SIAM J. MATRIX ANAL. APPL.
Vol. 43, No. 2, pp. 1003–1028

FAST DETERMINISTIC APPROXIMATION OF SYMMETRIC INDEFINITE KERNEL MATRICES WITH HIGH DIMENSIONAL DATASETS*

DIFENG CAI†, JAMES NAGY†, AND YUANZHE XI†

Abstract. Kernel methods are used frequently in various applications of machine learning. For large-scale high dimensional applications, the success of kernel methods hinges on the ability to operate certain large dense kernel matrix K. An enormous amount of literature has been devoted to the study of symmetric positive semidefinite (SPSD) kernels, where Nyström methods compute a low-rank approximation to the kernel matrix via choosing landmark points. In this paper, we study the Nyström method for approximating both symmetric *indefinite* kernel matrices as well SPSD ones. We first develop a theoretical framework for general symmetric kernel matrices, which provides a theoretical guidance for the selection of landmark points. We then leverage discrepancy theory to propose the *anchor net method* for computing accurate Nyström approximations with optimal complexity. The anchor net method operates entirely on the dataset without requiring the access to K or its matrix-vector product. Results on various types of kernels (both indefinite and SPSD ones) and machine learning datasets demonstrate that the new method achieves better accuracy and stability with lower computational cost compared to the state-of-the-art Nyström methods.

Key words. indefinite kernel, low-rank approximation, error analysis, high dimensional data

MSC codes. 15A23, 68W25, 11K38, 65D99

DOI. 10.1137/21M1424627

1. Introduction.

Kernel methods provide a powerful tool for solving nonlinear problems in data science and are used in various machine learning tools such as support vector machines (SVMs), kernel ridge regression, spectral clustering, and Gaussian processes (cf. [7]). Given n data points $x_1, \ldots, x_n \in \mathbb{R}^d$ and a kernel function $\kappa(\cdot, \cdot)$, kernel methods form an $n \times n$ kernel matrix $K_{i,j} = \kappa(x_i, x_j)$ to implicitly map data to a kernel feature space, where the originally nonlinear relationship between categories can be transformed into a linear one. The kernel function κ is often taken to be symmetric positive semidefinite (SPSD) in the literature [43, 9]. Recently, methods based on indefinite kernel functions such as the jittering kernel [10], Kullback–Leibler divergence kernel [30], tangent distance kernel [19], and multiquadric kernel [16] have also been developed. In addition, indefinite kernel matrices also occur as the derivatives of SPSD kernels in solving optimization problems (see, for example, [2]). Theoretical justifications for the SVMs associated with indefinite kernels can be found in [36, 18].

Due to the need to assemble and operate dense kernel matrices K, kernel methods are often quoted to scale at least $O(n^2)$. A popular approach to circumvent this computational bottleneck is to work with a low-rank approximation to K. Nyström methods are widely used to derive low-rank approximations to SPSD kernel matrices that arise frequently in SVMs and other applications. The method first generates a

*Received by the editors June 3, 2021; accepted for publication (in revised form) by S. Le Borne February 23, 2022; published electronically June 28, 2022.
https://doi.org/10.1137/21M1424627
Funding: The work of the first and third authors was supported by the National Science Foundation (NSF) grant OAC 2003720. The work of the second author was supported by the NSF grant DMS-1819042.
†Department of Mathematics, Emory University, Atlanta, GA 30322 USA (dcai7@emory.edu, jnagy@emory.edu, yxi26@emory.edu).

small subset of points S, known as *landmark points*, and then computes a low-rank approximation of the following form:

$$K \approx K_{XS} K_{SS}^{+} K_{SX}, \tag{1.1}$$

where we use K_{IJ} to denote the matrix with entries given by $\kappa(x, y)$ for $x \in I \subset \mathbb{R}^d, y \in J \subset \mathbb{R}^d$ and K_{SS}^{+} denotes the pseudoinverse of K_{SS}. Different ways of choosing S yield different variants of the Nyström method. The original Nyström method in [44] selects S via a uniform sampling over the dataset and is often called the *uniform* Nyström method. Later developments for generating S include nonuniform sampling techniques such as ridge leverage score–based sampling [27, 1, 17, 32, 39] and determinantal point processes [25, 15, 14], the k-means clustering–based method [46, 45], the randomized projection method [33], etc. Since the choice of S dictates the approximation accuracy, a fundamental question for Nyström method is the following:

- **Question 1.** Given a dataset X, what kind of subset S yields a good Nyström approximation?

For SPSD kernels, a probabilistic interpretation is given by ridge leverage score. For general possibly *indefinite* kernels, from a computational point of view, it is more desirable to have a straightforward geometric understanding of good landmark points S, which will facilitate the fast computation of S in $O(n)$ complexity.

Despite the lack of discussion, the query for applying Nyström approximation in (1.1) to *indefinite* kernel matrices is quite natural because, mathematically, (1.1) does *not* require K to be SPSD. Thus it is natural to ask the following questions:

- **Question 2.** Does Nyström approximation (1.1) apply to *indefinite* kernel matrices?
- **Question 3.** For a symmetric (possibly indefinite) kernel matrix, how should one choose S in $O(n)$ complexity to obtain an accurate Nyström approximation?

Note that ridge leverage score is only defined for a SPSD kernel matrix and a different low-rank approximation method called the random kitchen sinks method (or random Fourier features method) [37, 38, 28] not only requires the kernel to be SPSD but also shift-invariant. Hence those methods can *not* be directly applied to *indefinite* kernels.

The questions above motivate the work in this paper, and the main contributions of the paper are summarized below.

1. **Theoretical guidance for landmark point selection**. To guide the choice of landmark points, we present a new framework to analyze the Nyström approximation error in the general setting, where the kernel matrix can be *indefinite*. The new error estimate takes the following form and is independent of the underlying scheme to select S:

$$||K - K_{XS} K_{SS}^{+} K_{SX}||_{\max} \leq \epsilon_1 + 2\epsilon_2 + C_S \epsilon_2^2, \tag{1.2}$$

   where ϵ_1, ϵ_2 measure certain deviation between S and X, $C_S = ||K_{SS}^{+}||_2$, and $||\cdot||_{\max}$ denotes the max norm, e.g., $\|A\|_{\max} := \max_{i,j} |A_{i,j}|$. A geometric interpretation of ϵ_1 and ϵ_2 suggests that landmark points should spread evenly in the dataset in order to achieve small approximation error.
2. **Optimal complexity for general symmetric kernels**. Based on the analysis, we leverage discrepancy theory and propose an efficient deterministic Nyström method for arbitrary symmetric (possibly *indefinite*) kernels. The proposed method scales $O(dmn)$ for selecting m landmarkpoints in a dataset

of n points in $\mathbb{R}^d$ and forming the associated low-rank factors K_{XS} and K_{SS}. This process is highly parallelizable and does *not* require any access to the kernel matrix or its matrix-vector products.

3. **Improved efficiency, approximation accuracy, and stability**. Comprehensive experiments have been performed to show that the proposed method outperforms several state-of-the-art methods for various kinds of kernels on both synthetic and real datasets when the same rank is used. We also show that the choice of S significantly affects the numerical stability of the resulting Nyström approximation and numerical regularization or stabilization techniques *cannot* fully resolve the stability issue.

The rest of the paper is organized as follows. Section 2 reviews existing Nyström methods, and section 3 presents a general error analysis for Nyström approximations to guide the selection of landmark points, valid for both *indefinite* kernels and SPSD ones. Section 4 introduces the *anchor net method* for computing Nyström approximation in linear complexity. Extensive experiments are provided in section 5, and concluding remarks are drawn in section 6. In the remaining sections, the following notations will be used throughout the paper:

- $|x-y|$ denotes the Euclidean distance between $x, y \in \mathbb{R}^d$;
- $\|\cdot\|$ denotes the 2-norm of a vector or a matrix;
- $\|\cdot\|_{\max}$ denotes the max norm of a matrix, i.e., $\|A\|_{\max} := \max_{i,j} |A_{i,j}|$;
- $\mathrm{dist}_\infty(\cdot,\cdot)$ denotes the distance function in l^∞-norm;
- $\lambda(\Omega)$ denotes the Lebesgue measure of a bounded measurable set Ω in $\mathbb{R}^d$.

2. General Nyström method: SPSD and indefinite cases. Given an input dataset $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ and a symmetric (possibly indefinite) kernel function $\kappa(x, y)$, the corresponding kernel matrix is defined by $K = [\kappa(x_i, x_j)]_{i,j=1}^n$. For kernel functions supported on the entire domain of definition, such as $\kappa(x, y) = e^{-\|x-y\|^2}, \tanh(x \cdot y + 1)$, the corresponding kernel matrix K is dense, and the corresponding cost for storing the matrix or applying it to a vector is $O(n^2)$.

The Nyström method was proposed in [44] to reduce the quadratic cost by computing an approximate low-rank factorization in the form $K \approx K_{XS} K_{SS}^+ K_{SX}$, where the size of S is significantly smaller than n. Different variants of the Nyström method use different methods to compute the landmark points S. Some methods require K to be SPSD, including nonuniform sampling–based approaches like leverage score sampling, determinantal point processes, etc., while others like uniform sampling and k-means clustering can also be potentially applied to *indefinite* kernel matrices. We review below some popular Nyström methods.

The original Nyström method in [44], known as the *uniform* Nyström method, selects landmark points via a uniform sampling over X (or, equivalently, over the index set from 1 to n). Since then, a variety of schemes have been developed to select landmark points. See, for example, [46, 26, 45, 27, 1, 17, 32, 39]. Computationally, the uniform Nyström method is the most efficient one, since it does *not* require any access to the kernel matrix or its matrix-vector product, and is not iterative. As a result, the uniform Nyström method is easy to compute and can be applied to a broad class of kernel matrices. However, the uniform Nyström method also suffers from several issues. Firstly, due to the stochastic nature, it suffers from possibly large variance [39]. Secondly, the approximation accuracy usually fails to increase consistently with the increase of the number of landmark points. Thirdly, the random choice of landmark points may lead to numerically unstable approximations. The accuracy slowdown and numerical instability will be illustrated via extensive numerical experiments.

Nonuniform sampling techniques have been developed to improve the approximation accuracy with strong theoretical guarantees [27, 1, 17, 32, 25]. These methods measure the importance of each data point with some statistical scores. A notable example is the leverage score–based sampling [29, 1, 17], including determinantal point processes [25, 14]. Each point x_i in the dataset is associated with a leverage score defined as $l_i^\gamma(K) := (K(K+\gamma I)^{-1})_{i,i}$ with $\gamma > 0$ a user-specified parameter. To generate the landmark points, each point x_i is sampled with a probability proportional to $l_i^\gamma(K)$. Since computing leverage scores involves the dense kernel matrix K and computing the matrix inverse $(K+\gamma I)^{-1}$, these methods cost at least $O(n^2)$. Recently, several iterative schemes have been proposed to accelerate its computations [32, 39]. Different from uniform sampling, those methods require K to be SPSD in order to guarantee the nonnegativeness of l_i^γ.

Another variant of Nyström methods is the k-means Nyström method [46, 45]. This method performs the k-means clustering over the dataset and chooses the cluster centers as the landmark points. Similar to uniform sampling, the k-means method does not require any access to the kernel matrix. Experiments show that it tends to be more accurate than the uniform Nyström method [46, 45] but still suffers from numerical instability.

In existing literature, discussion on the choice of good landmark points out that work for *indefinite* kernels has been lacking. In fact, the Nyström approximation $K \approx K_{XS}K_{SS}^{+}K_{SX}$ has not been investigated for *indefinite* kernels, theoretically and numerically. We will show that the Nyström approximation is well defined for *indefinite* kernels but faces much more numerical challenges such as numerical instability, as compared to SPSD kernels.

3. Error estimates for the general Nyström method. In this section, we derive error estimates for the *general* Nyström method, which are valid for all symmetric kernels, including indefinite ones. The only assumption is that the landmark points are chosen from the original dataset. The analysis reveals the inherent relation between landmark points and the quality of the corresponding Nyström approximation. It serves as the theoretical foundation of the new linear complexity method proposed in section 4.

The lemma below will be used in proving the main result in Theorem 3.2.

LEMMA 3.1. *Assume A is an n-by-n matrix and $\alpha, \hat{\alpha}, \beta, \hat{\beta}$ are n-by-1 vectors. Define $\epsilon_1 := \|\hat{\alpha} - \alpha\|$ and $\epsilon_2 := \|\hat{\beta} - \beta\|$. Then*

$$\left|\hat{\alpha}^T A\hat{\beta} - \alpha^T A\beta\right| \le \left\|\alpha^T A\right\| \cdot \epsilon_2 + \|A\beta\| \cdot \epsilon_1 + \|A\| \cdot \epsilon_1\epsilon_2. \tag{3.1}$$

Proof. Define $e_1 := \hat{\alpha} - \alpha$ and $e_2 := \hat{\beta} - \beta$. Then (3.1) follows from the fact that

$$\hat{\alpha}^T A\hat{\beta} - \alpha^T A\beta = \alpha^T Ae_2 + e_1^T A\beta + e_1^T Ae_2. \qquad \square$$

In the theorem below, we derive a universal error bound for the Nyström method. The kernel function is assumed to be symmetric and continuous, not necessarily positive-definite. Unlike existing error estimates, the result below is independent of the specific Nyström scheme. The only assumption is that the landmark points belong to the original dataset, which is indeed the case in all Nyström schemes except the one based on k-means clustering [46, 45].

THEOREM 3.2. *Let $\kappa(x, y)$ be a symmetric function, e.g., $\kappa(x, y) = \kappa(y, x)$. Suppose $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ and $K = K_{XX} := [\kappa(x_i, x_j)]_{i,j=1}^n$. If $S = \{z_1, \ldots, z_r\} \subset$*

X, then

$$\left\|K - K_{XS}K_{SS}^{+}K_{XS}^{T}\right\|_{\max} \le E_r + 2\hat{E}_r + \left\|K_{SS}^{+}\right\|\hat{E}_r^2, \tag{3.2}$$

where

$$E_r := \max_{x,y\in X}\min_{u,v\in S}|\kappa(x,y)-\kappa(u,v)|, \quad \hat{E}_r := \max_{x\in X}\min_{u\in S}\left(\sum_{i=1}^{r}|\kappa(x,z_i)-\kappa(u,z_i)|^2\right)^{\frac{1}{2}}. \tag{3.3}$$

Proof. Define $R = X\backslash S$. Since $S \subset X$, for some permutation matrix P, there holds

$$K - K_{XS}K_{SS}^{+}K_{XS}^{T} = P\begin{bmatrix} O & O \\ O & K_{RR} - K_{RS}K_{SS}^{+}K_{RS}^{T}\end{bmatrix}P^{T}.$$

Consequently, $\left\|K - K_{XS}K_{SS}^{+}K_{XS}^{T}\right\|_{\max} = \left\|K_{RR} - K_{RS}K_{SS}^{+}K_{RS}^{T}\right\|_{\max}$. It suffices to estimate the difference below:

$$\kappa(x,y) - K_{xS}K_{SS}^{+}K_{yS}^{T}, \tag{3.4}$$

where

$$K_{xS} := \begin{bmatrix}\kappa(x,z_1) & \cdots & \kappa(x,z_r)\end{bmatrix} \quad \text{for any } \ x \in \mathbb{R}^d.$$

Note that, for any $u, v \in S$,

$$\kappa(u,v) = K_{uS}K_{SS}^{+}K_{vS}^{T} \tag{3.5}$$

because $K_{SS}K_{SS}^{+}K_{SS} = K_{SS}$ and $K_{SS} = K_{SS}^{T}$. Define the column vectors

$$\alpha := K_{uS}^{T}, \quad \hat{\alpha} := K_{xS}^{T}, \quad \beta := K_{vS}^{T}, \quad \hat{\beta} := K_{yS}^{T} \tag{3.6}$$

and the scalars

$$\begin{aligned}\epsilon_1 &:= ||\hat{\alpha}-\alpha|| = \left(\sum_{i=1}^{r}|\kappa(x,z_i)-\kappa(u,z_i)|^2\right)^{\frac{1}{2}},\\ \epsilon_2 &:= ||\hat{\beta}-\beta|| = \left(\sum_{i=1}^{r}|\kappa(y,z_i)-\kappa(v,z_i)|^2\right)^{\frac{1}{2}}.\end{aligned} \tag{3.7}$$

We can then use (3.5) and (3.6) to rewrite (3.4) as

$$\kappa(x,y) - \hat{\alpha}^{T}K_{SS}^{+}\hat{\beta} = (\kappa(x,y)-\kappa(u,v)) + (\alpha^{T}K_{SS}^{+}\beta - \hat{\alpha}^{T}K_{SS}^{+}\hat{\beta}), \tag{3.8}$$

where $u, v \in S$ can be arbitrary.

The second part on the right-hand side of (3.8) can be estimated as follows by using Lemma 3.1:

$$\begin{aligned}\left|\hat{\alpha}^{T}K_{SS}^{+}\hat{\beta} - \alpha^{T}K_{SS}^{+}\beta\right| &\le \left\|\alpha^{T}K_{SS}^{+}\right\|\epsilon_2 + \left\|K_{SS}^{+}\beta\right\|\epsilon_1 + \left\|K_{SS}^{+}\right\|\epsilon_1\epsilon_2\\ &\le \epsilon_2 + \epsilon_1 + \left\|K_{SS}^{+}\right\|\epsilon_1\epsilon_2.\end{aligned} \tag{3.9}$$

Here, the last inequality is due to the fact that both $\alpha^{T}K_{SS}^{+} = K_{uS}K_{SS}^{+}$ and $K_{SS}^{+}\beta = K_{SS}^{+}K_{Sv}$ are the row or column of the matrix

$$K_{SS}K_{SS}^{+} = K_{SS}^{+}K_{SS} = U\begin{bmatrix} e_1 & & \\ & \ddots & \\ & & e_r\end{bmatrix}U^{T},$$

where U is an orthogonal matrix and $e_i \in \{0,1\}$. As a result, $\left\|\alpha^T K_{SS}^+\right\| \leq 1$ and $\left\|K_{SS}^+\beta\right\| \leq 1$.

Finally, (3.9) and (3.8) imply the estimate

$$\left|\kappa(x,y) - \hat{\alpha}^T K_{SS}^+ \hat{\beta}\right| \leq |\kappa(x,y) - \kappa(u,v)| + \epsilon_1 + \epsilon_2 + \left\|K_{SS}^+\right\|\epsilon_1\epsilon_2, \tag{3.10}$$

which holds for any $u, v \in S$. Minimizing the upper bound in (3.10) over all $u, v \in S$ immediately yields

$$\left|\kappa(x,y) - \hat{\alpha}^T K_{SS}^+ \hat{\beta}\right| \leq \min_{u,v\in S}|\kappa(x,y) - \kappa(u,v)| + \min_{u\in S}\epsilon_1 + \min_{v\in S}\epsilon_2 + \left\|K_{SS}^+\right\| \min_{u\in S}\epsilon_1 \cdot \min_{v\in S}\epsilon_2, \tag{3.11}$$

where ϵ_1, ϵ_2 are defined in (3.7). The proof is completed by taking a maximum of the upper bound in (3.11) over $x, y \in X$. That is,

$$\begin{aligned}\left|\kappa(x,y) - \hat{\alpha}^T K_{SS}^+ \hat{\beta}\right| &\leq \max_{x,y\in S}\min_{u,v\in S}|\kappa(x,y) - \kappa(u,v)| + \max_{x\in X}\min_{u\in S}\epsilon_1 + \max_{y\in X}\min_{v\in S}\epsilon_2 \\ &\quad + \left\|K_{SS}^+\right\| \max_{x\in X}\min_{u\in S}\epsilon_1 \cdot \max_{y\in Y}\min_{v\in S}\epsilon_2 \\ &= E_r + 2\hat{E}_r + \left\|K_{SS}^+\right\|\hat{E}_r^2. \end{aligned}$$

∎

Remark 3.3. Note that Theorem 3.2 only requires the kernel function to be symmetric. Thus the result applies to a broad class of kernels, including SPSD kernels, like Gaussian or more generally Matérn kernels, and indefinite kernels, such as multiquadrics, thin plate spline, sigmoid kernel, etc.

We call E_r and $\hat{E}_r$ in Theorem 3.2 the bivariate and univariate *kernelized marking errors*, respectively, as both quantities are measured in terms of either bivariate or univariate kernel function evaluations and indicate the overall capacity of the landmark points S to approximate the dataset X. There are two variables that affect the approximation error of the Nyström method: the number of the landmark points r and the set of landmark points S. Here we focus on how to choose landmark points S when r is fixed. In this case, both the quantities $\hat{E}_r$ and E_r can be used to investigate how different choices of landmark points S would impact the Nyström approximation. If r is viewed as a variable, then $\hat{E}_r$ may or may not grow as r increases. Consider the one dimensional toy problem, where $\kappa(x,y) = |x-y|^2$, $X = \{x_i\}_{i=0}^{10} = \{\frac{i}{10}\}$, $S_1 = \{x_{10}\}$, $S_2 = \{x_0, x_{10}\}$. Let $\hat{E}_k$ denote the quantity for $S = S_k$. Then it can be computed that $\hat{E}_1 = 1$ (achieved at $x = x_0$) and $\hat{E}_2 = \frac{1}{2\sqrt{2}} \approx 0.35$ (achieved at $x = x_5$). In this case, $\hat{E}_r$ does decay as r increases. If we further assume that the kernel function $k(x,y)$ is Lipschitz continuous, then $|k(x,y) - k(u,v)|$ will be small if (x,y) and (u,v) are close. Under this assumption, the distance between points reflects the difference between the respective kernel evaluations. Hence the set of a fixed number of landmark points S is considered good if it is able to minimize the deviation from X, namely, making $\operatorname{dist}(x,S)$ small for each point $x \in X$. A similar result has recently been conducted in [4], which shows the exponential convergence of adaptive cross approximation [5] with respect to the fill-distance of pivoting points. We also want to emphasize that the estimate (3.2) is mainly used to motivate the selection of S rather than to select the number of the points in S in order to satisfy certain approximation accuracy.

In the next corollary, we further show that the approximation error can be bounded by $\max_{x\in X}\operatorname{dist}(x,S)$ when the kernel function κ is Lipschitz continuous.

COROLLARY 3.4. *Under the assumption of Theorem* 3.2*, if* $\kappa(x,y) \in C(\mathbb{R}^d \times \mathbb{R}^d)$ *is Lipschitz continuous, i.e.,* $|\kappa(x',y') - \kappa(x,y)| \le L(|x-x'|^2 + |y-y'|^2)^{1/2}$ *with Lipschitz constant* L*, then*

$$\left\|K - K_{XS}K_{SS}^{+}K_{XS}^{T}\right\|_{\max} \le \sqrt{2}L\delta_{X,S} + 2\sqrt{r}L\delta_{X,S} + \left\|K_{SS}^{+}\right\| rL^2\delta_{X,S}^2, \tag{3.12}$$

where $\delta_{X,S} = \max_{x\in X} dist(x,S)$.

Proof. The proof relies on (3.2), and it suffices to relate E_r, $\hat{E}_r$ to $\max_{x\in X} \operatorname{dist}(x,S)$. First we estimate E_r. For each $x \in X$, choose $z_x \in S$ to be the nearest point to x. Then it follows that

$$E_r^2 \le L^2 \max_{x,y\in X}(|x-z_x|^2 + |y-z_y|^2) = 2L^2 \max_{x\in X}|x-z_x|^2 = 2L^2 \max_{x\in X}\operatorname{dist}(x,S)^2.$$

Similarly, for $\hat{E}_r$, we deduce that

$$\hat{E}_r^2 \le L^2 \max_{x\in X}\sum_{i=1}^{r}|x-z_x|^2 = L^2 \max_{x\in X} r|x-z_x|^2 = rL^2 \max_{x\in X}\operatorname{dist}(x,S)^2.$$

The two estimates and (3.2) immediately imply (3.12), and the proof is complete. ∎

It can also be seen from (3.12) that, to achieve a better approximation, landmark points are encouraged to spread over the entire dataset to capture its geometry, thus reducing $\delta_{X,S}$. Roughly speaking, this means that any point in X is not "too far" from a landmark point in S. In fact, this principle can also lead to a submatrix K_{SS} with a relatively large numerical rank in general, an improved numerical stability, and accuracy. In two and three dimensions, one way to generate evenly spaced samples is to use farthest point sampling (FPS) [13]. FPS constructs a subset S of X by first initializing S with one point and then sequentially adding to S a point in $X\backslash S$ that is farthest from S. However, in high dimensions, the method tends to sample points on the boundary of the dataset and may ignore the interior of the dataset unless the number of samples is large enough. Computationally, the sequential procedure of FPS can be quite expensive in high dimensions since each step requires solving a minimization problem over $O(n)$ points and the overall complexity is $O(m^2n)$ for generating m landmark points, which is not optimal in m. We present in the next section an efficient, fully parallelizable algorithm with linear complexity in m and n to generate the desired subset S.

Note that several existing works have analyzed low-rank approximation–associated kernel matrices based on analytic approximation of the kernel function [21, 6, 40, 42]. Although these results are independent of the positive-definiteness of the kernel function, they are restricted to low dimensions because of the curse of dimensionality associated with analytic techniques. That is, the number of terms in an analytic approximation increases *exponentially* with the dimension, and the resulting matrix approximation is *not* low-rank for high dimensional problems. In the context of integral equations, a popular method called ACA [5] serves as a column-pivoted LU factorization. Thus it is able to perform low-rank factorization for a kernel matrix with high dimensional data in linear complexity.

Remark 3.5. One may use a different norm to measure the approximation error. The set of optimal landmark points that minimize the error bound may differ, depending on the underlying norm. A detailed investigation on how the norm affects the choice of landmark points will be discussed in a forthcoming paper. The analysis in

this section aims to provide an intuitive understanding of the desired qualities of landmark points, which will then serve as a theoretical guidance for choosing landmark points.

Remark 3.6. It should be pointed out that directly minimizing the bound in Theorem 3.2 is *not* a practical way to generate S due to the high computational cost, for example, $O(n^2)$ in computing E_r in (3.3). Our goal is to design a fast algorithm (with optimal complexity) for generating landmark points with good quality. Thus the error bound is used as a theoretical guidance for designing more efficient algorithms on generating landmark points S.

4. Anchor net method. In this section, we introduce the anchor net method to facilitate the selection of landmark points. From the analysis in section 3, we see that landmark points that spread evenly in the dataset and contain no clumps are more favorable in reducing the Nystrom approximation error. If the dataset is the unit cube, then the uniform grid points satisfy the desired properties. In general, the study of uniformity is a central topic in discrepancy theory for solving high dimensional problems. The discrepancy of a given point set measures how far the distribution deviates from the uniform one. Existing work on discrepancy theory all focuses on distribution in the unit cube, while distribution in a general region has not been investigated yet, theoretically or computationally. In section 4.1, we review low discrepancy sets and give the definition of discrepancy for a *general* region instead of the unit cube. Based on low discrepancy sets, the anchor net is introduced in section 4.2, which is able to capture the geometry of the given dataset. In section 4.3, we use anchor net to design a linear complexity landmark point selection algorithm. Discussion on implementation details is provided in section 4.4.

4.1. Low discrepancy sets. We start with the concept of low discrepancy sets. Roughly speaking, a dataset with low discrepancy contains points that spread evenly in the space with almost no local accumulations. There are several kinds of discrepancies [24], and the most widely used one is the star discrepancy, as defined below.

DEFINITION 4.1. *The star discrepancy* $D_N^*(\mathcal{A})$ *of* $\mathcal{A} = \{x_1, \ldots, x_N\} \subset [0,1]^d$ *is defined by*

$$D_N^*(\mathcal{A}) := \sup_{J \in \mathcal{J}_1} |\#(\mathcal{A} \cap J)/N - \lambda(J)|,$$

where $\mathcal{J}_1$ *is the family of all boxes in* $[0,1)^d$ *of the form* $\prod_{i=1}^d [0, a_i)$ *and* $\lambda(J)$ *denotes the Lebesgue measure of* J.

Low discrepancy sets have been studied in a number of literature as a means of generating quasi-random sequences [22, 41, 31, 24]. The most widely used ones include Halton sequences [22], digital nets, and digital sequences [41, 34, 11]. They are known to have low discrepancies in the sense that

$$D_N^*(\mathcal{A}_N) = O(N^{-1}(\log N)^d),$$

where $\mathcal{A}_N$ denotes the first N terms of a Halton sequence or a digital sequence [31, 24]. Note that uniform tensor grids are also representative low discrepancy sets but are not popular in practice due to the curse of dimensionality. We present adaptive tensor grids in section 4.4 to alleviate the issue, which allows the practical use of tensor grids in high dimensions.

The above low discrepancy sets themselves are only defined for the unit cube and are inefficient in tessellating a real dataset whose "shape" may not be regular.

Therefore, we introduce what we call the *anchor net* in section 4.2, which is built upon a collection of low discrepancy sets adjusted to the structure of the dataset. Loosely speaking, anchor nets can be viewed as generalized low discrepancy sets dictated by and specific to the given dataset. In order to measure the uniformity of a dataset in a general region, we first generalize Definition 4.1 below.

DEFINITION 4.2. *Let $\mathcal{A} = \{x_1, \ldots, x_N\} \subset [0,\infty)^d$ and Ω be a bounded measurable set in $[0,\infty)^d$ such that $\lambda(\Omega) > 0$ and $\mathcal{A} \subset \Omega$. The generalized star discrepancy $D^*_{N,\Omega}(\mathcal{A})$ of $\mathcal{A}$ in Ω is defined by*

$$D^*_{N,\Omega}(\mathcal{A}) = \sup_{J\in\mathcal{J}} |\#(\mathcal{A}\cap J)/N - \lambda(\Omega\cap J)/\lambda(\Omega)|,$$

where $\mathcal{J}$ is the family of all boxes in $[0,\infty)^d$ of the form $\prod_{i=1}^d [0,a_i)$.

Note that the generalized star discrepancy $D^*_{N,\Omega}(\mathcal{A})$ coincides with the standard one $D^*_N(\mathcal{A})$ if $\mathcal{A} \subset \Omega = [0,1)^d$. Given an arbitrary dataset, finding a region Ω that contains the dataset and reflects the geometry of the data will be beneficial in generating efficient samples that can effectively minimize the approximation error. However, the perfect region is extremely challenging to find in general since the discrete dataset can be *arbitrary*. In section 4.2, we introduce *anchor nets* as a computationally efficient way for constructing such a region Ω. Anchor nets will be used in section 4.3 to facilitate the selection of landmark points with linear complexity.

4.2. Anchor nets. In this section, we present an efficient algorithm to construct the so-called *anchor net* for a given dataset. We then verify two major properties of the anchor net: it is able to capture the entire dataset, and it has low discrepancy. As discussed in section 3, good landmark points are expected to spread over the entire dataset without forming clumps. The anchor net is designed to achieve this goal by leveraging discrepancy theory [34, 11], where one tries to construct low discrepancy sequences (deterministically) in order to avoid clumps that are frequently found in pure random sequences. Low discrepancy sequences can achieve faster convergence than pure random sequences in Monte Carlo methods [35, 31, 11]. Intuitively, one can view the anchor net as the counterpart of low discrepancy sequence and uniform sampling as the counterpart of random sequence.

The anchor net can be considered as a two-level low discrepancy set. The first level is used to decompose the dataset into smaller subsets, and the second level is used to generate "anchors." The construction procedure is sketched in Algorithm 4.1. The inputs are the dataset and a net size m. Line 1 first generates a low discrepancy set $\mathcal{T}$ of a given size in the smallest box B_0 that contains X. Lines 3–6 decompose X into smaller subset G_i's. Lines 8–10 then construct a low discrepancy set $\mathcal{A}^{(i)}$ for each (nonempty) G_i. Here we choose the number of points in $\mathcal{A}^{(i)}$ to be equal to $\lceil m * \lambda(B_i)/\sum_i \lambda(B_i)\rceil$ based on the following guideline that the size of $\mathcal{A}^{(i)}$ is proportional to $\lambda(B_i)$:

$$\frac{M_i}{\sum_{i=1}^Q M_i} = \frac{\lambda(B_i)}{\sum_{i=1}^Q \lambda(B_i)}, \tag{4.1}$$

where $M_i = \#\mathcal{A}^{(i)}$. This guideline is necessary for proving the property of anchor nets in Theorem 4.5.

In lines 1 and 11, the choice of the particular low discrepancy set is determined by the user. Options include Halton sequences, digital nets, tensor grids, etc. More

Algorithm 4.1. *Anchor net construction.*

Input: Given dataset $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ with n data points, net size m
Output: Anchor net $\mathcal{A}_X$

1: Create a low discrepancy set $\mathcal{T} = \{t_1, t_2, \ldots, t_s\}$ with $s = O(m)$ points in the smallest rectangular box B_0 that contains X
2: Initialize $G_i = \{\}$ for $i = 1, 2, \ldots, s$.
3: **for** $j = 1, 2, \ldots, n$ **do**
4: Find index i such that $i = \text{argmin}_{k=1,\ldots,s} \|x_j - t_k\|_\infty$
5: Update $G_i = G_i \cup \{x_j\}$
6: **end for**
7: Check the number of nonempty G-sets: $G_1, \ldots, G_Q$
8: **for** $i = 1, 2, \ldots, Q$ **do**
9: Find the smallest closed box B_i that contains G_i, and compute its Lebesgue measure $\lambda(B_i)$
10: **end for**
11: Choose Q low discrepancy sets $\mathcal{A}^{(1)} \subset B_1, \ldots, \mathcal{A}^{(Q)} \subset B_Q$ such that $\#\mathcal{A}^{(i)} = \lceil m * \lambda(B_i) / \sum_i \lambda(B_i) \rceil$
12: **return** $\mathcal{A}_X = \bigcup_{i=1}^{Q} \mathcal{A}^{(i)}$

details on the implementation of Algorithm 4.1 are discussed in section 4.4. See Figure 1 for an illustration of anchor nets with increasing net size m constructed for a two dimensional highly nonuniform synthetic dataset.

Next we prove major properties of $\mathcal{A}_X$ returned by Algorithm 4.1. The main result is stated in Theorem 4.5.

First we prove the following lemma.

LEMMA 4.3. *For $i = 1, \ldots, Q$, define*

$$A_\epsilon^{(i)} := \limsup_{M_i \to \infty} \left\{ x \in \mathbb{R}^d : dist_\infty(x, \mathcal{A}^{(i)}) \le \epsilon \right\},$$

where $M_i = \#\mathcal{A}^{(i)}$. Then $\bigcap_{\epsilon>0} A_\epsilon^{(i)} = B_i$.

Proof. Without loss of generality, assume $B_i = [0,1]^d$. Our goal is to prove that $\bigcap_{\epsilon>0} A_\epsilon^{(i)} = B_i$. It is easy to see that $A_{\epsilon_1}^{(i)} \subset A_{\epsilon_2}^{(i)}$ whenever $\epsilon_1 < \epsilon_2$, so $\bigcap_{\epsilon>0} A_\epsilon^{(i)}$

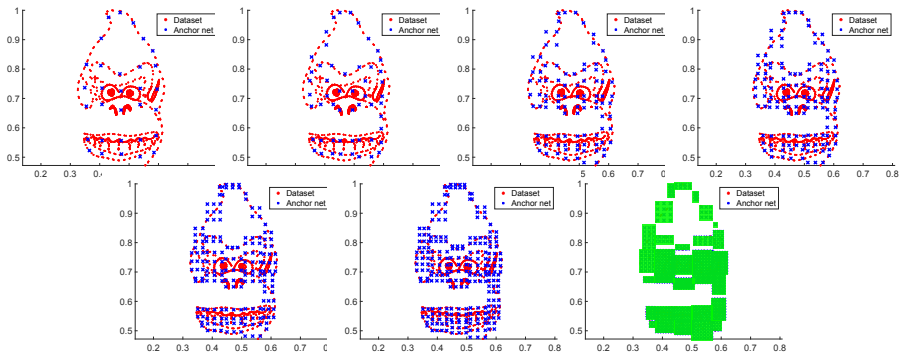

FIG. 1. *An illustration of six anchor nets (blue "x") of increasing net sizes m and $\Omega = \bigcup_{i=1}^{Q} B_i$ (green) on a highly nonuniform synthetic dataset (red dots).*

can be viewed as the limit of sets as $\epsilon \to 0$. We first show that $B_i \subset A_\epsilon^{(i)}$ for each $0 < \epsilon < 1$. Fix an $\epsilon \in (0,1)$. For an arbitrary $x \in B_i$, let J_x be the box centered at x with side ϵ. Define $J = J_x \cap [0,1)^d$. Then $\lambda(J) \geq (\frac{1}{2})^d \lambda(J_x) = \frac{1}{2^d}\epsilon^d$. Since $\mathcal{A}^{(i)}$ is a low discrepancy set, $D^*_{M_i}(\mathcal{A}^{(i)}) \to 0$ as $M_i \to \infty$. Therefore, for the tolerance $\frac{1}{5^d}\epsilon^d$, if M_i is large enough, we have

$$\left|\#(\mathcal{A}^{(i)} \cap J)/M_i - \lambda(J)\right| \leq D^*_{M_i}(\mathcal{A}^{(i)}) < \frac{\epsilon^d}{5^d} < \lambda(J),$$

which implies that $\mathcal{A}^{(i)} \cap J \neq \emptyset$. Hence there is a point in $\mathcal{A}^{(i)}$ whose l^∞ distance to x is within ϵ, i.e.,

$$\operatorname{dist}_\infty(x, \mathcal{A}^{(i)}) \leq \epsilon. \tag{4.2}$$

Note that (4.2) is true as long as M_i is large enough. Consequently, there are infinitely many M_i such that (4.2) holds true. According to the definition of lim sup, it follows that

$$x \in \limsup_{M_i \to \infty} \left\{ x \in \mathbb{R}^d : \operatorname{dist}_\infty(x, \mathcal{A}^{(i)}) \leq \epsilon \right\} = A_\epsilon^{(i)}.$$

This shows that $B_i \subset A_\epsilon^{(i)}$ since x is arbitrary in B_i. Because $\epsilon > 0$ is arbitrary, we see that $B_i \subset \bigcap_{\epsilon>0} A_\epsilon^{(i)}$. It remains to prove the other direction: $\bigcap_{\epsilon>0} A_\epsilon^{(i)} \subset B_i$. This is equivalent to the fact that if $x \notin B_i$, then $x \notin \bigcap_{\epsilon>0} A_\epsilon^{(i)}$. Now suppose $x \notin B_i = [0,1]^d$. Then $\operatorname{dist}_\infty(x, B_i) = \delta > 0$ for some positive constant δ. We know that $\mathcal{A}^{(i)} \subset B_i$, so $\operatorname{dist}_\infty(x, \mathcal{A}^{(i)}) \geq \delta > 0$ for any M_i. Therefore, $x \notin A_\delta^{(i)}$, which yields that $x \notin \bigcap_{\epsilon>0} A_\epsilon^{(i)}$. Now the second direction is proved, and we conclude that $\bigcap_{\epsilon>0} A_\epsilon^{(i)} = B_i$. ∎

The next lemma is a property of the generalized discrepancy.

LEMMA 4.4. *Let S_1 and S_2 be two finite subsets of $\Omega_1 \subset [0,\infty)^d$ and $\Omega_2 \subset [0,\infty)^d$, respectively. Suppose $\lambda(\Omega_1 \cap \Omega_2) = 0$ and $S_1 \cap S_2 = \emptyset$. If $D^*_{N_1,\Omega_1}(S_1) < \epsilon$ and $D^*_{N_2,\Omega_2}(S_2) < \epsilon$, where $N_i = \#S_i$, then*

$$D^*_{N_1+N_2,\Omega_1\cup\Omega_2}(S_1 \cup S_2) < \left|\frac{N_1}{N_1+N_2} - \frac{\lambda(\Omega_1)}{\lambda(\Omega_1)+\lambda(\Omega_2)}\right| + \epsilon. \tag{4.3}$$

Proof. Denote $\lambda_i = \lambda(\Omega_i)$ with $i = 1, 2$. Let $\mathcal{J}$ be the family of boxes as in Definition 4.2. For any $J \in \mathcal{J}$, define

$$A_i := \#(S_i \cap J), \quad b_i := \lambda(\Omega_i \cap J), \quad i = 1, 2.$$

According to Definition 4.2 and the assumptions in the claim, it suffices to show that

$$\begin{aligned} \left|\frac{\#((S_1 \cup S_2) \cap J)}{N_1+N_2} - \frac{\lambda((\Omega_1 \cup \Omega_2) \cap J)}{\lambda(\Omega_1 \cup \Omega_2)}\right| &= \left|\frac{A_1+A_2}{N_1+N_2} - \frac{b_1+b_2}{\lambda_1+\lambda_2}\right| \\ &< \left|\frac{N_1}{N_1+N_2} - \frac{\lambda_1}{\lambda_1+\lambda_2}\right| + \epsilon. \end{aligned} \tag{4.4}$$

Note first that the definition of $D^*_{N_i,\Omega_i}(S_i)$ yields

$$\left|\frac{A_i}{N_i} - \frac{b_i}{\lambda_i}\right| \leq D^*_{N_i,\Omega_i}(S_i) < \epsilon, \quad i = 1, 2. \tag{4.5}$$

It is easy to see that

$$\frac{A_1+A_2}{N_1+N_2}-\frac{b_1+b_2}{\lambda_1+\lambda_2}=\frac{N_1}{N_1+N_2}\cdot\frac{A_1}{N_1}+\frac{N_2}{N_1+N_2}\cdot\frac{A_2}{N_2}-\left(\frac{\lambda_1}{\lambda_1+\lambda_2}\cdot\frac{b_1}{\lambda_1}+\frac{\lambda_2}{\lambda_1+\lambda_2}\cdot\frac{b_2}{\lambda_2}\right).$$

Together with (4.5), we deduce that

$$\begin{aligned}&\left|\frac{A_1+A_2}{N_1+N_2}-\frac{b_1+b_2}{\lambda_1+\lambda_2}\right|<\\&\left|\left(\frac{N_1}{N_1+N_2}-\frac{\lambda_1}{\lambda_1+\lambda_2}\right)\cdot\frac{A_1}{N_1}+\left(\frac{N_2}{N_1+N_2}-\frac{\lambda_2}{\lambda_2+\lambda_2}\right)\cdot\frac{A_2}{N_2}\right|+\epsilon\\&\leq\left|\frac{N_1}{N_1+N_2}-\frac{\lambda_1}{\lambda_1+\lambda_2}\right|+\epsilon,\end{aligned}$$

where we have used the fact that $|\frac{A_1}{N_1}-\frac{A_2}{N_2}|\leq 1$. Since (4.4) is proved for any $J\in\mathcal{J}$, by taking a sup of the left-hand side of (4.4) over $J\in\mathcal{J}$, we conclude that (4.3) holds true. ∎

Based on Lemmas 4.3 and 4.4, we show in Theorem 4.5 the properties of the output of Algorithm 4.1. The first property says that the region $\Omega=\bigcup_{i=1}^{Q}B_i$ associated with anchor nets is able to compactly capture X, and the second property indicates that the anchor nets have low discrepancy in Ω.

THEOREM 4.5. *For a given dataset X, let $\mathcal{A}_X$ be the output of Algorithm* 4.1 *and $N:=\#\mathcal{A}_X$, $M_i:=\#\mathcal{A}^{(i)}$. Assume that $0<\tau_1\leq M_i/N\leq\tau_2<1$ for some constants $\tau_1,\tau_2\in(0,1)$. Define $\Omega=\bigcup_{i=1}^{Q}B_i$; then*

1. *Ω has the equivalent expression*

$$\Omega=\bigcap_{\epsilon>0}\limsup_{N\to\infty}\left\{x\in\mathbb{R}^d:dist_\infty(x,\mathcal{A}_X)\leq\epsilon\right\}\tag{4.6}$$

   with $\lambda(\Omega)>0$ and $X\subset\Omega$;
2. $\lim_{N\to\infty}D^*_{N,\Omega}(\mathcal{A}_X)=0$.

Furthermore, if (4.1) *holds for every i, then*

$$D^*_{N,\Omega}(\mathcal{A}_X)=O(N^{-1}(\log N)^d).\tag{4.7}$$

Proof. We verify that the two conditions are satisfied by $\mathcal{A}_X=\bigcup_{i=1}^{Q}\mathcal{A}^{(i)}$.

Since $\mathcal{A}_X=\bigcup_{i=1}^{Q}\mathcal{A}^{(i)}$ and $M_i/N\in(\tau_1,\tau_2)$, we see that

$$\Omega=\bigcup_{i=1}^{Q}\bigcap_{\epsilon>0}\limsup_{M_i\to\infty}\left\{x\in\mathbb{R}^d:\operatorname{dist}_\infty(x,\mathcal{A}^{(i)})\leq\epsilon\right\}=\bigcup_{i=1}^{Q}\bigcap_{\epsilon>0}A^{(i)}_\epsilon,$$

where $A^{(i)}_\epsilon$ is defined as in Lemma 4.3. According to Lemma 4.3, it follows that $\Omega=\bigcup_{i=1}^{Q}B_i$. In addition, we have the estimation

$$\lambda(\Omega)\geq\lambda(B_1)>0\quad\text{and}\quad X\subset\bigcup_{i=1}^{Q}G_i\subset\bigcup_{i=1}^{Q}B_i=\Omega,$$

which justifies the first condition.

Next we prove the second property:

$$\lim_{N\to\infty} D^*_{N,\Omega}(\mathcal{A}_X) = 0. \tag{4.8}$$

This is proved by using Lemma 4.4. Assume at this moment $Q = 2$. Then $\mathcal{A}_X = \mathcal{A}^{(1)} \cup \mathcal{A}^{(2)}$, $\Omega = B_1 \cup B_2$. We deduce from Lemma 4.4 that

$$\lim_{N\to\infty} D^*_{N,\Omega}(\mathcal{A}_X) \le \lim_{N\to\infty} \left| \frac{M_1}{M_1 + M_2} - \frac{\lambda(B_1)}{\lambda(B_1) + \lambda(B_2)} \right| + \lim_{N\to\infty} \max_{i=1,2} D^*_{M_i,B_i}(\mathcal{A}^{(i)}), \tag{4.9}$$

where the first term in the upper bound goes to zero due to (4.1) and the second term also vanishes because of the fact that $\lim_{M_i\to\infty} D^*_{M_i,B_i}(\mathcal{A}^{(i)}) = 0$ and $M_i/N \in (\tau_1, \tau_2)$. If $Q > 2$, based on the result for $Q = 2$, we can apply Lemma 4.4 inductively to show that the condition holds true for $Q = 3, 4, \ldots$. Therefore, (4.8) is justified.

Finally it remains to prove (4.7). This is actually an immediate result of (4.9). Consider $Q = 2$. Under the above assumption, it follows from (4.9) that

$$\lim_{N\to\infty} D^*_{N,\Omega}(\mathcal{A}_X) \le \lim_{N\to\infty} \max_{i=1,2} D^*_{M_i,B_i}(\mathcal{A}^{(i)}). \tag{4.10}$$

Since $\mathcal{A}^{(i)}$ is a low discrepancy set in B_i, $D^*_{M_i,B_i}(\mathcal{A}^{(i)}) = O(M_i^{-1}(\log M_i)^d)$. According to the assumption in the theorem, i.e., there are constants $\tau_1, \tau_2 \in (0, 1)$ such that $\tau_1 N \le M_i \le \tau_2 N$, we see that $O(M_i^{-1}(\log M_i)^d) = O(N^{-1}(\log N)^d)$. Therefore, (4.10) implies $D^*_{N,\Omega}(\mathcal{A}_X) = O(N^{-1}(\log N)^d)$, which completes the proof. ∎

It should be pointed out that even though the first condition in Theorem 4.5 says that Ω is large enough to capture X, it does *not* indicate that Ω will be *unnecessarily* large. Note that Ω adapts to the geometry of X and can be roughly viewed as a region spanned by X, as illustrated in Figure 1. For highly nonuniform datasets, sampling in Ω will be more efficient than in one single box that contains X. This is because Ω nicely reflects the geometry of the dataset X and thus uniform distribution (guaranteed by the second property) in Ω is expected to yield uniform distribution in X, as can be seen from the last subfigure in Figure 1.

4.3. Anchor net method. In this section we propose the anchor net method for selecting landmark points and prove its computational complexity. The anchor net method starts with the construction of an anchor net for the given dataset X and then searchs for the landmark points in the vicinity of the anchor net. The algorithm is presented in Algorithm 4.2.

Algorithm 4.2. *Anchor net method.*

Input: Dataset $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, integer m
Output: The set of landmark points S

1: Apply Algorithm 4.1 with net size m to construct the anchor net $\mathcal{A}_X$ for X
2: **for** each point y in $\mathcal{A}_X$ **do**
3: Find x_i such that $x_i = \text{argmin}_{x_k \in X} ||x_k - y||_\infty$
4: Update $S = S \cup \{x_i\}$
5: **end for**
6: **return** S

Since the landmark points are selected in the vicinity of the anchor net in Algorithm 4.2, the selected landmark points are uniformly spread inside the dataset. In

Proposition 4.6, we show that the computational cost of Algorithm 4.2 scales linearly in n.

PROPOSITION 4.6. *The complexity of the anchor net method described in Algorithm* 4.2 *with net size* m *is* $O(mdn)$.

Proof. First we calculate the complexity of Algorithm 4.1. Since $s = O(m)$, it is easy to see that step 1 costs $O(md)$ and the **for** loop in steps 3–6 amounts to $O(mdn)$. Since $G_1, \ldots, G_Q$ forms a disjoint partition of X, we have $\#G_1 + \cdots + \#G_Q = n$. The cost of the **for** loop in steps 8–10 is then $d \cdot \#G_1 + \cdots + d \cdot \#G_Q = dn$. The cost of step 11 is $d \cdot \#\mathcal{A}^{(1)} + \cdots + d \cdot \#\mathcal{A}^{(Q)} = dO(m)$. Overall, we see that the complexity of Algorithm 4.1 is $O(mdn)$.

Now we compute the overall complexity of Algorithm 4.2. Since $\#\mathcal{A}_X = O(m)$, the **for** loop in steps 2–5 of Algorithm 4.2 costs $O(mdn)$. Thus we conclude that the overall complexity of Algorithm 4.2 is $O(mdn)$, and the proof is complete. ∎

It is known that both uniform sampling and k-means Nyström methods tend to generate more sample points from regions with a high density of points, which can *not* effectively help reduce the Nyström approximation error. Different from those density-based approaches, anchor net $\mathcal{A}_X$ is designed to efficiently tessellate the given data to avoid the formation of clumps. Because of the geometric properties of anchor nets, the anchor net method can yield more accurate Nyström approximation with same approximation rank, regardless of the positive-definiteness of the kernel function. It should be emphasized that a good selection of landmark points also benefits the numerical stability of the Nyström method, which significantly affects the quality of the approximation. We discuss in section 4.4.2 the stability issue associated with the Nyström method and provide an numerical example in section 5.2 to demonstrate the impact of landmark points on approximation accuracy and numerical stability.

4.4. Practical implementation. In this section, we discuss several implementation details of the proposed method.

4.4.1. Adaptive tensor grids. Though tensor grids display perfect uniformity, they are not used for high dimensional data due to the curse of dimensionality. The naive construction of tensor grid by employing a parameter that specifies the number of points per direction is not practical in high dimension since the degrees of freedom (DOFs) depend exponentially on dimension. In this section, we propose an adaptive tensor grid to significantly reduce the exponential growth of DOFs with dimension, which enables the practical use of tensor grids.

Instead of treating approximation in each dimension independently, we control the *total* number of nodes per direction over *all* dimensions. That is, for a nonnegative integer p, if i_k is the number of nodes in the kth dimension, then we require $i_1 + \cdots + i_d = p + d$. This new strategy yields significantly fewer DOFs and results in a much slower growth of DOFs with respect to d or p, as illustrated in Figure 2. An upper bound of the DOFs is given in Proposition 4.7.

PROPOSITION 4.7. *Let* p *be a nonnegative integer. Consider a tensor grid in* $\mathbb{R}^d$ *with* i_k *points* ($i_k \geq 1$) *in the* k*th dimension such that* $i_1 + \cdots + i_d = p + d$. *Then the total number of nodes is bounded by* e^p, *i.e.,* $i_1 i_2 \cdots i_d \leq (\frac{p+d}{d})^d < e^p$.

Proof. The second inequality in the estimate follows from the fact that

$$\left(1 + \frac{p}{d}\right)^{d/p} < e.$$

We now prove the first inequality by induction on d. For $d = 1$, the inequality automatically holds true. Assume that the inequality holds true for $\mathbb{R}^d$. For $\mathbb{R}^{d+1}$,

there are $p+1$ possible values for i_{d+1}. That is, $i_{d+1} = k$ and $i_1 + \cdots + i_d = p + d + 1 - k$, $k = 1, \ldots, p+1$. Applying the induction assumption for $\mathbb{R}^d$ gives $i_1 i_2 \cdots i_d \leq (\frac{p+d+1-k}{d})^d$. Hence

$$\max_{\substack{|\mathbf{i}|=p+d+1 \\ i_{d+1}=k}} i_1 i_2 \cdots i_d i_{d+1} \leq k \left(\frac{p+d+1-k}{d} \right)^d \leq \max_{1 \leq k \leq p+1} g(k),$$

where $g(x) := x(\frac{p+d+1-x}{d})^d$. Next we show that $g(k)$ is bounded by $(\frac{p+d+1}{d+1})^{d+1}$. By computing $g'(x)$,

$$g'(x) = \left(\frac{p+d+1-x}{d} \right)^{d-1} \frac{p+d+1-(d+1)x}{d},$$

we see that g has a unique maximizer at $x_* = (p+d+1)/(d+1)$ in $[0, p+1]$. Therefore,

$$\max_{|\mathbf{i}|=p+d+1} i_1 i_2 \cdots i_d i_{d+1} \leq \max_{1 \leq k \leq p+1} g(k) \leq g(x_*) = \left(\frac{p+d+1}{d+1} \right)^{d+1}.$$

We conclude that the inequality holds for $\mathbb{R}^{d+1}$, and the proof is complete. ∎

Figure 2 shows a comparison between DOFs of the uniform tensor grid (dotted line) and the new one (solid line). In the uniform tensor grid, $p+1$ denotes the number of nodes in each dimension, while in adaptive tensor grid, $p+d$ controls the sum of numbers of nodes in each dimension. The left subfigure plots the DOFs with respect to dimension d when $p = 2, 3, 4, 5$, and the right subfigure plots the DOFs with respect to p at different dimensions $d = 2, 3, 4, 5$. It can be seen from the left plot in Figure 2 that the classical tensor grid (dotted line) yields exponentially increasing DOFs with the dimension, while the new one (solid line) is immune to the increase of dimension. The right plot in Figure 2 shows that, compared to the old method, the new method yields a much slower growth of DOFs as p increases. We see from both figures that the new method is not sensitive to the increase of dimension d. Adaptive tensor grids control the *rate of increase* of DOFs across different levels of approximation by adding more intermediate levels. The numerical experiments in section 5 demonstrate that the approximation error decreases as more DOFs are used in the adaptive tensor grid.

Although adaptive tensor grids share the same goal as sparse grids [3] to control the number of generated nodes in high dimensions, there are several major differences

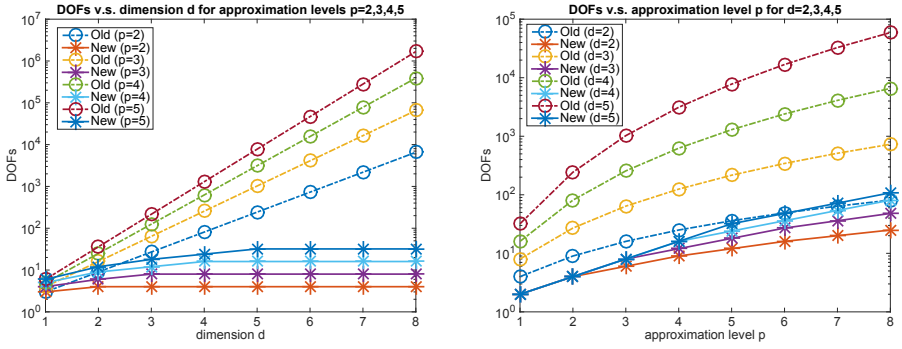

FIG. 2. *Left: DOFs vs dimension d; Right: DOFs vs p.*

between them: (1) Sparse grids use highly nonuniform nodes in the cubic domain. For example, along a specific dimension (for example, $y = 0$ in the two dimensional case), the nodes are sparser in the interior and denser near the boundary. On the other hand, adaptive tensor grids tend to generate uniformly distributed nodes in the dataset. (2) Despite the fact that sparse grids reduce the exponential dependence on the dimension d to a polynomial one, from p^d to d^p, the actual number of DOFs can still be very large even for a moderate d. For example, as shown in [3], when p (maximum number of nodes per dimension) increases from 1 to 7, the number of DOFs increases from 21 to 652065 for a dimension $d = 10$ problem. Therefore, if one wants higher accuracy by increasing p, significantly more DOFs will be generated. On the other hand, as shown in the right subfigure of Figure 2 the number of nodes increases at a much slower rate in adaptive tensor grids as the approximation level p increases. (3) Sparse grids are used for approximating functions and high dimensional integrals instead of matrix approximations, particularly the Nyström method for low-rank factorization. The motivation of sparse grid is to reduce the cost in approximating a continuous problem (e.g., a function, an integral) in high dimensions, while a matrix is a discrete object.

4.4.2. Numerical techniques for improving stability. The Nyström formula requires computing K_{SS}^{+}, the pseudoinverse of the kernel matrix associated with the landmark points. In some cases, the resulting kernel matrix can be nearly singular, causing numerical instability when computing the exact pseudoinverse. The issue can be circumvented for SPSD kernels by regularization techniques, i.e., adding a scalar matrix αI with a small constant $\alpha > 0$ to lift all eigenvalues to (α, ∞) and computing the inverse of the sum. For indefinite kernels, however, regularization is no longer effective since K_{SS} may have both positive and negative eigenvalues around 0. A well-known method that can handle both cases is to use the ϵ-pseudoinverse $K_{SS,\epsilon}^{+}$ in place of K_{SS}^{+}, where $K_{SS,\epsilon}$ is derived from K_{SS} by treating singular values smaller than ϵ as zeros. The modified Nyström approximation with a truncated pseudoinverse then becomes

$$K_{XX} \approx K_{XS} K_{SS,\epsilon}^{+} K_{SX}. \tag{4.11}$$

Some other alternatives have also been proposed. For example, [33] proposed the following QR-based approximation in place of K_{SS}:

$$K_{XX} \approx (K_{XS} R_{\epsilon}^{+})(Q^T K_{SX}), \tag{4.12}$$

where $K_{SS} = QR$ is the QR factorization of K_{SS} and R_ϵ is derived from R by truncating singular values smaller than ϵ, similar to $K_{SS,\epsilon}$ with respect to K_{SS}. In section 5, we perform numerical tests to show that the truncation techniques do rectify the stability issue. However, aside from improved stability, numerical results show that (4.11) impairs the accuracy of the original Nyström approximation. Although (4.12) performs better than (4.11), the approximation still becomes less accurate as the number of landmark points increases. In general, numerical techniques require accurate computation of singular values close to zero for a numerically low-rank matrix and are *not* able to fully resolve the structural issues on accuracy and stability. In this paper, we alleviate this issue by choosing a good selection of landmark points to improve the conditioning of the K_{SS}, as demonstrated in section 5.

5. Numerical experiments. In this section we present various experiments to demonstrate the performance of the anchor net method and the numerical instability

TABLE 1
Datasets (n instances in d dimensions).

	Donkey Kong	Abalone	Anuran Calls (MFCC)	Covertype
d	2	8	22	54
n	3000	4177	7195	581012

of some Nyström methods for kernel matrices with rapidly decaying singular values. The datasets are shown in Table 1. All experiments were performed in MATLAB 2020b on a desktop with an Intel i9-9900K 3.60 GHz CPU and 64 GB of RAM. The 2-norm is used to measure the Nyström approximation error in all experiments except the one in Figure 9, where the 2-norm cannot be computed accurately and the Frobenius norm is used instead. For probabilistic methods like uniform sampling, the error is averaged over 10 repeated runs, and in each error-rank plot, the solid line corresponds to the averaged error, while the dotted line corresponds to the error in an individual run. See, for example, Figures 3–5. For the anchor net construction, we choose the low discrepancy set to be the adaptive tensor grid discussed in section 4.4.1, as it is straightforward to parametrize adaptive tensor grids using the sum of the number of nodes in each direction. In Algorithm 4.1, we choose $\mathcal{T}$ to be larger than $\mathcal{A}^{(i)}$ to tessellate the dataset more efficiently, especially in high dimensions. For example, empirical results show that the size of $\mathcal{T}$ can be chosen to be 2 to 20 times larger than the size of $\mathcal{A}^{(i)}$, with a larger ratio for higher dimensions.

5.1. Indefinite kernels. We consider the following indefinite kernels:

$$\begin{aligned} \text{Multiquadrics}: &\quad \kappa(x,y) = \sqrt{|x-y|^2/\sigma^2 + 1}, \\ \text{sigmoid}: &\quad \kappa(x,y) = \tanh(x \cdot y/\sigma + 1), \\ \text{Thin plate spline}: &\quad \kappa(x,y) = \frac{|x-y|^2}{\sigma^2} \ln\left(\frac{|x-y|^2}{\sigma^2}\right). \end{aligned}$$

Those kernels are commonly seen in deep learning, kernel density estimation, statistics, etc. To the best of our knowledge, the only Nyström methods that could potentially work for indefinite kernels are the uniform method [44] and the k-means Nyström method [46, 45]. Hence we compare our method to those two. (Note that leverage score sampling–based Nyström methods, such as [12, 17, 32], cannot be applied here since they require the kernel matrices to be SPSD.) The k-means method is implemented with an efficient vectorized function to compute L_2 distances between points and centroids at each iteration. The iteration number is set to 5. We test the three Nyström methods over the following high dimensional datasets from the UC Irvine Machine Learning Repository:[1] Abalone, Anuran Calls (mel-frequency cepstral coefficient (MFCC)), Covertype. See Table 1 for the statistics of the datasets. The datasets are standardized to have zero mean and unit variance. For each kernel, we choose σ to be the half radius of the standardized dataset, where the radius is defined as the maximum distance from a point to the center. The choice ensures that the resulting kernel matrices have fast singular value decay and are thus suitable for low-rank approximations. For the Covertype dataset ($n = 581012$), the Nyström approximation error is evaluated over 10000 randomly sampled points from the dataset.

[1] Datasets can be found at https://archive.ics.uci.edu/ml/index.php.

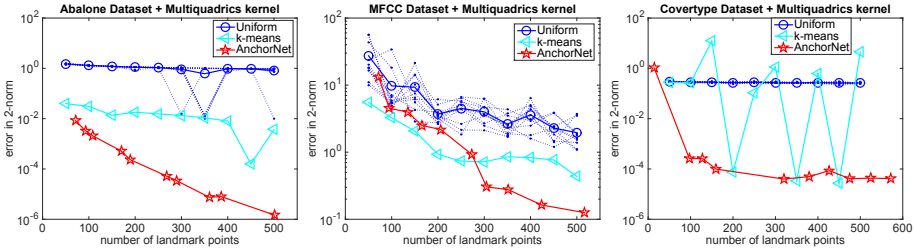

FIG. 3. *Multiquadrics: Abalone (left), MFCC (middle), Covertype (right).*

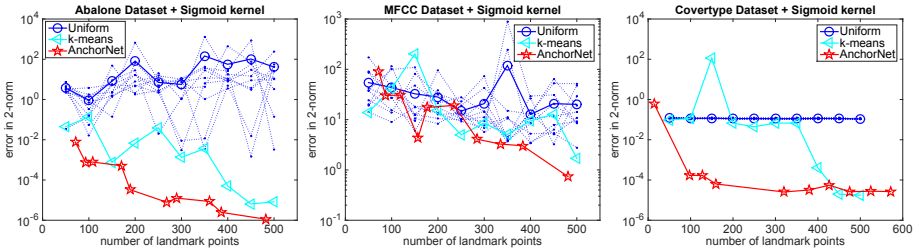

FIG. 4. *Sigmoid: Abalone (left), MFCC (middle), Covertype (right).*

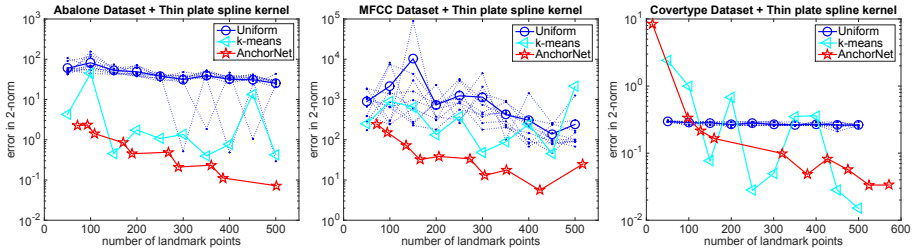

FIG. 5. *Thin plate spline: Abalone (left), MFCC (middle), Covertype (right).*

The error-rank plots in Figures 3–5 illustrate how the Nyström approximation error changes as the number of landmark points increases. The computational cost associated with each method is shown in the error-time plots in Figures 6–8, where the runtime for each method is computed over ten repeated runs and the approximation error for uniform Nyström method is averaged over ten runs.

We have the following observations regarding the accuracy and stability of the Nyström schemes under comparison for approximating different kinds of *indefinite* kernel matrices.

1. According to Figures 3–8, we see that, for different indefinite kernels and datasets, the anchor net method achieves overall the best accuracy for a given approximation rank (i.e., the number of landmark points) and requires the least computation time. It is overall more stable than uniform sampling and k-means methods. We also note that the advantage of anchor net method is more prominent for large-scale high dimensional datasets like Covertype.
2. Compared to uniform sampling and anchor net methods, the k-means clustering can be quite unstable as one increases the approximation rank, as illustrated in Figures 3 (right), 4, and 5. This is due to the heuristic and iterative nature of the k-means clustering: the computed cluster centers after

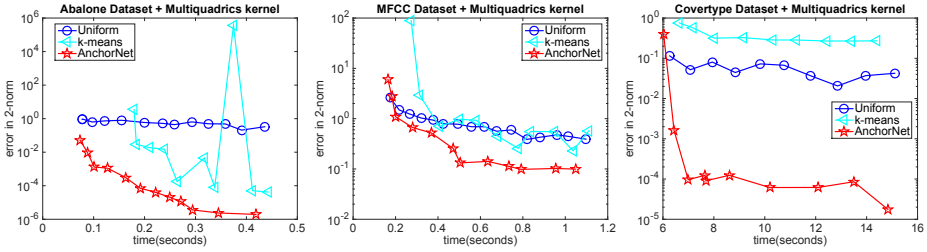

FIG. 6. *Multiquadrics error-time plot: Abalone (left), MFCC (middle), Covertype (right).*

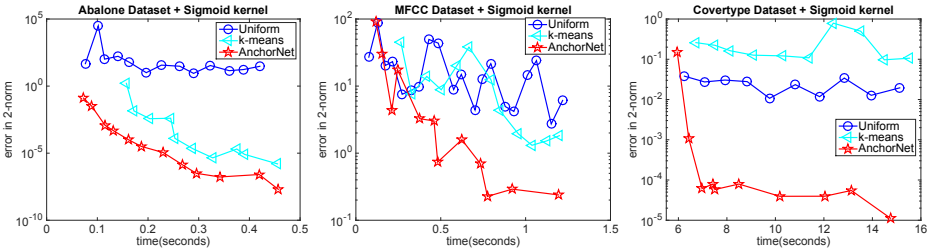

FIG. 7. *Sigmoid error-time plot: Abalone (left), MFCC (middle), Covertype (right).*

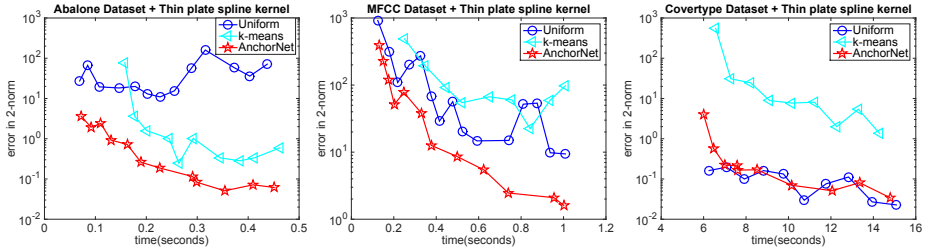

FIG. 8. *Thin plate spline error-time plot: Abalone (left), MFCC (middle), Covertype (right).*

a few iterations are unpredictable, and it's hard to predict whether the final output can yield a better Nyström approximation accuracy than the initial guess.

3. It's easy to see from Figure 3 (right) and Figure 4 (right) that the anchor net method converges much faster to a fixed accuracy.
4. For large-scale datasets in high dimensions (e.g., Covertype), the k-means Nyström method is quite slow and is outperformed by uniform sampling according to Figures 6 (right), 7, and 8 (right).
5. For the sigmoid kernel with MFCC dataset in Figure 7 (middle), all three Nyström schemes display oscillatory behaviors, but the anchor net method stays at a much lower error level, so it actually oscillates with a much smaller amplitude than the other two methods.
6. We see that indefinite kernel matrices are in general much harder for Nyström methods to approximate than SPSD matrices. This is because indefinite kernels have both positive and negative eigenvalues around the origin. As a result, the Nyström approximation is more sensitive to numerical instability. Existing general Nyström schemes (uniform sampling and k-means) can be quite unstable for indefinite kernels, while the anchor net method is very robust and meanwhile achieves better accuracy with less computational cost.

5.2. Geometry of landmark points and numerical issues for indefinite kernels. In this subsection, we investigate two issues: (1) how the geometry of landmark points impacts the accuracy as well as numerical stability of the resulting Nyström approximation; (2) how the stabilization techniques (4.11)–(4.12) influence the accuracy of Nyström approximation.

Geometry of landmark points. To illustrate the effect of geometry of landmark points on the Nyström approximation, we consider the sigmoid kernel with $\sigma = 1$ over a two dimensional highly nonuniform dataset illustrated in Figure 9 (left). The singular values of the corresponding kernel matrix decay rapidly, and as a result, Nyström approximation is subject to numerical instability if landmark points are not well chosen.

In terms of the selection of landmark points, it can be clearly seen from Figure 9 that both uniform sampling and k-means clustering tend to generate more landmark points in denser regions of the dataset, for example, around $(0.4, 0.3)$, $(0.5, 0.7)$, etc. This does *not* contribute to a better approximation and, conversely, may lead to numerical instability and possibly a much worse approximation than the one with fewer landmark points.

As reflected in the error plot in Figure 9 (right), over ten repeated runs, uniform sampling often becomes ineffective due to the poor choice of landmark points S, which causes the approximation error to blow up when computing K_{SS}^{+}. The k-means Nyström method, on the other hand, can sometimes achieve high accuracy when k is small but becomes quite unstable as k increases. Figure 9 (right) shows that the k-means Nyström method breaks down when k increases from around 220 to 440. As the number of clusters increases, computing the centroids of the clusters puts more weight on small dense clusters that contain a large number of points close to each other. This will result in more landmark points (centroids) close to those dense clusters, eventually causing numerical instability when computing the Nyström approximation. It can be seen that the anchor net method remains robust besides being the most accurate as the number of landmark points increases. Overall, for indefinite kernel matrices and highly nonuniform data, existing Nyström methods tend to generate landmark points that result in an extremely unstable and inaccurate approximation, while the anchor net method is able to yield accurate and robust approximation by choosing geometrically well-balanced landmark points with no clumps.

Performance of stabilization techniques. We then consider the same problem as in Figure 9 but use the "stabilized" Nyström approximations based on (4.11) and (4.12) to investigate the impact of using the approximate pseudoinverse $K_{SS,\epsilon}^{+}$ as compared to K_{SS}^{+}. We compute each of the two "stabilized" Nyström approximations in (4.11) and (4.12) using three methods: uniform sampling, k-means, and anchor net. To study the impact of truncation in (4.11) and (4.12), we use four different values of truncation tolerance: $\epsilon = 10^{-8}, 10^{-10}, 10^{-12}, 10^{-14}$. For each ϵ, we compare the

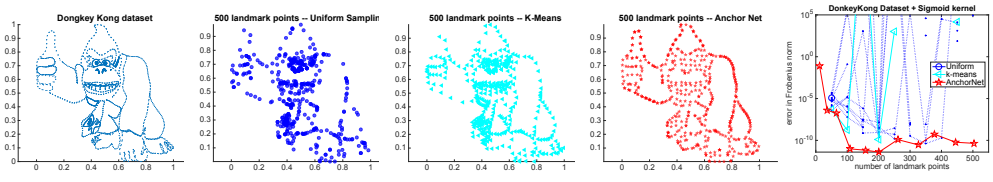

FIG. 9. *Left to right: Donkey Kong dataset,* 500 *landmark points generated by three methods, error-rank plot for approximating the sigmoid kernel matrix.*

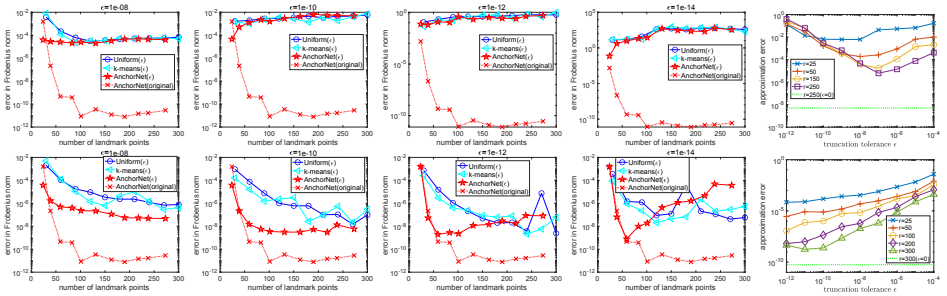

FIG. 10. *Approximation errors using stabilization techniques: ϵ-pseudoinverse (top row) in* (4.11) *and ϵ-QR (bottom row) in* (4.12). *First four figures in each row are error-rank plots of "stabilized" Nyström methods (uniform sampling, k-means, anchor net) with $\epsilon = 10^{-8}, 10^{-10}, 10^{-12}, 10^{-14}$ and original anchor net Nyström (dotted line with "×"), respectively. The last figure shows approximation errors of the k-means Nyström method versus truncation tolerance ϵ, where several ranks are used and the dotted line with the "×" symbol denotes a fixed-rank approximation error using the original Nyström formula* without *stabilization.*

performance of three Nyström schemes. The resulting four error-rank plots are shown in Figure 10. As expected, we see that the truncation techniques *do* stabilize the Nyström approximation for uniform sampling and k-means as compared to Figure 9. However, we also see that the stabilized Nyström approximation in (4.11) significantly worsens the *accuracy* of the Nyström approximation. In Figure 9, we see that, despite stability, all three methods are able to achieve high accuracy, for example, around 9 to 11 digits when the rank is 200. According to Figure 10 (top), with the stabilized approximation, all three methods can at most achieve around 5 digits of accuracy. Meanwhile, different values of ϵ yield quite different approximation accuracy, and in practice it is hard to determine which one should be used.

The results in Figure 10 also show that stabilization techniques may harm the accuracy when the original Nyström approximation is accurate enough. This is easily seen in Figure 10 by comparing stabilized anchor net–based approximation (red solid line) to the original version (red dotted line), where both stabilization techniques lead to orders of magnitude loss of accuracy. This can be seen from the right-most plots in Figure 10. We also see that the stabilized approximation may not achieve as good accuracy as the original Nyström method.

By looking at the fourth plot $\epsilon = 10^{-14}$ on the bottom row in Figure 10, we see that the QR-based stabilization in (4.12) is accurate when the rank is small but then leads to numerical instability as rank increases (see red solid line). Neither of the two stabilization techniques is able to achieve the same level of accuracy that the anchor net method attains without stabilization. Overall, the results show that numerical techniques to resolve stability issues may lead to worse approximation and the error from the ϵ-truncation may dominate the Nyström approximation error, especially in the high accuracy regime. Thus we see that stabilization techniques are not able to fully resolve the numerical issues associated with Nyström method, and a more appropriate solution should come from a good choice of landmark points, as demonstrated by the anchor net method in Figure 10.

5.3. Nyström variants for SPSD kernels. To illustrate the possible numerical instability of existing Nyström methods for SPSD kernel matrices, we consider the approximation of the Gaussian kernel matrix (which is SPSD) with rapidly decaying singular values. Since the kernel is SPSD, the numerical instability can be remedied

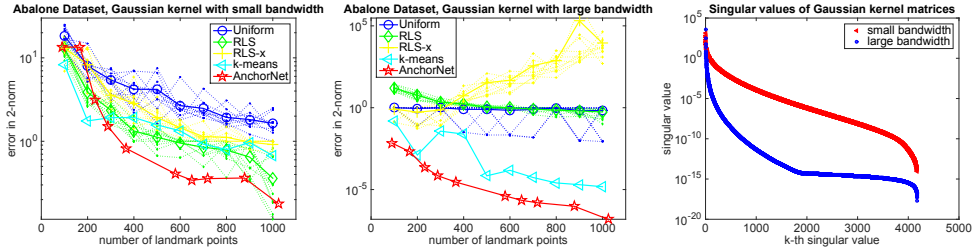

FIG. 11. *Error-rank plot for approximating a Gaussian kernel matrix with* $\sigma = 2.3$ *(left) and* $\sigma = 11.8$ *(middle) and singular values of the two kernel matrices (right).*

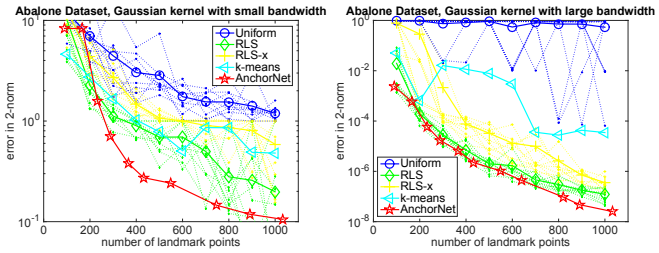

FIG. 12. *Error-rank plot for approximating a regularized Gaussian kernel matrix with* $\sigma = 2.3$ *(left) and* $\sigma = 11.8$ *(right).*

via regularization, i.e., approximating $K + \beta I$ for a small constant $\beta > 0$. We present results for both K and $K + \beta I$ and choose $\beta = 10^{-9}$. The proposed method (anchor net) is compared to the following Nyström schemes: (1) the original uniform Nyström method [44], which was observed in [27] to yield satisfactory overall performance (error-time trade-off) compared to several other methods; (2) the k-means clustering Nyström method [46, 45], which usually yields better accuracy than the uniform Nyström method; (3) the recursive ridge leverage score (RLS) Nyström method [32], which improves the efficiency of the original leverage score–based sampling; (4) the accelerated recursive RLS (RLS-x) Nyström method [32], which is much faster than RLS but may not be as robust. For probabilistic approaches (uniform samplig, RLS, RLS-x), the error is averaged over ten repeated runs.

The methods above are compared from two perspectives: numerical stability and computational efficiency. The Gaussian kernel $\kappa(x, y) = e^{-|x-y|^2/\sigma^2}$ is used, and the two experiment settings are listed below.

1. **Numerical stability.** We consider two Gaussian kernels with different choices of the bandwidth parameter σ: 10% and 50% times the radius of the standardized Abalone dataset. Note that larger σ leads to faster singular value decay of the kernel matrix. Without regularization, the results are presented in Figure 11. With regularization, the results are shown in Figure 12.
2. **Computational efficiency**. We consider two datasets: Abalone ($n = 4177, d = 8$) and Covertype ($n = 581012, d = 54$). For Abalone, we choose $\sigma = 2.3$; for Covertype, σ is same as the one used in section 5.1. The experiment results are collected as error-time plots in Figure 13 for K and Figure 14 for $K + 10^{-9}I$. The Covertype dataset is quite large and high dimensional compared to the Abalone dataset, and the results for the two datasets are quite different, as can be seen in Figure 13.

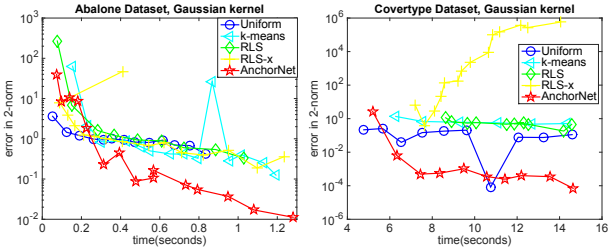

FIG. 13. *Error-time plot for approximating a Gaussian kernel matrix with the Abalone dataset (left) and Covertype dataset (right).*

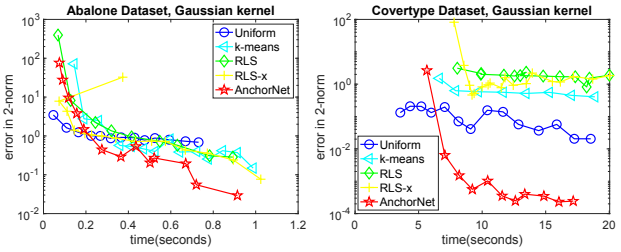

FIG. 14. *Error-time plot for approximating a regularized Gaussian kernel matrix with the Abalone dataset (left) and Covertype dataset (right).*

According to Figures 11–Figure 14, we have the following observations:

1. Overall, the anchor net method is more accurate and robust compared to other Nyström methods. It achieves significantly better error-time trade-off for large-scale high dimensional datasets.
2. As can be seen from Figure 11 (middle), for SPSD kernel matrices with rapidly decaying singular values, probabilistic methods are subject to numerical instability. Via regularization, the issue can be resolved for RLS and RLS-x but not for uniform sampling; cf. Figure 12 (right). The anchor net method, on the other hand, achieves best accuracy without requiring regularization.
3. For large-scale high dimensional datasets like Covertype, we see from Figure 13 and Figure 14 that the anchor net method is able to reach high accuracy in a significantly shorter time than other methods. Aside from numerical stability, this demonstrates the superior efficiency of anchor net method in practice.

Remark 5.1. As shown in Figure 11 (right), the kernel matrix with larger σ has faster singular value decay and consequently is more suitable for low-rank approximations. Nevertheless, it should be emphasized that better spectral property does *not* necessarily imply more accurate Nyström approximations. Instead, it poses a great numerical challenge for the effective use of Nyström approximations: K_{SS} may have many singular values near 0, and computing K_{SS}^{+} will be numerically *unstable* unless the landmark points S are well chosen. This indicates that the Nyström approximation accuracy can become even worse as the number of landmark points increases. As one can see in Figure 11 (middle) as well as Figure 9 (right), this is indeed the case for many Nyström schemes.

5.4. Nyström methods and pivoted Cholesky factorization for SPSD matrices. In this section, we compare the k-means Nyström method and anchor net

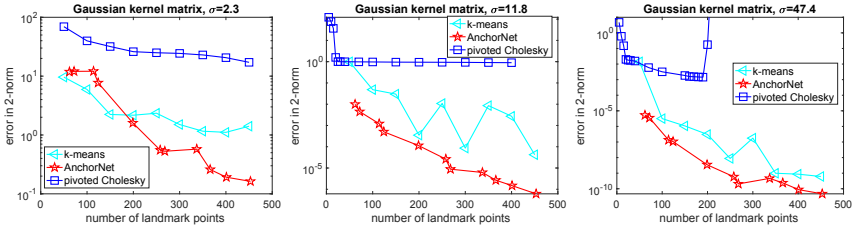

FIG. 15. *Approximating three Gaussian kernel matrices with bandwidths:* $\sigma = 2.3, 11.8, 47.4$ *(left to right).*

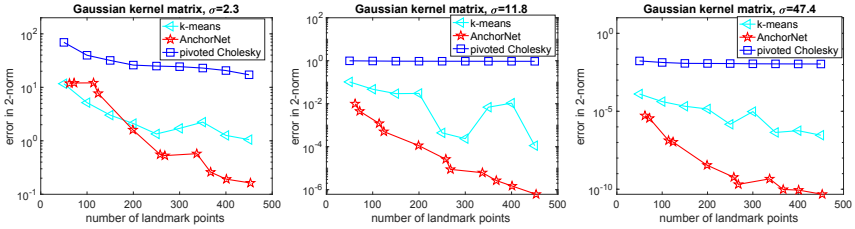

FIG. 16. *Approximating regularized Gaussian kernel matrices with bandwidths:* $\sigma = 2.3, 11.8, 47.4$ *(left to right).*

method to partially pivoted Cholesky decomposition in [23], which was shown to work well for SPSD kernel matrices associated with *low* dimensional datasets. We consider the Gaussian kernel $\kappa(x, y) = \exp(-|x - y|^2/\sigma^2)$ and form the matrix K_{XX} with Abalone dataset $(n = 4177, d = 8)$. For the bandwidth parameter σ, we use three different values, $\sigma = 2.3, 11.8, 47.4$, to investigate the performance of three methods. The matrix with $\sigma = 2.3$ has the slowest singular value decay, while the matrix with $\sigma = 47.4$ has the fastest singular value decay.

We consider approximating kernel matrices without and with regularization, i.e., K and $K + \beta I$, where the regularization parameter is chosen as $\beta = 10^{-9}$. The test results are presented in Figure 15 and Figure 16, respectively. From Figure 15, we see that the performance of partially pivoted Cholesky decomposition is quite sensitive to the bandwidth parameter if no regularization is applied to K. In this case, large σ can lead to numerical instability as approximation rank increases, while small σ can lead to slow error decay and poor approximation accuracy. The numerical instability of the partially pivoted Cholesky method is not seen when regularization is applied to K according to Figure 16.

The Nyström methods achieve better accuracy than pivoted Cholesky decomposition in all cases. It is easy to see that the anchor net method is least sensitive to σ, achieving the best accuracy and numerical stability.

6. Conclusion. In this paper, we first analyze the Nyström approximation error in the most general setting covering both SPSD and indefinite kernel matrices. The theoretical finding indicates that landmark points should encode the geometry of the dataset to avoid numerical instability and meanwhile to improve the approximation accuracy. Guided by the theoretical results, we propose the anchor net method for performing Nyström approximation with linear complexity in time and space. The proposed method is valid for both SPSD and indefinite kernels and is efficient in high dimensions. Comprehensive experiments covering indefinite and SPSD kernels, low and high dimensional data, and original and stabilized Nyström approximations are

performed to investigate the performance of existing methods in terms of accuracy, numerical stability, and speed. Overall, the anchor net method displays the best numerical stability and computational efficiency. It is able to achieve better accuracy than other Nyström schemes with smaller computational costs and demonstrate excellent accuracy and numerical stability for indefinite kernels compared to other methods with stabilized techniques. We plan to integrate the method into the computation of hierarchical matrices [20, 5, 4, 8], which will significantly extend the scope of applications.

Acknowledgments. The authors are indebted to Michele Benzi for his suggestion on improving the presentation of the theoretical analysis and to Yuji Nakatsukasa for the helpful discussion on the stable implementation of pseudoinverse.

REFERENCES

[1] A. Alaoui and M. W. Mahoney, *Fast randomized kernel ridge regression with statistical guarantees*, in Advances in Neural Information Processing Systems 28, MIT Press, Cambridge, MA, 2015, pp. 775–783.
[2] F. R. Bach and M. I. Jordan, *Kernel independent component analysis*, J. Mach. Learn. Res., 3 (2002), pp. 1–48.
[3] V. Barthelmann, E. Novak, and K. Ritter, *High dimensional polynomial interpolation on sparse grids*, Adv. Comput. Math., 12 (2000), pp. 273–288.
[4] M. Bauer, M. Bebendorf, and B. Feist, *Kernel-Independent Adaptive Construction of $\mathcal{H}^2$-Matrix Approximations*, preprint, arXiv:2006.01556, 2020, https://arxiv.org/abs/2006.01556.
[5] M. Bebendorf, *Approximation of boundary element matrices*, Numer. Math., 86 (2000), pp. 565–589.
[6] M. Bebendorf, *Adaptive cross approximation of multivariate functions*, Constr. Approx., 34 (2011), pp. 149–179.
[7] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, 2006.
[8] D. Cai, E. Chow, L. Erlandson, Y. Saad, and Y. Xi, *Smash: Structured matrix approximation by separation and hierarchy*, Numer. Linear Algebra Appl., 25 (2018), e2204.
[9] D. Cai and P. S. Vassilevski, *Eigenvalue problems for exponential-type kernels*, Comput. Methods Appl. Math., 20 (2020), pp. 61–78.
[10] D. Decoste and B. Schölkopf, *Training invariant support vector machines*, Mach. Learn., 46 (2002), pp. 161–190.
[11] J. Dick and F. Pillichshammer, *Digital nets and sequences: Discrepancy theory and quasi–Monte Carlo integration*, Cambridge University Press, Cambridge, UK, 2010.
[12] P. Drineas and M. W. Mahoney, *On the Nyström method for approximating a Gram matrix for improved kernel-based learning*, J. Mach. Learn. Res., 6 (2005), pp. 2153–2175.
[13] Y. Eldar, M. Lindenbaum, M. Porat, and Y. Zeevi, *The farthest point strategy for progressive image sampling*, IEEE Trans. Image Process., 6 (1997), pp. 1305–1315.
[14] M. Fanuel, J. Schreurs, and J. Suykens, *Diversity sampling is an implicit regularization for kernel methods*, SIAM J. Math. Data Sci., 3 (2021), pp. 280–297.
[15] M. Fanuel, J. Schreurs, and J. A. Suykens, *Determinantal Point Processes Implicitly Regularize Semi-Parametric Regression Problems*, preprint, arXiv:2011.06964, 2020.
[16] B. Fornberg and G. Wright, *Stable computation of multiquadric interpolants for all values of the shape parameter*, Comput. Math. Appl., 48 (2004), pp. 853–867.
[17] A. Gittens and M. W. Mahoney, *Revisiting the Nyström method for improved large-scale machine learning*, J. Mach. Learn. Res., 17 (2016), pp. 3977–4041.
[18] B. Haasdonk, *Feature space interpretation of SVMS with indefinite kernels*, IEEE Trans. Pattern Anal. Mach. Intell., 27 (2005), pp. 482–492.
[19] B. Haasdonk and D. Keysers, *Tangent distance kernels for support vector machines*, in Proceedings of the 2002 International Conference on Pattern Recognition, IEEE, 2002, pp. 864–868.
[20] W. Hackbusch, *Hierarchical Matrices: Algorithms and Analysis*, Springer Ser. Comput. Math. 49, Springer, Berlin, 2015.
[21] W. Hackbusch and Z. P. Nowak, *On the fast matrix multiplication in the boundary element method by panel clustering*, Numer. Math., 54 (1989), pp. 463–491.

[22] J. H. Halton, *Algorithm* 247: *Radical-inverse quasi-random point sequence*, Comm. ACM, 7 (1964), pp. 701–702.
[23] H. Harbrecht, M. Peters, and R. Schneider, *On the low-rank approximation by the pivoted Cholesky decomposition*, Appl. Numer. Math., 62 (2012), pp. 428–440.
[24] L. Kuipers and H. Niederreiter, *Uniform distribution of sequences*, Dover, New York, 2012.
[25] A. Kulesza and B. Taskar, *k-dpps: Fixed-size determinantal point processes*, in Proceedings of the 28th International Conference on Machine Learning, Omnipress, 2011.
[26] S. Kumar, M. Mohri, and A. Talwalkar, *On sampling-based approximate spectral decomposition*, in Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 2009, pp. 553–560.
[27] S. Kumar, M. Mohri, and A. Talwalkar, *Sampling methods for the Nyström method*, J. Mach. Learn. Res., 13 (2012), pp. 981–1006.
[28] Q. Le, T. Sarlós, and A. Smola, *Fastfood-approximating kernel expansions in loglinear time*, in Proceedings of the 30th International Conference on Machine Learning, Vol. 85, JMLR, 2013.
[29] M. W. Mahoney and P. Drineas, *Cur matrix decompositions for improved data analysis*, Proc. Natl. Acad. Sci. USA, 106 (2009), pp. 697–702.
[30] P. Moreno, P. Ho, and N. Vasconcelos, *A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications*, in Advances in Neural Information Processing Systems 16, MIT Press, Cambridge, MA, 2003, pp. 1385–1392.
[31] W. J. Morokoff and R. E. Caflisch, *Quasi-random sequences and their discrepancies*, SIAM J. Sci. Comput., 15 (1994), pp. 1251–1279.
[32] C. Musco and C. Musco, *Recursive sampling for the Nyström method*, in Advances in Neural Information Processing Systems 30, Curran Associates, Red Hook, NY, 2017, pp. 3833–3845.
[33] Y. Nakatsukasa, *Fast and Stable Randomized Low-Rank Matrix Approximation*, preprint, arXiv:2009.11392, 2020.
[34] H. Niederreiter, *Point sets and sequences with small discrepancy*, Monatsh. Math., 104 (1987), pp. 273–337.
[35] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia, 1992.
[36] C. S. Ong, X. Mary, S. Canu, and A. J. Smola, *Learning with non-positive kernels*, in Proceedings of the Twenty-First International Conference on Machine Learning, ACM, 2004, p. 81.
[37] A. Rahimi and B. Recht, *Random features for large-scale kernel machines*, in Advances in Neural Information Processing Systems 20, Curran Associates, Red Hook, NY, 2007, pp. 1177–1184.
[38] A. Rahimi and B. Recht, *Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning*, in Advances in Neural Information Processing Systems 21, Curran Associates, Red Hook, NY, 2008, pp. 1313–1320.
[39] A. Rudi, D. Calandriello, L. Carratino, and L. Rosasco, *On fast leverage score sampling and optimal learning*, in Advances in Neural Information Processing Systems 31, Curran Associates, Red Hook, NY, 2018, pp. 5672–5682.
[40] J. Schneider, *Error estimates for two-dimensional cross approximation*, J. Approx. Theory, 162 (2010), pp. 1685–1700.
[41] I. M. Sobol, *Multidimensional Quadrature Formulas and Haar Functions*, Nauka, Moscow, 1969.
[42] A. Townsend and L. Trefethen, *Continuous analogues of matrix factorizations*, Proc. A, 471 (2015), 2014585.
[43] V. Vapnik, *The nature of Statistical Learning Theory*, Springer, New York, 2013.
[44] C. K. Williams and M. Seeger, *Using the Nyström method to speed up kernel machines*, in Advances in Neural Information Processing Systems 14, MIT Press, Cambridge, MA, 2001, pp. 682–688.
[45] K. Zhang and J. T. Kwok, *Clustered Nyström method for large scale manifold learning and dimension reduction*, IEEE Trans. Neural Netw., 21 (2010), pp. 1576–1587.
[46] K. Zhang, I. W. Tsang, and J. T. Kwok, *Improved Nyström low-rank approximation and error analysis*, in Proceedings of the 25th International Conference on Machine Learning, ACM, 2008, pp. 1232–1239.