{\alpha}{(1-\alpha)(1-\alpha+c\alpha)}, \\ \gamma_3 &= \frac{(s-c)\alpha^2}{c(s-c)\alpha^2 + (1-\alpha+c\alpha)^2}. \end{aligned}$$

Then

$$\begin{aligned}[\mathbf{I}_c, \mathbf{H}^T]\mathbf{B}_1[\mathbf{I}_c, \mathbf{H}^T]^T &= \mathbf{W} + \mathbf{B}_{13}^T\mathbf{H} + \mathbf{H}^T\mathbf{B}_{13} + \mathbf{H}^T\mathbf{B}_{12}\mathbf{H} \\ &= (1-\alpha)\mathbf{I}_c + \gamma_4 \mathbf{1}_c\mathbf{1}_c^T, \end{aligned} \tag{14}$$

where

$$\gamma_4 = \frac{\alpha(3\alpha s - \alpha c - 2\alpha + \alpha^2 c - 3\alpha^2 s + \alpha^2 + \alpha^2 s^2 + 1)}{(\alpha c - \alpha + 1)^2}.$$

It follows from (13) (14) that

$$\begin{aligned}\mathbf{U} = \mathbf{C}_1^\dagger \mathbf{B}_1 (\mathbf{C}_1^\dagger)^T &= \Big(\frac{1}{1-\alpha}\mathbf{I}_c + \gamma_1 \mathbf{1}_c\mathbf{1}_c^T\Big)\Big((1-\alpha)\mathbf{I}_c + \gamma_4 \mathbf{1}_c\mathbf{1}_c^T\Big)\Big(\frac{1}{1-\alpha}\mathbf{I}_c + \gamma_1 \mathbf{1}_c\mathbf{1}_c^T\Big) \\ &= \frac{1}{1-\alpha}\mathbf{I}_c + \gamma_5 \mathbf{1}_c\mathbf{1}_c^T,\end{aligned}$$

where

$$\gamma_5 \;=\; \gamma_1 + \Big(c\gamma_1 + \frac{1}{1-\alpha}\Big)\Big(c\gamma_1\gamma_4 + \gamma_1(1-\alpha) + \frac{\gamma_4}{1-\alpha}\Big).$$

Then we have

$$\begin{aligned}\mathbf{W}\mathbf{U} &= \mathbf{I}_c + \gamma_6 \mathbf{1}_c\mathbf{1}_c^T, \\ \gamma_6 &= (1-\alpha+\alpha c)\gamma_5 + \frac{\alpha}{1-\alpha}.\end{aligned}$$

We partition the fast SPSD matrix approximation model by

$$\tilde{\mathbf{B}} = \left[\begin{array}{cc} \tilde{\mathbf{W}} & \tilde{\mathbf{B}}_{21}^T \\ \tilde{\mathbf{B}}_{21} & \tilde{\mathbf{B}}_{22} \end{array}\right],$$

where

$$\begin{aligned}\tilde{\mathbf{B}}_{11} &= \mathbf{W}\mathbf{U}\mathbf{W} \;=\; (1-\alpha)\mathbf{I}_c + \big(\alpha + (1-\alpha+c\alpha)\gamma_6\big)\mathbf{1}_c\mathbf{1}_c^T, \\ \tilde{\mathbf{B}}_{21} &= \mathbf{W}\mathbf{U}\big(\alpha\mathbf{1}_c\mathbf{1}_{n-c}^T\big) \;=\; \alpha(1+c\gamma_6)\mathbf{1}_c\mathbf{1}_{n-c}^T, \\ \tilde{\mathbf{B}}_{22} &= \big(\alpha\mathbf{1}_{n-c}\mathbf{1}_c^T\big)\mathbf{U}\big(\alpha\mathbf{1}_c\mathbf{1}_{n-c}^T\big) \;=\; \alpha^2 c\Big(\frac{1}{1-\alpha} + \gamma_5 c\Big)\mathbf{1}_c\mathbf{1}_{n-c}^T\end{aligned}$$

The approximate error is

$$\big\|\mathbf{B} - \tilde{\mathbf{B}}\big\|_F^2 \;=\; \big\|\mathbf{W} - \tilde{\mathbf{W}}\big\|_F^2 + 2\big\|\mathbf{B}_{21} - \tilde{\mathbf{B}}_{21}\big\|_F^2 + \big\|\mathbf{B}_{22} - \tilde{\mathbf{B}}_{22}\big\|_F^2,$$

where

$$\begin{aligned}\big\|\mathbf{W} - \tilde{\mathbf{W}}\big\|_F^2 &= \big\|(1-\alpha+c\alpha)\gamma_6\mathbf{1}_c\mathbf{1}_c^T\big\|_F^2 \;=\; c^2(1-\alpha+c\alpha)^2\gamma_6^2, \\ \big\|\mathbf{B}_{21} - \tilde{\mathbf{B}}_{21}\big\|_F^2 &= \big\|\alpha c\gamma_6\mathbf{1}_c\mathbf{1}_{n-c}^T\big\|_F^2 \;=\; \alpha^2 c^3(n-c)\gamma_6^2, \\ \big\|\mathbf{B}_{22} - \tilde{\mathbf{B}}_{22}\big\|_F^2 &= \underbrace{(n-c)(n-c-1)\alpha^2\Big(\frac{\alpha c}{1-\alpha} + \alpha c^2\gamma_5 - 1\Big)^2}_{\text{off-diagonal}} + \underbrace{(n-c)\Big(\frac{\alpha^2 c}{1-\alpha} + \alpha^2 c^2\gamma_5 - 1\Big)^2}_{\text{diagonal}}.\end{aligned}$$

We let

$$\eta \triangleq \frac{\|\mathbf{B} - \tilde{\mathbf{B}}\|_F^2}{(1-\alpha)^2},$$

which is a symbolic expression of α, n, s, and c. We then simplify the expression using MATLAB and substitute the α in η by 1, and we obtain

$$\lim_{\alpha \to 1} \eta = (n-c)(1+2/c) + (n-s)^2/s^2,$$

by which the lemma follows. ■

Appendix G. Proof of Theorem 8

We define the projection operation $\mathcal{P}_{\mathbf{C},k}(\mathbf{A}) = \mathbf{C}\mathbf{X}$ where $\mathbf{X}$ is defined by

$$\mathbf{X} \;=\; \operatorname*{argmin}_{\mathrm{rank}(\mathbf{X}) \leq k} \big\|\mathbf{A} - \mathbf{C}\mathbf{X}\big\|_F^2.$$

By sampling $c = 2k\epsilon^{-1}\big(1 + o(1)\big)$ columns of $\mathbf{A}$ by the near-optimal algorithm of Boutsidis et al. (2014) to form $\mathbf{C} \in \mathbb{R}^{m \times c_1}$, we have that

$$\mathbb{E}\big\|\mathbf{A} - \mathcal{P}_{\mathbf{C},k}(\mathbf{A})\big\|_F^2 \;\leq\; (1+\epsilon)\big\|\mathbf{A} - \mathbf{A}_k\big\|_F^2.$$

Applying Lemma 3.11 of Boutsidis and Woodruff (2014), there exists a much smaller column orthogonal matrix $\mathbf{Z} \in \mathbb{R}^{m \times k}$ such that $\mathrm{range}(\mathbf{Z}) \subset \mathrm{range}(\mathbf{C})$ and

$$\mathbb{E}\big\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\big\|_F^2 \;\leq\; \mathbb{E}\big\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\big\|_F^2 \;\leq\; \big\|\mathbf{A} - \mathcal{P}_{\mathbf{C},k}(\mathbf{A})\big\|_F^2.$$

Notice that the algorithm does not compute $\mathbf{Z}$.

Let $\mathbf{R}_1^T \in \mathbb{R}^{n \times r_1}$ be columns of $\mathbf{A}^T$ selected by the randomized dual-set sparsification algorithm of Boutsidis et al. (2014). When $r_1 = \mathcal{O}(k)$, it holds that

$$\mathbb{E}\big\|\mathbf{A} - \mathbf{R}_1\mathbf{R}_1^T\mathbf{A}\big\|_F^2 \;\leq\; 2(1+o(1))\|\mathbf{A} - \mathbf{A}_k\|_F^2.$$

Let $\mathbf{R}_2^T \in \mathbb{R}^{n \times r_2}$ be columns of $\mathbf{A}^T$ selected by adaptive sampling according to the residual $\mathbf{A}^T - \mathbf{R}_1^T(\mathbf{R}_1^T)^\dagger\mathbf{A}^T$. Set $r_2 = 2k\epsilon^{-1}\big(1+o(1)\big)$. Let $\mathbf{R}^T = [\mathbf{R}_1^T, \mathbf{R}_2^T]$. By the adaptive sampling theorem of Wang and Zhang (2013), we obtain

$$\begin{aligned}
\mathbb{E}\big\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\big\|_F^2 \;&\leq\; \mathbb{E}\big\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\big\|_F^2 + \frac{k}{r_2}\mathbb{E}\big\|\mathbf{A} - \mathbf{A}\mathbf{R}_1^\dagger\mathbf{R}_1^T\big\|_F^2 \\
&\leq\; (1+\epsilon)\big\|\mathbf{K} - \mathbf{K}_k\big\|_F^2 + \epsilon\big\|\mathbf{K} - \mathbf{K}_k\big\|_F^2 \\
&\leq\; (1+2\epsilon)\big\|\mathbf{K} - \mathbf{K}_k\big\|_F^2. \qquad (15)
\end{aligned}$$

Obviously $\mathbf{R}^T$ contains

$$r = r_1 + r_2 = 2k\epsilon^{-1}\big(1 + o(1)\big)$$

columns of $\mathbf{A}^T$.

It remains to show $\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \leq \|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2$. Since the columns of $\mathbf{Z}$ are contained in the column space of $\mathbf{C}$, for any matrix $\mathbf{Y}$ the inequality $\|(\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger)\mathbf{Y}\|_F^2 \leq (\mathbf{I}_m - \mathbf{Z}\mathbf{Z}^T)\mathbf{Y}\|_F^2$ holds. Then we obtain

$$\begin{aligned}
\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 &= \|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R} + \mathbf{A}\mathbf{R}^\dagger\mathbf{R} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \\
&= \|\mathbf{A}(\mathbf{I}_n - \mathbf{R}^\dagger\mathbf{R})\|_F^2 + \|(\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger)\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \\
&\leq \|\mathbf{A}(\mathbf{I}_n - \mathbf{R}^\dagger\mathbf{R})\|_F^2 + \|(\mathbf{I}_m - \mathbf{Z}\mathbf{Z}^T)\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \\
&= \|\mathbf{A}(\mathbf{I}_n - \mathbf{R}^\dagger\mathbf{R}) + (\mathbf{I}_m - \mathbf{Z}\mathbf{Z}^T)\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2 \\
&= \|\mathbf{A} - \mathbf{Z}\mathbf{Z}^T\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\|_F^2. \qquad (16)
\end{aligned}$$

The theorem follows from (15) and (16) and by setting $\epsilon' = 2\epsilon$.

Appendix H. Proof of Theorem 9

In Section H.1 we establish a key lemma to decompose the error incurred by the approximation. In Section H.2 we prove Theorem 9 using the key lemma.

H.1 Key Lemma

We establish the following lemma for decomposing the error of the approximate solution.

Lemma 24 *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{C} \in \mathbb{R}^{m\times c}$, and $\mathbf{R} \in \mathbb{R}^{r\times n}$ be any fixed matrices, and $\mathbf{A} = \mathbf{U}_\mathbf{A}\mathbf{\Sigma}_\mathbf{A}\mathbf{V}_\mathbf{A}^T$, $\mathbf{C} = \mathbf{U}_\mathbf{C}\mathbf{\Sigma}_\mathbf{C}\mathbf{V}_\mathbf{C}^T$, $\mathbf{R} = \mathbf{U}_\mathbf{R}\mathbf{\Sigma}_\mathbf{R}\mathbf{V}_\mathbf{R}^T$ be the SVD. Assume that $\mathbf{S}_C^T\mathbf{U}_\mathbf{C}$ and $\mathbf{S}_R^T\mathbf{V}_\mathbf{R}$ have full column rank. Let $\mathbf{U}^\star$ and $\tilde{\mathbf{U}}$ be defined in (8) and (9), respectively. Then the following inequalities hold:*

$$\begin{aligned}
\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 &\leq \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 + \Big(f_R\sqrt{h_R} + f_C\sqrt{h_C} + f_C f_R\sqrt{g_C' g_R}\Big)^2, \\
\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 &\leq \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 + \Big(f_R\sqrt{h_R} + f_C\sqrt{h_C} + f_C f_R\sqrt{g_C g_R'}\Big)^2,
\end{aligned}$$

where $\alpha \in [0, 1]$ is arbitrary, and

$$\begin{aligned}
f_C &= \sigma_{\min}^{-1}(\mathbf{U}_\mathbf{C}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_\mathbf{C}), & f_R &= \sigma_{\min}^{-1}(\mathbf{V}_\mathbf{R}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_\mathbf{R}), \\
h_C &= \big\|\mathbf{U}_\mathbf{C}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{A} - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T\mathbf{A})\big\|_F^2, & h_R &= \big\|(\mathbf{A} - \mathbf{A}\mathbf{V}_\mathbf{R}\mathbf{V}_\mathbf{R}^T)\mathbf{S}_C\mathbf{S}_C^T\mathbf{V}_\mathbf{R}\big\|_F^2, \\
g_C &= \big\|\mathbf{U}_\mathbf{C}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{U}_\mathbf{A}\mathbf{\Sigma}_\mathbf{A}^\alpha\big\|_F^2, & g_R &= \big\|\mathbf{\Sigma}_\mathbf{A}^{1-\alpha}\mathbf{V}_\mathbf{A}(\mathbf{I}_n - \mathbf{V}_\mathbf{R}\mathbf{V}_\mathbf{R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_\mathbf{R}\big\|_F^2, \\
g_C' &= \big\|\mathbf{U}_\mathbf{C}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_\mathbf{C}\mathbf{U}_\mathbf{C}^T)\mathbf{U}_\mathbf{A}\mathbf{\Sigma}_\mathbf{A}^\alpha\big\|_2^2, & g_R' &= \big\|\mathbf{\Sigma}_\mathbf{A}^{1-\alpha}\mathbf{V}_\mathbf{A}(\mathbf{I}_n - \mathbf{V}_\mathbf{R}\mathbf{V}_\mathbf{R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_\mathbf{R}\big\|_2^2.
\end{aligned}$$

Proof Let $k_c = \mathrm{rank}(\mathbf{C}) \leq c$ and $k_r = \mathrm{rank}(\mathbf{R}) \leq r$. Let $\mathbf{U}_\mathbf{C} \in \mathbb{R}^{m\times k_c}$ be the left singular vectors of $\mathbf{C}$ and $\mathbf{V}_\mathbf{R} \in \mathbb{R}^{n\times k_r}$ be the right singular vectors of $\mathbf{R}$. Define $\mathbf{Z}^\star, \tilde{\mathbf{Z}} \in \mathbb{R}^{k_c\times k_r}$ by

$$\mathbf{Z}^\star = \mathbf{U}_\mathbf{C}^T\mathbf{A}\mathbf{V}_\mathbf{R}, \qquad \tilde{\mathbf{Z}} = (\mathbf{S}_C^T\mathbf{U}_\mathbf{C})^\dagger(\mathbf{S}_C^T\mathbf{A}\mathbf{S}_R)(\mathbf{V}_\mathbf{R}^T\mathbf{S}_R)^\dagger.$$

We have that $\mathbf{C}\mathbf{U}^{\star}\mathbf{R} = \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}\mathbf{R}^{\dagger}\mathbf{R} = \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}\mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T} = \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}$. By definition, it holds that that

$$\begin{aligned}
\tilde{\mathbf{U}} &= (\mathbf{S}_{C}^{T}\mathbf{C})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{R}\mathbf{S}_{R})^{\dagger} \\
&= (\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}}\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{\dagger} \\
&= (\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{\dagger}(\mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}})^{\dagger} \\
&= (\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T})^{\dagger}\tilde{\mathbf{Z}}(\mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}})^{\dagger},
\end{aligned}$$

where the third equality follows from that $\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}}$ and $\mathbf{S}_{R}^{T}\mathbf{V}_{\mathbf{R}}$ have full column rank and that $\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T}$ and $\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R}$ have full row rank. It follows that

$$\mathbf{C}\tilde{\mathbf{U}}\mathbf{R} \;=\; \mathbf{U}_{\mathbf{C}}\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T}(\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^{T})^{\dagger}\tilde{\mathbf{Z}}(\mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}})^{\dagger}\mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T} \;=\; \mathbf{U}_{\mathbf{C}}\tilde{\mathbf{Z}}\mathbf{V}_{\mathbf{R}}^{T}.$$

Since $\mathbf{C}\mathbf{U}^{\star}\mathbf{R} = \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}$ and $\mathbf{C}\tilde{\mathbf{U}}\mathbf{R} = \mathbf{U}_{\mathbf{C}}\tilde{\mathbf{Z}}\mathbf{V}_{\mathbf{R}}^{T}$, it suffices to prove the two inequalities:

$$\begin{aligned}
\|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\tilde{\mathbf{Z}}\mathbf{V}_{\mathbf{R}}^{T}\|_{F}^{2} &\leq \|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}\|_{F}^{2} + \Big(f_{R}\sqrt{h_{R}} + f_{C}\sqrt{h_{C}} + f_{C}f_{R}\sqrt{g_{C}g_{R}'}\Big)^{2}, \\
\|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\tilde{\mathbf{Z}}\mathbf{V}_{\mathbf{R}}^{T}\|_{F}^{2} &\leq \|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}\|_{F}^{2} + \Big(f_{R}\sqrt{h_{R}} + f_{C}\sqrt{h_{C}} + f_{C}f_{R}\sqrt{g_{C}'g_{R}}\Big)^{2}. \qquad (17)
\end{aligned}$$

The left-hand side can be expressed as

$$\begin{aligned}
&\big\|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\tilde{\mathbf{Z}}\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \;=\; \big\|(\mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}) + \mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \\
&= \big\|(\mathbf{I}_{m} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T})\mathbf{A} + \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}(\mathbf{I}_{n} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T}) + \mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \\
&= \big\|(\mathbf{I}_{m} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T})\mathbf{A}\big\|_{F}^{2} + \big\|\mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}(\mathbf{I}_{n} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T}) + \mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \\
&= \big\|(\mathbf{I}_{m} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T})\mathbf{A}\big\|_{F}^{2} + \big\|\mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}(\mathbf{I}_{n} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T})\big\|_{F}^{2} + \big\|\mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \\
&= \big\|(\mathbf{I}_{m} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T})\mathbf{A} + \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}(\mathbf{I}_{n} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T})\big\|_{F}^{2} + \big\|\mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} \\
&= \big\|\mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^{T}\mathbf{A}\mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2} + \big\|\mathbf{U}_{\mathbf{C}}(\mathbf{Z}^{\star} - \tilde{\mathbf{Z}})\mathbf{V}_{\mathbf{R}}^{T}\big\|_{F}^{2}.
\end{aligned}$$

From (17) we can see that it suffices to prove the two inequalities:

$$\begin{aligned}
\big\|\mathbf{Z}^{\star} - \tilde{\mathbf{Z}}\big\|_{F} &\leq f_{R}\sqrt{h_{R}} + f_{C}\sqrt{h_{C}} + f_{C}f_{R}\sqrt{g_{C}g_{R}'}, \\
\big\|\mathbf{Z}^{\star} - \tilde{\mathbf{Z}}\big\|_{F} &\leq f_{R}\sqrt{h_{R}} + f_{C}\sqrt{h_{C}} + f_{C}f_{R}\sqrt{g_{C}'g_{R}}. \qquad (18)
\end{aligned}$$

We left multiply both sides of $\tilde{\mathbf{Z}} = (\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{\dagger}$ by $(\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{T}(\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})$ and right multiply by $(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{T}$. We obtain

$$\begin{aligned}
&(\mathbf{U}_{\mathbf{C}}^{T}\mathbf{S}_{C}\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})\tilde{\mathbf{Z}}(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R}\mathbf{S}_{R}^{T}\mathbf{V}_{\mathbf{R}}) \\
&= (\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{T}(\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})(\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{\dagger}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{\dagger}(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{T} \\
&= (\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})^{T}(\mathbf{S}_{C}^{T}\mathbf{A}\mathbf{S}_{R})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R})^{T} \\
&= \mathbf{U}_{\mathbf{C}}^{T}\mathbf{S}_{C}\mathbf{S}_{C}^{T}(\mathbf{A}^{\perp} + \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T})\mathbf{S}_{R}\mathbf{S}_{R}^{T}\mathbf{V}_{\mathbf{R}}.
\end{aligned}$$

Here the second equality follows from that $\mathbf{Y}^{T}\mathbf{Y}\mathbf{Y}^{\dagger} = \mathbf{Y}^{T}$ and $\mathbf{Y}^{\dagger}\mathbf{Y}\mathbf{Y}^{T} = \mathbf{Y}^{T}$ for any $\mathbf{Y}$, and the last equality follows by defining $\mathbf{A}^{\perp} = \mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{Z}^{\star}\mathbf{V}_{\mathbf{R}}^{T}$. It follows that

$$(\mathbf{U}_{\mathbf{C}}^{T}\mathbf{S}_{C}\mathbf{S}_{C}^{T}\mathbf{U}_{\mathbf{C}})(\tilde{\mathbf{Z}} - \mathbf{Z}^{\star})(\mathbf{V}_{\mathbf{R}}^{T}\mathbf{S}_{R}\mathbf{S}_{R}^{T}\mathbf{V}_{\mathbf{R}}) \;=\; \mathbf{U}_{\mathbf{C}}^{T}\mathbf{S}_{C}\mathbf{S}_{C}^{T}\mathbf{A}^{\perp}\mathbf{S}_{R}\mathbf{S}_{R}^{T}\mathbf{V}_{\mathbf{R}}.$$

We decompose $\mathbf{A}^{\perp}$ by

$$\begin{aligned}\mathbf{A}^{\perp} \;&=\; \mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\mathbf{A} + \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\mathbf{A} - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\mathbf{A}\mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T \\ &=\; \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T) + (\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}\mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T + (\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T).\end{aligned}$$

It follows that

$$\begin{aligned}&(\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}})(\tilde{\mathbf{Z}} - \mathbf{Z}^{\star})(\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}}) \\ &=\; \mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}} \\ &\quad + \mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}\mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}} \\ &\quad + \mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}},\end{aligned}$$

and thus

$$\begin{aligned}\tilde{\mathbf{Z}} - \mathbf{Z}^{\star} &= \mathbf{U}_{\mathbf{C}}^T\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}}(\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}})^{-1} \\ &\quad + (\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}})^{-1}\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}\mathbf{V}_{\mathbf{R}} \\ &\quad + (\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}})^{-1}\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}(\mathbf{I} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}}(\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}})^{-1}.\end{aligned}$$

It follows that

$$\begin{aligned}\|\tilde{\mathbf{Z}} - \mathbf{Z}^{\star}\|_F \;&\leq\; \sigma_{\min}^{-1}(\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}})\big\|\mathbf{A}(\mathbf{I}_n - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}}\big\|_F \\ &\quad + \sigma_{\min}^{-1}(\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}})\big\|\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}\mathbf{V}_{\mathbf{R}}\big\|_F \\ &\quad + \sigma_{\min}^{-1}(\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}})\sigma_{\min}^{-1}(\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}})\big\|\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T(\mathbf{I}_m - \mathbf{U}_{\mathbf{C}}\mathbf{U}_{\mathbf{C}}^T)\mathbf{A}(\mathbf{I} - \mathbf{V}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}}\big\|_F.\end{aligned}$$

This proves (18) and thereby concludes the proof. ■

H.2 Proof of the Theorem

Assumption 3 assumes that the sketching matrices $\mathbf{S}_C$ and $\mathbf{S}_R$ satisfy the first two approximate matrix multiplication properties. Under the assumption, we obtain Lemma 25, which shows that $\tilde{\mathbf{U}}$ is nearly as good as $\mathbf{U}^{\star}$ in terms of objective function value.

Assumption 3 *Let $\mathbf{B}$ be any fixed matrix. Let $\mathbf{C} \in \mathbb{R}^{m \times c}$ and $\mathbf{C} = \mathbf{U}_{\mathbf{C}}\mathbf{\Sigma}_{\mathbf{C}}\mathbf{V}_{\mathbf{C}}^T$ be the SVD. Assume that a certain sketching matrix $\mathbf{S}_C \in \mathbb{R}^{m \times s_c}$ satisfies*

$$\begin{aligned}\mathbb{P}\Big\{\big\|\mathbf{U}_{\mathbf{C}}\mathbf{S}_C\mathbf{S}_C^T\mathbf{U}_{\mathbf{C}} - \mathbf{I}\big\|_2 \;\geq\; \frac{1}{10}\Big\} \;&\leq\; \delta_1 \\ \mathbb{P}\Big\{\big\|\mathbf{U}_{\mathbf{C}}^T\mathbf{S}_C\mathbf{S}_C^T\mathbf{B} - \mathbf{U}_{\mathbf{C}}^T\mathbf{B}\big\|_F^2 \;\geq\; \epsilon\|\mathbf{B}\|_F^2\Big\} \;&\leq\; \delta_2\end{aligned}$$

for any $\delta_1, \delta_2 \in (0, 0.2)$. Let $\mathbf{R} \in \mathbb{R}^{r \times n}$ and $\mathbf{R} = \mathbf{U}_{\mathbf{R}}\mathbf{\Sigma}_{\mathbf{R}}\mathbf{V}_{\mathbf{R}}^T$ be the SVD. Similarly, assume $\mathbf{S}_R \in \mathbb{R}^{n \times s_r}$ satisfies

$$\begin{aligned}\mathbb{P}\Big\{\big\|\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{V}_{\mathbf{R}} - \mathbf{I}\big\|_2 \;\geq\; \frac{1}{10}\Big\} \;&\leq\; \delta_1 \\ \mathbb{P}\Big\{\big\|\mathbf{V}_{\mathbf{R}}^T\mathbf{S}_R\mathbf{S}_R^T\mathbf{B} - \mathbf{V}_{\mathbf{R}}^T\mathbf{B}\big\|_F^2 \;\geq\; \epsilon\|\mathbf{B}\|_F^2\Big\} \;&\leq\; \delta_2.\end{aligned}$$

Lemma 25 *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{C} \in \mathbb{R}^{m\times c}$, and $\mathbf{R} \in \mathbb{R}^{r\times n}$ be any fixed matrices. Let $\mathbf{U}^\star$ and $\tilde{\mathbf{U}}$ be defined in* (8) *and* (9)*, respectively. Let $k_c = \mathrm{rank}(\mathbf{C})$, $k_r = \mathrm{rank}(\mathbf{R})$, $q = \min\{m, n\}$, and $\epsilon \in (0, 1)$ be the error parameter. Assume that the sketching matrices $\mathbf{S}_C$ and $\mathbf{S}_R$ satisfy Assumption 3 and that $\epsilon^{-1} = o(q)$. Then*

$$\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 \;\; \leq \;\; (1 + 4\epsilon^2 q)\, \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2$$

holds with probability at least $1 - 2\delta_1 - 3\delta_2$.

Proof Let f_C, f_R, h_C, h_R, g_C, g_R, g'_C, g'_R be defined Lemma 24. Under Assumption 3, we have that

$$\begin{aligned} f_C \leq \frac{10}{9}, \qquad & h_C \leq \epsilon \|\mathbf{A} - \mathbf{U_C}\mathbf{U_C}^T\mathbf{A}\|_F^2 \leq \epsilon\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2, \\ f_R \leq \frac{10}{9}, \qquad & h_R \leq \epsilon \|\mathbf{A} - \mathbf{A}\mathbf{V_R}\mathbf{V_R}^T\|_F^2 \leq \epsilon\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2, \end{aligned}$$

hold simultaneously with probability at least $1 - 2\delta_1 - 2\delta_2$.

We fix $\alpha = 1$, then $g_C = h_C$, and $g'_R \leq \big\|(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V_R}\big\|_2^2$. Under Assumption 3, we have that

$$\begin{aligned} \sqrt{g'_R} \;\; & \leq \;\; \big\|(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V_R} - (\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{V_R}\big\|_F \\ & \leq \;\; \sqrt{\epsilon}\big\|(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\big\|_F \;\leq\; \sqrt{\epsilon n} \end{aligned}$$

holds with probability at least $1 - \delta_2$. It follows from Lemma 24 that

$$\begin{aligned} & \|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 - \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 \\ \leq \;\; & \Big(f_R\sqrt{h_R} + f_C\sqrt{h_C} + f_C f_R\sqrt{g_C g'_R}\Big)^2 \\ \leq \;\; & \Big(\frac{20}{9}\sqrt{\epsilon}\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F + \frac{10^2}{9^2}\epsilon\sqrt{n}\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F\Big)^2 \\ = \;\; & \frac{10^4}{9^4}\epsilon^2 n\big(1 + o(1)\big)\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2 \;\leq\; 4\epsilon^2 n\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2 \end{aligned}$$

holds with probability at least $1-2\delta_1-3\delta_2$. Here the equality follows from that $\epsilon^{-1} = o(n)$.

Alternatively, if we fix $\alpha = 0$, we will obtain that

$$\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 \;\leq\; \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 + 4\epsilon^2 m\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2$$

with probability $1 - 2\delta_1 - 3\delta_2$. Therefore, if $n \leq m$, we fix $\alpha = 1$; otherwise we fix $\alpha = 0$. This concludes the proof. ■

In the following we further assume that the sketching matrices $\mathbf{S}_C$ and $\mathbf{S}_R$ satisfy the third approximate matrix multiplication property. Under Assumption 3 and Assumption 4, we obtain Lemma 26 which is stronger than Lemma 25.

Assumption 4 *Let* $\mathbf{B}$ *be any fixed matrix. Let* $\mathbf{C} \in \mathbb{R}^{m\times c}$*,* $k_c = \mathrm{rank}(\mathbf{C})$*, and* $\mathbf{C} = \mathbf{U_C}\mathbf{\Sigma_C}\mathbf{V_C}^T$ *be the SVD. Assume that a certain sketching matrix* $\mathbf{S}_C \in \mathbb{R}^{n\times s_c}$ *satisfies*

$$\mathbb{P}\Big\{ \big\|\mathbf{U_C}^T \mathbf{S}_C \mathbf{S}_C^T \mathbf{B} - \mathbf{U_C}^T \mathbf{B}\big\|_2^2 \;\geq\; \epsilon \|\mathbf{B}\|_2^2 + \frac{\epsilon}{k_c}\|\mathbf{B}\|_F^2 \Big\} \;\leq\; \delta_3$$

for any $\epsilon \in (0,1)$ *and* $\delta_3 \in (0, 0.2)$*. Let* $\mathbf{R} \in \mathbb{R}^{r\times n}$*,* $k_r = \mathrm{rank}(\mathbf{R})$*, and* $\mathbf{R} = \mathbf{U_R}\mathbf{\Sigma_R}\mathbf{V_R}^T$ *be the SVD. Similarly, assume that* $\mathbf{S}_R \in \mathbb{R}^{n\times s_r}$ *satisfies*

$$\mathbb{P}\Big\{ \big\|\mathbf{V_R}^T \mathbf{S}_R \mathbf{S}_R^T \mathbf{B} - \mathbf{V_R}^T \mathbf{B}\big\|_2^2 \;\geq\; \epsilon \|\mathbf{B}\|_2^2 + \frac{\epsilon}{k_r}\|\mathbf{B}\|_F^2 \Big\} \;\leq\; \delta_3 .$$

Lemma 26 *Let* $\mathbf{A} \in \mathbb{R}^{m\times n}$*,* $\mathbf{C}$*,* $\mathbf{R}$*,* $\mathbf{U}^\star$*,* $\tilde{\mathbf{U}}$*,* k_c*,* k_r *be defined in Lemma 25. Let* $q = \min\{m,n\}$ *and* $\tilde{q} = \min\{m/k_c, n/k_r\}$*. Assume that the sketching matrices* $\mathbf{S}_C$ *and* $\mathbf{S}_R$ *satisfy Assumption 3 and Assumption 4 and that* $\epsilon^{-1} = o\big(\tilde{q}\big)$*. Then*

$$\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{X}}\mathbf{R}\|_F^2 \quad\leq\quad (1 + 4\epsilon^2 \tilde{q})\, \|\mathbf{A} - \mathbf{C}\mathbf{X}^\star\mathbf{R}\|_F^2$$

holds with probability at least $1 - 2\delta_1 - 2\delta_2 - \delta_3$*.*

Proof Let f_C, f_R, h_C, h_R, g_C, g_R, g'_C, g'_R be defined Lemma 24. Under Assumption 3, we have shown in the proof of Lemma 25 that

$$\begin{aligned} f_C \leq \frac{10}{9}, &\qquad h_C \leq \epsilon \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2, \\ f_R \leq \frac{10}{9}, &\qquad h_R \leq \epsilon \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2, \end{aligned}$$

hold simultaneously with probability at least $1 - 2\delta_1 - 2\delta_2$.

We fix $\alpha = 1$, then $g_C = h_C$, and $g'_R \leq \big\|(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V_R}\big\|_2^2$. Under Assumption 4, we have that

$$\begin{aligned} g'_R &\leq \big\|(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{S}_R\mathbf{S}_R^T\mathbf{V_R} - \underbrace{(\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T)\mathbf{V_R}}_{=\mathbf{0}}\big\|_2^2 \\ &\leq \epsilon\big\|\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T\big\|_2^2 + \frac{\epsilon}{k_r}\big\|\mathbf{I}_n - \mathbf{V_R}\mathbf{V_R}^T\big\|_F^2 \;\leq\; \epsilon + \frac{\epsilon(n-k_r)}{k_r} \;=\; \frac{\epsilon n}{k_r} \end{aligned}$$

holds with probability at least $1 - \delta_3$. It follows from Lemma 24 that

$$\begin{aligned} & \|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 - \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 \\ \leq\;& \Big(f_R\sqrt{h_R} + f_C\sqrt{h_C} + f_C f_R \sqrt{g_C g'_R}\Big)^2 \\ \leq\;& \Big(\frac{20}{9}\sqrt{\epsilon}\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F + \frac{10^2}{9^2}\epsilon\sqrt{n/k_r}\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F\Big)^2 \\ =\;& \frac{10^4}{9^4}\epsilon^2 n k_r^{-1}\big(1 + o(1)\big)\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2 \;\leq\; 4\epsilon^2 n k_r^{-1}\|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2 \end{aligned}$$

holds with probability at least $1 - 2\delta_1 - 2\delta_2 - \delta_3$. Here the equality follows from that $\epsilon^{-1} = o(n/k_r)$.

Analogously, by fixing $\alpha = 0$ and assuming $\epsilon^{-1} = o(m/k_c)$, we can show that

$$\|\mathbf{A} - \mathbf{C}\tilde{\mathbf{U}}\mathbf{R}\|_F^2 - \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}\|_F^2 \;\leq\; 4\epsilon^2 m k_c^{-1} \|\mathbf{A} - \mathbf{C}\mathbf{U}^\star\mathbf{R}^T\|_F^2$$

holds with probability at least $1 - 2\delta_1 - 2\delta_2 - \delta_3$. This concludes the proof. ■

Finally, we prove Theorem 9 using Lemma 25 and Lemma 26.

For leverage score sampling, uniform sampling, and count sketch, Assumption 3 is satisfied. Then the bound follows by setting $\epsilon = 0.5\sqrt{\epsilon'/q}$ and applying Lemma 25. Here $q = \min\{m, n\}$. For the three sketching methods, we set $\delta_1 = 0.01$ and $\delta_2 = 0.093$.

For Gaussian projection and SRHT, Assumption 3 and Assumption 4 are satisfied. Then the bound follows by setting $\epsilon = 0.5\sqrt{\epsilon'/\tilde{q}}$ and applying Lemma 26. Here $\tilde{q} = \min\{m/k_c, n/k_r\}$. For Gaussian projection, we set $\delta_1 = 0.01$, $\delta_2 = 0.09$, and $\delta_3 = 0.1$. For SRHT, we set $\delta_1 = 0.02$, $\delta_2 = 0.08$, and $\delta_3 = 0.1$.

References

Adi Ben-Israel and Thomas N.E. Greville. *Generalized Inverses: Theory and Applications. Second Edition.* Springer, 2003.

Christos Boutsidis and David P. Woodruff. Optimal CUR matrix decompositions. *arXiv preprint arXiv:1405.7910*, 2014.

Christos Boutsidis, Petros Drineas, and Malik Magdon-Ismail. Near-optimal column-based matrix reconstruction. *SIAM Journal on Computing*, 43(2):687–717, 2014.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. *Theoretical Computer Science*, 312(1):3–15, 2004.

Kenneth L. Clarkson and David P. Woodruff. Low rank approximation and regression in input sparsity time. In *Annual ACM Symposium on theory of computing (STOC)*. ACM, 2013.

Michael B Cohen, Jelani Nelson, and David P Woodruff. Optimal approximate matrix product in terms of stable rank. *arXiv preprint arXiv:1507.02268*, 2015.

Bo Dai, Bo Xie, Niao He, Yingyu Liang, Anant Raj, Maria-Florina F Balcan, and Le Song. Scalable kernel methods via doubly stochastic gradients. In *Neural Information Processing Systems (NIPS)*. 2014.

Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *the 17th Annual ACM-SIAM Symposium On Discrete Algorithm (SODA)*, 2006.

Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, September 2008.

Petros Drineas, Michael W. Mahoney, S. Muthukrishnan, and Tamás Sarlós. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, 2011.

Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the Nyström method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.

Alex Gittens. The spectral norm error of the naive Nyström extension. *arXiv preprint arXiv:1110.5305*, 2011.

Alex Gittens and Michael W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. *Journal of Machine Learning Research*, 17(117):1–65, 2016.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

Cho-Jui Hsieh, Si Si, and Inderjit S Dhillon. Fast prediction for large-scale kernel machines. In *Neural Information Processing Systems (NIPS)*. 2014.

William B. Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26(189-206), 1984.

Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. On sampling-based approximate spectral decomposition. In *International Conference on Machine Learning (ICML)*, 2009.

Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. Sampling methods for the Nyström method. *Journal of Machine Learning Research*, 13:981–1006, 2012.

Yichao Lu, Paramveer Dhillon, Dean P Foster, and Lyle Ungar. Faster ridge regression via the subsampled randomized Hadamard transform. In *Neural Information Processing Systems (NIPS)*, 2013.

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

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

Xiangrui Meng and Michael W Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *the 45th Annual ACM Symposium on Theory Of Computing (STOC)*, 2013.

John Nelson and Huy L Nguyên. OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, 2013.

Rob Patro and Carl Kingsford. Global network alignment using multiscale spectral signatures. *Bioinformatics*, 28(23):3105–3114, 2012.

Ninh Pham and Rasmus Pagh. Fast and scalable polynomial kernels via explicit feature maps. In *the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2013.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Neural Information Processing Systems (NIPS)*, pages 1177–1184, 2007.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, 2002.

Donghyuk Shin, Si Si, and Inderjit S Dhillon. Multi-scale link prediction. In *International Conference on Information and Knowledge Management (CIKM)*. ACM, 2012.

Si Si, Cho-Jui Hsieh, and Inderjit Dhillon. Memory efficient kernel approximation. In *International Conference on Machine Learning (ICML)*, pages 701–709, 2014a.

Si Si, Donghyuk Shin, Inderjit S Dhillon, and Beresford N Parlett. Multi-scale spectral decomposition of massive graphs. In *Neural Information Processing Systems (NIPS)*. 2014b.

G. W. Stewart. Four algorithms for the efficient computation of truncated pivoted QR approximations to a sparse matrix. *Numerische Mathematik*, 83(2):313–323, 1999.

Ameet Talwalkar and Afshin Rostamizadeh. Matrix coherence and the Nyström method. *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010.

Mikkel Thorup and Yin Zhang. Tabulation-based 5-independent hashing with applications to linear probing and second moment estimation. *SIAM Journal on Computing.*, 41(2): 293–331, April 2012. ISSN 0097-5397.

Joel A Tropp. Improved analysis of the subsampled randomized hadamard transform. *Advances in Adaptive Data Analysis*, 3(01n02):115–126, 2011.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Shusen Wang and Zhihua Zhang. Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling. *Journal of Machine Learning Research*, 14:2729–2769, 2013.

Shusen Wang, Chao Zhang, Hui Qian, and Zhihua Zhang. Improving the modified Nystrom method using spectral shifting. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2014.

Shusen Wang, Luo Luo, and Zhihua Zhang. SPSD matrix approximation vis column selection: theories, algorithms, and extensions. *Journal of Machine Learning Research*, 17(49):1–49, 2016.

Kilian Weinberger, Anirban Dasgupta, John Langford, Alex Smola, and Josh Attenberg. Feature hashing for large scale multitask learning. In *International Conference on Machine Learning (ICML)*, 2009.

Christopher Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Neural Information Processing Systems (NIPS)*, 2001.

David P Woodruff. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.

Tianbao Yang, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou. Nyström method vs random fourier features: A theoretical and empirical comparison. In *Neural Information Processing Systems (NIPS)*, 2012.

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