PRELIMINARY PREPRINT VERSION: DO NOT CITE
The AAAI Digital Library will contain the published
version some time after the conference.

Distributed Randomized Sketching Kernel Learning

Rong Yin, [1,2] Yong Liu, [3,4*] Dan Meng[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
[4] Beijing Key Laboratory of Big Data Management and Analysis Methods, Beijing, China
yinrong@iie.ac.cn, liuyonggsai@ruc.edu.cn, mengdan@iie.ac.cn

Abstract

We investigate the statistical and computational requirements for distributed kernel ridge regression with randomized sketching (DKRR-RS) and successfully achieve the optimal learning rates with only a fraction of computations. More precisely, the proposed DKRR-RS combines sparse randomized sketching, divide-and-conquer and KRR to scale up kernel methods and successfully derives the same learning rate as the exact KRR with only $\mathcal{O}(N^{0.5+\gamma})$ time in expectation, at the basic setting, which outperforms previous state of the art solutions, where N is the number of data and $\gamma \in [0, 1]$. Then, for the sake of the gap between theory and experiments, we derive the optimal learning rate in probability for DKRR-RS to reflect its generalization performance. Finally, to further improve the learning performance, we construct an efficient communication strategy for DKRR-RS and demonstrate the power of communications via theoretical assessment. An extensive experiment validates the effectiveness of DKRR-RS and the communication strategy on real-world datasets.

1 Introduction

Kernel methods have been widely used in data mining, machine learning, and other fields (Yin et al. 2020c; Saunders, Gammerman, and Vovk 1998; Liu 2021; Li and Liu 2021b; Yin et al. 2020b; Li and Liu 2021a; Wang et al. 2021; Li et al. 2018). However, they are unfeasible to deal with large-scale scenarios due to the high computational requirements, typically at least quadratic in the number of data.

To address these issues, a variety of approximate kernel ridge regression (KRR) are proposed. The main principle is to characterize statistical and computational trade-offs, that is, to sacrifice statistical accuracy to gain computational benefits. The representative methods include Nyström (Yin et al. 2021, 2020a; Rudi, Carratino, and Rosasco 2017; Li, Kwok, and Lu 2010) which constructs the approximate kernel matrix with a few anchor points, random features (Liu, Liu, and Wang 2021; Li, Liu, and Wang 2019; Rudi, Camoriano, and Rosasco 2016; Rahimi and Recht 2007), iterative optimization (Lin and Cevher 2020; Lin, Lei, and Zhou 2019; Shalev Shwartz et al. 2011), distributed learning (Liu, Liu, and Wang 2021; Lin, Wang, and Zhou 2020; Wang 2019; Guo, Lin, and Shi 2019; Chang, Lin, and Zhou 2017; Lin, Guo, and Zhou 2017; Zhang, Duchi, and Wainwright 2015, 2013) which divides the training data into some subsets for processing on local processors and carry out necessary communications, and randomized sketching (Lin and Cevher 2020; Lian, Liu, and Fan 2021; Liu, Shang, and Cheng 2019; Yang, Pilanci, and Wainwright 2017) which projects the kernel matrix into a small one based on the sketch matrix. The above studies show that randomized sketching and distributed learning have outstanding effects in kernel methods. Recently, combinations of those accelerated algorithms benefit a lot and attract much attention, of which learning properties have been explored including the combination of Nyström and iterative optimization (Rudi, Carratino, and Rosasco 2017), randomized sketching and iterative optimization (Lin and Cevher 2020), and divide-and-conquer and multi-pass SGD (Lin and Cevher 2018). However, the computational requirements of the existing approximate KRR estimates are still high, and it still remains unclear for complexity reduction and statistical analysis to the combination of randomized sketching and distributed learning in kernel learning.

To further overcome the computational bottlenecks, we propose the method, called DKRR-RS, of combining divide-and-conquer learning and a sparse randomized sketching method for KRR, which keeps the optimal learning rate. This paper makes the following main contributions. 1) The proposed DKRR-RS improves the existing state-of-the-art results of the distributed learning and randomized sketching. In particular, at the basic setting ($2\zeta + \gamma = 2$) and the basic assumptions (see section 5.1 for details), DKRR-RS requires only $\mathcal{O}(N^{0.5+\gamma})$ time and $\mathcal{O}(N)$ space in expectation to guarantee the same learning rate as the exact KRR, where N is the number of data, $\zeta \in [1/2, 1]$, and $\gamma \in [0, 1]$. We take a substantial step in provably reducing the computational requirements by combining randomized sketching with distributed learning; 2) The theoretical performance in expectation reflects the average results of enough trials, but may fail to capture the learning performance for a single trial. To essentially reflect the generalization performance, we obtain the optimal learning rate of DKRR-RS in probability and numerically verify its effectiveness; 3) We propose a communication strategy to further improve the learning performance of DKRR-RS, called DKRR-RS-CM, and

*Corresponding author: Yong Liu
Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

validate the theoretical bounds via numerical experiments.

The rest of the paper is organized as follows. Section 2 is the related work. Section 3 and 4 introduce the preliminaries and the proposed methods of DKRR-RS and DKRR-RS-CM. The theoretical results are shown in section 5. The following are experiments and conclusions.

2 Related Work

The optimal learning rate of randomized sketching in KRR is first proposed in (Yang, Pilanci, and Wainwright 2017), which utilizes the fast Fourier transform to accelerate the product of matrices in random orthogonal system sketches (ROS). However, due to the dense sketch matrices including the discrete Fourier transform matrix and the Hadamard matrix, $\mathcal{O}(N^2 + Nm^2)$ time and $\mathcal{O}(N^2)$ space are needed for the optimal learning rate with the sketch dimension $m = \Omega(d_N \log^4(N))$, where $d_N > 1$. Subsequently, the randomized sketching method in (Liu, Shang, and Cheng 2019), called Gauss, also achieves the optimal learning rate with time $\mathcal{O}(N^2 m)$ and space $\mathcal{O}(N^2)$, based on dense Gaussian sketch matrix and local Rademacher framework, when $m = \Omega(d_N)$. Recently, Lin and Cevher (Lin and Cevher 2020) construct a new randomized sketching method called Subgauss. The entries of the sketch matrix in Subgauss are i.i.d. Subgaussian (such as Gaussian or Bernoulli). Lin and Cevher firstly introduce the integral operator framework to derive the the optimal learning rate with time $\mathcal{O}(N^2 m)$ and space $\mathcal{O}(N^2)$, when $m = \Omega(N^{\frac{\gamma}{2\zeta+\gamma}})$, where $\frac{\gamma}{2\zeta+\gamma} > 0$. Heng Lian et al. (Lian, Liu, and Fan 2021) combined divide-and-conquer and random sketching based on two well-known types of dense sketching matrix, the Gaussian sketch and the ROS sketch. The dense sketching matrices lead to high time and space complexity as mentioned above. The condition of theoretical analysis in (Lian, Liu, and Fan 2021) is different from this paper. By comparison with them, DKRR-RS keeps less time and space complexity with the optimal generalization error under the same condition.

The representative distributed KRR includes DKRR (Guo, Lin, and Shi 2019; Chang, Lin, and Zhou 2017; Lin, Guo, and Zhou 2017; Zhang, Duchi, and Wainwright 2015, 2013) based on divide-and-conquer, DKRR-RF (Li, Liu, and Wang 2019) based on DKRR and random features (Rudi, Camoriano, and Rosasco 2016), and DKRR-NY-PCG (Yin et al. 2020a) based on DKRR and Nyström-PCG (Rudi, Carratino, and Rosasco 2017), which derive the optimal learning rate in expectation. However, they have a restricted limitation in the number of local processors p, that is, to derive the optimal learning rate, p should be restricted to a constant at the most popular case ($\zeta = 1/2, \gamma = 1$). Subsequently, Yin et al.(Yin et al. 2021), Liu et al. (Liu, Liu, and Wang 2021), and Lin et al. (Lin, Wang, and Zhou 2020) introduce communication strategies into DKRR and DKRR-RF to relax the restriction on p. However, in (Lin, Wang, and Zhou 2020), they require communicating the input data between each local processor, which brings difficulties to the protection of data privacy. In addition, the high communication complexity $\mathcal{O}(Nd)$ at each iteration makes it infeasible in practice for large-scale datasets, where d is the dimension.

3 Preliminaries

Let $\mathcal{X}$ be the input space and $\mathbb{R}$ be the output space. The training set $D = \cup_{j=1}^{p} D_j = \{(x_i, y_i)\}_{i=1}^{N}$ is sampled identically and independently from $\mathcal{X} \times \mathbb{R}$ with respect to ρ, where ρ be a fixed but unknown distribution and $p > 1$. All subsets $\{D_j\}_{j=1}^{p}$ are disjoint and $|D_1| = \ldots = |D_p| = n$. Let $\rho_X(\cdot)$ be the induced marginal measure on $\mathcal{X}$ of ρ, $\rho(\cdot|x)$ be the conditional probability measure on $\mathbb{R}$ with respect to $x \in \mathcal{X}$ and ρ. We denote by K_x the function $K(x, \cdot)$ and by $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}})$ the Hilbert space of functions with the associated inner product induced by K, defined by $\langle K_x, K_{x'}\rangle_{\mathcal{H}} = K(x, x'), \forall x, x' \in \mathcal{X}$. Let $L^2_{\rho_X}$ be the Lebesgue space of ρ_X square integrable functions, endowed with the inner product $\langle\phi, \psi\rangle_{\rho_X} = \int \phi(x)\psi(x)d\rho_X(x)$, $\forall \phi, \psi \in L^2_{\rho_X}$, and norm $\|\psi\|_\rho = \sqrt{\langle\psi, \psi\rangle}_{\rho_X}$ for all $\psi \in L^2_{\rho_X}$. For any $f \in \mathcal{H}$, $\phi \in L^2_{\rho_X}$, define a linear map $S : \mathcal{H} \to L^2_{\rho_X}$, such that $Sf = \langle f, K_{(\cdot)}\rangle_{\mathcal{H}} \in L^2_{\rho_X}$, with adjoint $S^* : L^2_{\rho_X} \to \mathcal{H}$, such that $S^*\phi = \int \phi(x)K_x d\rho_X(x) \in \mathcal{H}$. $\mathcal{L}$: $L^2_{\rho_X} \to L^2_{\rho_X}$, such that $\mathcal{L} = SS^*$ and $\mathcal{T} : \mathcal{H} \to \mathcal{H}$, such that $\mathcal{T} = S^*S$. For any $\upsilon \in \mathbb{R}^n$, define $S_n : \mathcal{H} \to \mathbb{R}^n$, such that $S_n f = \frac{1}{\sqrt{n}}(\langle f, K_{x_i}\rangle)_{i=1}^{n} \in \mathbb{R}^n$, with adjoint $S_n^* : \mathbb{R}^n \to \mathcal{H}$, such that $S_n^*\upsilon = \frac{1}{\sqrt{n}}\Sigma_{i=1}^{n}\upsilon_i K_{x_i} \in \mathcal{H}$. $\mathcal{T}_n : \mathcal{H} \to \mathcal{H}$, such that $\mathcal{T}_n = S_n^* S_n$.

3.1 Kernel Ridge Regression (KRR)

Given a hypothesis space $\mathcal{H}$ of measurable functions from $\mathcal{X}$ to $\mathbb{R}$, the goal of the supervised learning problem can be formalized as minimizing the expected risk

$$\inf_{f\in\mathcal{H}} \mathcal{E}(f), \quad \mathcal{E}(f) = \int_{\mathcal{X}\times\mathbb{R}} (f(x) - y)^2 d\rho(x, y). \quad (1)$$

Define the regression function (Steinwart and Christmann 2008) that minimizes the expected risk over all measurable functions as $f_\rho(x) = \int y d\rho(y|x)$, almost everywhere.

A good empirical solution $\hat{f}$ should correspond to the small excess risk $\mathcal{E}(\hat{f}) - \inf_{f\in\mathcal{H}} \mathcal{E}(f)$. Suppose there is such an $f_{\mathcal{H}} \in \mathcal{H}$: $\mathcal{E}(f_{\mathcal{H}}) = \min_{f\in\mathcal{H}} \mathcal{E}(f)$. This paper focuses on KRR. Given a Mercer kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, KRR can be state as

$$\hat{f}_{D,\lambda} = \arg\min_{f\in\mathcal{H}} \frac{1}{N}\sum_{i=1}^{N}(f(x_i) - y_i)^2 + \lambda\|f\|^2_{\mathcal{H}}, \lambda > 0. \quad (2)$$

There is a unique closed form solution in Eq.(2) according to the Representer Theorem (Schölkopf, Herbrich, and Smola 2001)

$$\hat{f}_{D,\lambda}(x) = \sum_{i=1}^{N} \hat{\alpha}_i K(x_i, x), \text{ with } \hat{\boldsymbol{\alpha}} = (\mathbf{K}_N + \lambda N\mathbf{I})^{-1}\mathbf{y}, \quad (3)$$

where $\mathbf{K}_N$ is the $N \times N$ kernel matrix with $\mathbf{K}_N(i, j) = K(x_i, x_j)$, $\mathbf{y} = \mathbf{y}_N = [y_1, \ldots, y_N]^T$.

The time $\mathcal{O}(N^3)$ for solving the linear system in Eq.(3) and the memory $\mathcal{O}(N^2)$ for storing the kernel matrix $\mathbf{K}_N$ are computationally unfeasible in large-scale setting.

3.2 KRR with Divide-and-Conquer (DKRR)

KRR with divide-and-conquer (DKRR) is defined as

$$\hat{f}_{D,\lambda} = \frac{1}{p}\sum_{j=1}^{p} \hat{f}_{D_j,\lambda}, \tag{4}$$

where $\hat{f}_{D_j,\lambda}$ is the solution in Eq.(3). Its time complexity and space complexity are $\mathcal{O}(N^3/p^3)$ and $\mathcal{O}(N^2/p^2)$.

4 The Proposed Algorithms

4.1 DKRR with Randomized Sketching (DKRR-RS)

We propose a novel randomized sketching into DKRR in Eq.(4), called DKRR-RS. The randomized sketching is based on the sparse sketch matrix $\mathbf{R} \in \mathbb{R}^{m\times n}$, where the sketch dimension $m < n$. Let $\sigma(i) \in \{-1/m, +1/m\}$ be 2-wise independent hash function and $\sigma(i) = t$ for $t \in \{-1/m, +1/m\}$ with probability $1/2$. The entries $\mathbf{R}_{i,j}$ of $\mathbf{R}$ is designed as

$$\mathbf{R}_{i,j} = \begin{cases} \sigma(i), & \text{with prob. } \dfrac{m}{n}, \\ 0, & \text{with prob. } 1-\dfrac{m}{n}. \end{cases} \tag{5}$$

Note that the novel sketch matrix $\mathbf{R}$ is sparse. One only needs to store the non-zero elements. Moreover, in a matrix-matrix product (say of the form $\mathbf{R}\times\mathbf{A}$ for some matrix $\mathbf{A}$), one can achieve a significant n/m-fold speedup compared with a dense matrix.

Projecting the kernel matrix $\mathbf{K}_n$ by the sketch matrix $\mathbf{R}$. That is, one can restrict the solver of approximate KRR to the hypothesis space, $\mathcal{H}_m = \{f | f = \sum_{i=1}^{n}(\mathbf{R}^T\hat{\boldsymbol{\alpha}}_j)_i K(x_i,\cdot), \hat{\boldsymbol{\alpha}}_j \in \mathbb{R}^m\}$.

Define $f_{D_j,m,\lambda}$ be the local estimator for j-th subset D_j in DKRR-RS. Therefore, the following problem

$$f_{D_j,m,\lambda} = \arg\min_{f\in\mathcal{H}_m} \frac{1}{n}\sum_{i=1}^{n}(f(x_i)-y_i)^2 + \lambda\|f\|_{\mathcal{H}}^2, \tag{6}$$

can be transformed into

$$f_{D_j,m,\lambda}(x) = \sum_{i=1}^{n}(\mathbf{R}^T\hat{\boldsymbol{\alpha}}_j)_i K(x_i,x), \tag{7}$$

with

$$\hat{\boldsymbol{\alpha}}_j = (\mathbf{R}\mathbf{K}_n^2\mathbf{R}^T + \lambda n\mathbf{R}\mathbf{K}_n\mathbf{R}^T)^{-1}(\mathbf{R}\mathbf{K}_n\mathbf{y}_n). \tag{8}$$

Equivalently, $f_{D_j,m,\lambda}$ is characterized by the following equation

$$(P_m\mathcal{T}_nP_m + \lambda I)f_{D_j,m,\lambda} = P_mS_n^*\hat{\mathbf{y}}_n, \tag{9}$$

where P_m be the projection operator with range $\mathcal{H}_m$ and $\hat{\mathbf{y}}_n = \frac{1}{\sqrt{n}}\mathbf{y}_n$ (Rudi, Camoriano, and Rosasco 2015).

The global estimator can be obtained by the weighted average of approximate local estimators

$$\bar{f}^0_{D,m,\lambda} = \frac{1}{p}\sum_{j=1}^{p} f_{D_j,m,\lambda}. \tag{10}$$

The prediction stage is based on the approximate parameters $\hat{\boldsymbol{\alpha}}_j$ of each local processor and we obtain the final prediction estimator by averaging the local estimators of each local processor.

Complexity Analysis of DKRR-RS *Time Complexity*: Due to the sparsity of the sketch matrix $\mathbf{R}$, only the non-zero elements need to be multiplied instead of each element in the matrix-matrix product. Therefore, the cost of computing $\mathbf{R}\mathbf{K}_n$ is only $\mathcal{O}(Nm^2/p)$. The matrix $\mathbf{K}_n^2$ does not need to be represented explicitly. $\mathbf{R}\mathbf{K}_n^2\mathbf{R}$ can be converted to the form of $\mathbf{R}\mathbf{K}_n(\mathbf{R}\mathbf{K}_n)^T$, whose time complexity is $\mathcal{O}(Nm^2/p)$ except for $\mathbf{R}\mathbf{K}_n$. Taking into account the fragmented time, the time complexity of DKRR-RS can be summed up as $\mathcal{O}(Nm^2/p)$. *Space Complexity*: The approximate kernel matrix $\mathbf{K}_n$ is the decisive part in the memory consumption, whose space complexity is $\mathcal{O}(N^2/p^2)$. *Communication Complexity*: It is $\mathcal{O}(m)$ in DKRR-RS.

4.2 DKRR-RS with Communications (DKRR-RS-CM)

For further improving the learning performance of approximate KRR, we propose a novel communication-based DKRR-RS, called DKRR-RS-CM. The efficient communication strategy can enlarge the range of partition p with guaranteeing the optimal statistical performance of distributed KRR, which is adapted from (Lin, Wang, and Zhou 2020) and avoids local data communication.

Let $G_{D,m,\lambda}$ be

$$G_{D,m,\lambda}(f) = (P_m\mathcal{T}_NP_m + \lambda I)f - P_mS_N^*\hat{\mathbf{y}}_N. \tag{11}$$

Note that $G_{D,m,\lambda}(f)$ is the half gradient of the empirical risk of $\arg\min_{f\in\mathcal{H}_m}\frac{1}{N}\sum_{i=1}^{N}(f(x_i)-y_i)^2 + \lambda\|f\|_{\mathcal{H}}^2$ over $\mathcal{H}_m$. According to Eq.(9), it can be found that

$$f_{D,m,\lambda} = (P_m\mathcal{T}_NP_m + \lambda I)^{-1}P_mS_N^*\hat{\mathbf{y}}_N, \tag{12}$$

and

$$\bar{f}^0_{D,m,\lambda} = \frac{1}{p}\sum_{j=1}^{p}(P_m\mathcal{T}_nP_m + \lambda I)^{-1}P_mS_n^*\hat{\mathbf{y}}_n. \tag{13}$$

Therefore, for any $f \in \mathcal{H}$, we have

$$\begin{aligned} f_{D,m,\lambda} =& f - (P_m\mathcal{T}_NP_m + \lambda I)^{-1} \\ &\times\left[(P_m\mathcal{T}_NP_m + \lambda I)f - P_mS_N^*\hat{\mathbf{y}}_N\right], \end{aligned} \tag{14}$$

and

$$\begin{aligned} \bar{f}^0_{D,m,\lambda} =& f - \frac{1}{p}\sum_{j=1}^{p}(P_m\mathcal{T}_nP_m + \lambda I)^{-1} \\ &\times\left[(P_m\mathcal{T}_nP_m + \lambda I)f - P_mS_n^*\hat{\mathbf{y}}_n\right]. \end{aligned} \tag{15}$$

Comparing Eq.(14) and Eq.(15), and noting that the global gradient can be obtained via the communications of each local gradient, i.e., $G_{D,m,\lambda}(f) = \frac{1}{p}\sum_{j=1}^{p}G_{D_j,m,\lambda}(f)$, we consider the communication strategy of Newton-Raphson iteration-based:

$$\bar{f}^l_{D,m,\lambda} = \bar{f}^{l-1}_{D,m,\lambda} - \frac{1}{p}\sum_{j=1}^{p}\beta_j^{l-1}, l > 0, \tag{16}$$

where $\beta_j^{l-1} = (P_m\mathcal{T}_nP_m + \lambda I)^{-1}G_{D,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda})$.

Algorithm 1: DKRR-RS with Communications (DKRR-RS-CM)

Input: p disjoint subsets $\{D_j\}_{j=1}^p$ with $D = \cup_{j=1}^p D_j$, kernel parameter, regularization parameter λ, sketch dimension m.

Output: $\bar{f}_{D,m,\lambda}^M$

1: **If** $l = 0$

Local processor: compute $f_{D_j,m,\lambda}$ in Eq.(7), and communicate back to global processor.

Global processor: compute $\bar{f}_{D,m,\lambda}^0$ in Eq.(10), and communicate to each local processor.

2: **End If**

3: **For** $l = 1$ **to** M **do**

Local processor: compute local gradient $G_{D_j,m,\lambda}(\bar{f}_{D,m,\lambda}^{l-1})$ and communicate back to the global processor.

Global processor: compute global gradient $G_{D,m,\lambda}(\bar{f}_{D,m,\lambda}^{l-1}) = \frac{1}{p}\sum_{j=1}^p G_{D_j,m,\lambda}(\bar{f}_{D,m,\lambda}^{l-1})$, and communicate to each local processor.

Local processor: compute $\beta_j^{l-1} = (P_m \mathcal{T}_n P_m + \lambda I)^{-1} G_{D,m,\lambda}(\bar{f}_{D,m,\lambda}^{l-1})$ and communicate back to the global processor.

Global processor: compute $\bar{f}_{D,m,\lambda}^l$ in Eq.(16), and communicate to each local processor.

4: **End For**

The process of DKRR-RS-CM is summarized in Algorithm 1. When $l = 0$, DKRR-RS-CM executes the computation of DKRR-RS. Otherwise, the four-step communication strategy is implemented. The first step is to compute the local gradients in each local processor and communicate them back to the global processor. The second step is to compute the global gradient by the local gradients in the global processor and communicate it to each local processor. The following is to compute β_j^{l-1} by the global gradient in each local processor and communicate them back to the global processor. The fourth is to compute $\bar{f}_{D,m,\lambda}^l$ in the global processor and communicate it to each local processor. Here, one iteration ends. Repeat the four-step communication strategy until $l = M$ and output $\bar{f}_{D,m,\lambda}^M$. The testing flow is shown in Appendix.

Complexity Analysis of DKRR-RS-CM *Time Complexity*: For each local processor, the matrices product $\mathbf{RK}_n$ and $(\mathbf{RK}_n)(\mathbf{RK}_n)^T$ and the inverse of $\mathbf{RK}_n(\mathbf{RK}_n)^T + \lambda n \mathbf{RK}_n \mathbf{R}^T$ need to be computed once. In each iteration, one need to compute the local gradient $G_{D_j,m,\lambda}(\bar{f}_{D,m,\lambda}^{l-1})$ and β_j^{l-1} in each local processor. Therefore, the total time complexity is $\mathcal{O}(m^2N/p + MmN/p)$ for each local processor. *Space Complexity*: The key element in memory is the matrix $\mathbf{K}_n$. Therefore, the space complexity in each local processor is $\mathcal{O}(N^2/p^2)$. *Communication Complexity*: For $l = 0$, the global processor and each local processor need to communicate $f_{D_j,m,\lambda}$ and $\bar{f}_{D,m,\lambda}^0$. In each iteration, the local processors receive $G_{D,m,\lambda}$ and $\bar{f}_{D,m,\lambda}^l$, and communicate $G_{D_j,m,\lambda}$ and β_j^{l-1} back to the global processor. Therefore, the communication complexity is $\mathcal{O}(Mm)$.

5 Theoretical Analysis

In this section, we characterize the generalization performances of DKRR-RS and DKRR-RS-CM showing they achieve the same optimal learning rate as KRR, with dramatically reduced computations.

5.1 Basic Assumptions

Assumption 1 (**Moment Assumption**). *There exist positive constants Q and b such that for all $j \geq 2$ with $j \in \mathbb{N}$, $\int_{\mathbb{R}} |y|^j d\rho(y|x) \leq \frac{1}{2} j! b^{j-2} Q^2$, almost everywhere on $\mathcal{X}$.*

Typically, this assumption is related to a noise assumption in the regression model, which is satisfied if y is bounded almost surely (Rudi, Carratino, and Rosasco 2017).

Assumption 2 (**Regularity Assumption**). *$f_{\mathcal{H}}$ satisfies $\int (f_{\mathcal{H}}(x) - f_\rho(x))^2 K_x \otimes K_x d\rho_X(x) \preceq B^2 \mathcal{T}$, and the following Hölder source condition*

$$f_{\mathcal{H}} = \mathcal{L}^\zeta g_0, \textit{with } \|g_0\|_\rho \leq R. \tag{17}$$

Here, B, R are non-negative numbers, $\zeta \in [1/2, 1]$.

The equation Eq.(17) reflects the regularity of the function $f_{\mathcal{H}}$ (Smale and Zhou 2007). The bigger ζ is, the higher the regularity of $f_{\mathcal{H}}$ is and the faster the convergence rate is. $\otimes$ denotes the tensor product.

Assumption 3 (**Capacity Assumption**). *For some $\gamma \in [0, 1]$ and $c_\gamma > 0$, $\mathcal{T}$ satisfies*

$$tr(\mathcal{T}(\mathcal{T} + \lambda I)^{-1}) \leq c_\gamma \lambda^{-\gamma}, \textit{ for all } \lambda > 0. \tag{18}$$

This assumption controls the variance of the estimator and is equivalent to the classic entropy and covering number conditions (Steinwart and Christmann 2008). The effective dimension, the left hand-side of Eq.(18), is often used to measure the complexity of the hypothesis space $\mathcal{H}$ (Caponnetto and Vito 2007). γ reflects the size of $\mathcal{H}$. This assumption is always true for $\gamma = 1$, since $\mathcal{T}$ is a trace class operator. It is satisfied, e.g., if the eigenvalues of $\mathcal{T}$ satisfy a polynomial decaying condition $\sigma_i \sim i^{-1/\gamma}$, or with $\gamma = 0$ if $\mathcal{T}$ is finite rank.

5.2 Optimal Learning Rate for DKRR-RS in Expectation

Theorem 1. *Under Assumptions 1-3 and the sketch matrix $\mathbf{R}$ constructed by Eq.(5), let $\delta \in (0, 1]$, $\gamma \in [0, 1]$, $\zeta \in [1/2, 1]$, and $\bar{f}_{D,m,\lambda}^0$ be the estimator. When $\lambda = \Omega(N^{-\frac{1}{2\zeta+\gamma}})$, $m = \Omega\left(N^{\frac{\gamma}{2\zeta+\gamma}}\right)$, and $p = \mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{2\zeta+\gamma}}\right)$, with probability at least $1 - \delta$, we have*

$$\mathbb{E}\left\|\bar{f}_{D,m,\lambda}^0 - f_{\mathcal{H}}\right\|_\rho^2 = \mathcal{O}\left(N^{-\frac{2\zeta}{2\zeta+\gamma}}\right).$$

Remark 1. *Note that $\mathbb{E}\left[\mathcal{E}(\bar{f}_{D,m,\lambda}^0)\right] - \inf_{f\in\mathcal{H}} \mathcal{E}(f) = \mathbb{E}\left\|\bar{f}_{D,m,\lambda}^0 - f_{\mathcal{H}}\right\|_\rho^2$ (Smale and Zhou 2007). From a theoretical perspective, Theorem 1 shows that if the sketch dimension $m = \Omega\left(N^{\frac{\gamma}{2\zeta+\gamma}}\right)$ and the number of partitions*

$p = \mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{2\zeta+\gamma}}\right)$, *DKRR-RS achieves the same optimal learning rate* $\mathcal{O}\left(N^{-\frac{2\zeta}{2\zeta+\gamma}}\right)$ [1] *as the exact KRR (Lin and Cevher 2020; Caponnetto and Vito 2007). At the basic setting* ($2\zeta+\gamma=2$)*, the time and space complexity of DKRR-RS are* $\mathcal{O}(N^{0.5+\gamma})$ *and* $\mathcal{O}(N)$ *with the optimal learning rate. The most popular case* ($\zeta=1/2$, $\gamma=1$) *and the worst case* ($\zeta=1$, $\gamma=0$) *are included in the basic setting. In the case* $\gamma=0$*, the time cost is only* $\mathcal{O}(N^{0.5})$*, which is a substantial step in scaling up kernel methods.*

Remark 2. *Optimal learning rates for DKRR (Lin, Guo, and Zhou 2017; Guo, Lin, and Shi 2019), DKRR-NY-PCG (Yin et al. 2020a), and DKRR-RF (Li, Liu, and Wang 2019) in expectation have been established. However, they have a strict restriction on the number of local processors* p*. More precisely, at the most popular case, to reach the optimal learning rate,* p *in them should be restricted to a constant* $\mathcal{O}(1)$*, but for our result is* $\mathcal{O}(\sqrt{N})$*. DKRR-RF (Liu, Liu, and Wang 2021) obtains the optimal learning rate with* $p=\mathcal{O}(N^{\frac{2\zeta+\gamma-1}{2\zeta+\gamma}})$ *and* $m=\Omega(N^{\frac{(2\zeta-1)\gamma+1}{2\zeta+\gamma}})$ *in expectation. Compared with DKRR-RF (Liu, Liu, and Wang 2021) in expectation, DKRR-RS reduces the time complexity by a factor of* $N^{\frac{4(\zeta-1)\gamma+2}{2\zeta+\gamma}}$ *with the optimal learning rate, where* $2(\zeta-1)\gamma+1>0$*. Compared to DKRR-NY-PCG (Yin et al. 2020a), DKRR-RS reduces the time complexity and space complexity by factors of* $N^{\frac{1-\gamma}{2\zeta+\gamma}}$ *and* $N^{\frac{\gamma}{2\zeta+\gamma}}$ *with the optimal learning rate, where* $1-\gamma\geq 0$*.*

5.3 Optimal Learning Rate for DKRR-RS in Probability

The expectation in Theorem 1 describes the average properties of multiple trials but may fail to capture the learning performance for a single trial. To essentially reflect the generalization performance of DKRR-RS, we achieve the optimal learning rate in probability.

Theorem 2. *Under Assumptions 1-3 and the sketch matrix* $\mathbf{R}$ *constructed by Eq.(5), let* $\delta\in(0,1]$, $\gamma\in[0,1]$, $\zeta\in[1/2,1]$*, and* $\bar{f}^0_{D,m,\lambda}$ *be the estimator. When* $\lambda=\Omega(N^{-\frac{1}{2\zeta+\gamma}})$, $m=\Omega\left(N^{\frac{\gamma}{2\zeta+\gamma}}\right)$*, and* $p=\mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{4\zeta+2\gamma}}\right)$*, with probability at least* $1-\delta$*, we have*

$$\left\|\bar{f}^0_{D,m,\lambda}-f_{\mathcal{H}}\right\|_\rho^2=\mathcal{O}\left(N^{-\frac{2\zeta}{2\zeta+\gamma}}\right).$$

Remark 3. *DRKK-RS obtains the optimal learning rate not only in expectation but also in probability, and the upper bound* $\mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{4\zeta+2\gamma}}\right)$ *of* p *in Theorem 2 is stricter than* $\mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{2\zeta+\gamma}}\right)$ *in Theorem 1. This is because that the error decomposition in probability is not easy to separate a distributed error that controls* p *compared to the one in expectation. To derive the optimal learning rate, we provide a novel error decomposition in probability, please see the details in Appendix.*

[1] We hide the logarithmic terms of learning rate and complexity in this paper.

5.4 Optimal Learning Rate for DKRR-RS-CM in Probability

This part shows that the proposed communication strategy can improve the learning performance of DKRR-RS, that is, enlarge the range of partition p.

Theorem 3. *Under Assumptions 1-3 and the sketch matrix* $\mathbf{R}$ *constructed by Eq.(5), let* $\delta\in(0,1]$, $\gamma\in[0,1]$, $\zeta\in[1/2,1]$*, and* $\bar{f}^M_{D,m,\lambda}$ *be the estimator. When* $\lambda=\Omega(N^{-\frac{1}{2\zeta+\gamma}})$, $m=\Omega\left(N^{\frac{\gamma}{2\zeta+\gamma}}\right)$*, and* $p=\mathcal{O}\left(N^{\frac{(2\zeta+\gamma-1)(M+1)}{(2\zeta+\gamma)(M+2)}}\right)$*, with probability at least* $1-\delta$*, we have*

$$\left\|\bar{f}^M_{D,m,\lambda}-f_{\mathcal{H}}\right\|_\rho^2=\mathcal{O}\left(N^{-\frac{2\zeta}{2\zeta+\gamma}}\right).$$

Proof. The proof of Theorem 1, 2, and 3 is given in Appendix. □

Remark 4. *It is clear that the upper bound of number of partitions can be relaxed to* $\mathcal{O}\left(N^{\frac{(2\zeta+\gamma-1)(M+1)}{(2\zeta+\gamma)(M+2)}}\right)$ *in Theorem 3 compared to* $\mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{4\zeta+2\gamma}}\right)$ *in Theorem 2 with the optimal learning rate, which demonstrates the function of communication strategy in improving the performance of DKRR-RS. Note that as the number of communication* M *increases, the upper bound of* p *is increasing. When* $M\to\infty$*, the partitions* p *can reach the same bound* $\mathcal{O}\left(N^{\frac{2\zeta+\gamma-1}{2\zeta+\gamma}}\right)$ *in expectation.*

5.5 Compared with the Related Works

Comparisons of Time and Space Complexity Table 1 shows the computational complexity of the state-of-the-art approximate KRR estimates with the same statistical accuracy as the exact KRR at the basic setting. We know that the proposed DKRR-RS only require $N^{0.5+\gamma}$ time and N space with the optimal learning rate in expectation, which is more effective than other methods, where $\gamma\in[0,1]$. DKRR-RS-CM can also keep the least time and communication complexity with the optimal learning rate in probability than the communication-based methods of DKRR-CM (Lin, Wang, and Zhou 2020), DKRR-RF-CM (Liu, Liu, and Wang 2021), and DKRR-NY-CM (Yin et al. 2021). At the same time, the proposed DKRR-RS and DKRR-RS-CM keep the best upper bound of p under the same conditions.

Proof Techniques From a theoretical perspective, this paper is a non-trivial extension of these approximate methods. Compared with (Lin and Cevher 2020): They study stochastic gradient methods and randomized sketching, but we consider the distributed learning and randomized sketching. To obtain the optimal learning rate, we deduce a new decomposition $\|\tilde{f}_{D_j,m,\lambda}-f_{m,\lambda}\|_\rho$ and lead into some techniques to obtain a tight distributed error bound in expectation, and deduce tight bounds of $\|\bar{f}^0_{D,m,\lambda}-f_{D,m,\lambda}\|_\rho$ and $\|\bar{f}^l_{D,m,\lambda}-f_{D,m,\lambda}\|_\rho$ in probability, which are not available in (Lin and Cevher 2020). See Appendix for details.

Compared with (Liu, Liu, and Wang 2021; Yin et al. 2021; Lin, Wang, and Zhou 2020; Yin et al. 2020a; Li, Liu, and

Table 1: Complexity of the state-of-the-art KRR estimates with the same learning rate as the exact KRR at the basic setting. "Comm", "Pro", and "Exp" denote communication complexity, "In probability", and "In expectation". m denotes the sketch dimension in randomized sketching, sampling scale in Nyström and the number of random features in random features methods. N and M are the number of training data and communication. $d > 0$, $\gamma \in [0, 1]$, $\Delta_1 = \frac{(1-\gamma)\gamma}{2} \geq 0$, $\Delta_2 = \frac{\gamma}{2} \in [0, 0.5]$, $0.5 + \Delta_1 \geq \Delta_2$, $d_N > 1$. Logarithmic terms are not showed.

Algorithms	Time	Space	Comm	p	m	Types
KRR (Caponnetto and Vito 2007)	N^3	N^2	/	/	/	Pro
DKRR (Chang, Lin, and Zhou 2017)	N^2	N	$N^{0.5}$	$N^{0.5}$	/	Exp
DKRR (Lin, Wang, and Zhou 2020)	$N^{2.25}$	$N^{1.5}$	$N^{0.75}$	$N^{0.25}$	/	Pro
DKRR-CM (Lin, Wang, and Zhou 2020)	$N^{\frac{3(M+3)}{2M+4}}$	$N^{\frac{M+3}{M+2}}$	MdN	$N^{\frac{M+1}{2M+4}}$	/	Pro
Nyström (Rudi, Camoriano, and Rosasco 2015)	N^2	$N^{1.5}$	/	/	$N^{0.5}$	Pro
Nyström-PCG (Rudi, Carratino, and Rosasco 2017)	$N^{1.5}$	$N^{1.5}$	/	/	$N^{0.5}$	Pro
DKRR-NY-PCG (Yin et al. 2020a)	$N^{1.5}$	$N^{1+\Delta_2}$	$N^{0.5}$	$N^{0.5-\Delta_2}$	$N^{0.5}$	Exp
DKRR-NY-CM (Yin et al. 2021)	$N^{\frac{3M+7}{2M+4}}$	$N^{\frac{2M+5}{2M+4}}$	$MN^{0.5}$	$N^{\frac{M+1}{2M+4}}$	$N^{0.5}$	Pro
Random Features (Rudi, Camoriano, and Rosasco 2016)	$N^{2+2\Delta_1}$	$N^{1.5+\Delta_1}$	/	/	$N^{0.5+\Delta_1}$	Pro
DKRR-RF (Li, Liu, and Wang 2019)	$N^{1.5+2\Delta_1+\Delta_2}$	$N^{1+\Delta_1+\Delta_2}$	$N^{0.5+\Delta_1}$	$N^{0.5-\Delta_2}$	$N^{0.5+\Delta_1}$	Exp
DKRR-RF (Liu, Liu, and Wang 2021)	$N^{1.5+2\Delta_1}$	$N^{1+\Delta_1}$	$N^{0.5+\Delta_1}$	$N^{0.5}$	$N^{0.5+\Delta_1}$	Exp
DKRR-RF (Liu, Liu, and Wang 2021)	$N^{1.75+2\Delta_1}$	$N^{1.25+\Delta_1}$	$N^{0.5+\Delta_1}$	$N^{0.25}$	$N^{0.5+\Delta_1}$	Pro
DKRR-RF-CM (Liu, Liu, and Wang 2021)	$N^{\frac{3M+7}{2M+4}+2\Delta_1}$	$N^{\frac{2M+5}{2M+4}+\Delta_1}$	$MN^{0.5+\Delta_1}$	$N^{\frac{M+1}{2M+4}}$	$N^{0.5+\Delta_1}$	Pro
ROS (Yang, Pilanci, and Wainwright 2017)	$N^2 + Nd_N^2$	N^2	/	/	$d_N \log^4(N)$	Pro
Gauss (Liu, Shang, and Cheng 2019)	$N^2 d_N$	N^2	/	/	d_N	Pro
Subgauss (Lin and Cevher 2020)	N^2	N^2	/	/	N^{Δ_2}	Pro
DKRR-RS (Theorem 1)	$N^{0.5+2\Delta_2}$	N	N^{Δ_2}	$N^{0.5}$	N^{Δ_2}	Exp
DKRR-RS (Theorem 2)	$N^{0.75+2\Delta_2}$	$N^{1.5}$	N^{Δ_2}	$N^{0.25}$	N^{Δ_2}	Pro
DKRR-RS-CM (Theorem 3)	$N^{\frac{M+3}{2M+4}+2\Delta_2}$	$N^{\frac{M+3}{M+2}}$	MN^{Δ_2}	$N^{\frac{M+1}{2M+4}}$	N^{Δ_2}	Pro

Wang 2019): Although they also adopt distributed learning and/or communication strategies, the techniques of proof are different from ours. This is because the distributed errors in this paper are related to the proposed randomized sketching method, which does not exist in them. In addition, by introducing the proof techniques and new operator representations, we relax the restriction on p from $\mathcal{O}(1)$ to $\mathcal{O}(\sqrt{N})$ at the most popular case, and reduce the lower bound of m compared to (Yin et al. 2020a; Li, Liu, and Wang 2019). Finally, in (Lin, Wang, and Zhou 2020), it requires communicating data among each local processor for each iteration, which causes the high communication complexity for large-scale datasets and is difficult to protect the privacy of data in local processors. However, we only require communicating the gradients and model parameters instead of data, which have a smaller communication complexity $\mathcal{O}(m)$ at each iteration and can do better on privacy protection. Meanwhile, we also have a smaller communication and time complexity than (Liu, Liu, and Wang 2021; Yin et al. 2021) due to the smaller lower bound of m so that we are more suitable for large-scale datasets.

Overall, we provide novel distributed bounds with randomized sketching in expectation and probability, and communication-based distributed bound in probability. By introducing some novel techniques and decompositions, we derive the best upper bound of p, which are a non-trivial extension of (Liu, Liu, and Wang 2021; Yin et al. 2021; Lin and Cevher 2020; Lin, Wang, and Zhou 2020; Yin et al. 2020a; Li, Liu, and Wang 2019). Please see the details in Appendix.

6 Experiments

In this section, we present an extensive experiment on the commonly datasets to verify our theoretical predictions.

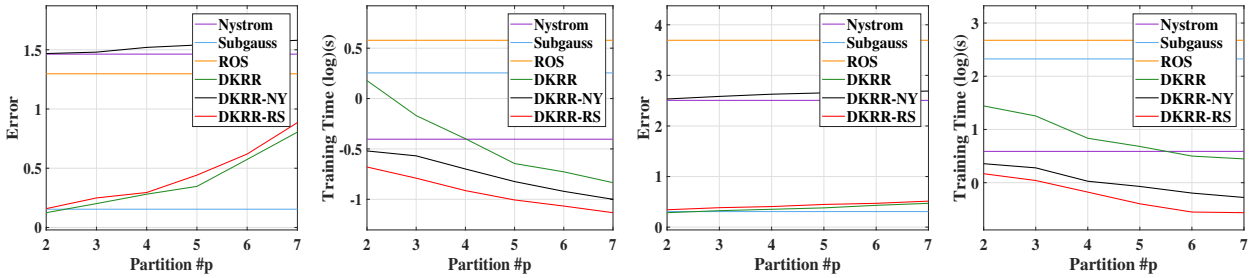
The empirical evaluations of DKRR-RS and DKRR-RS-CM use Gaussian kernel, $e^{-\frac{1}{2h^2}(x_1-x_2)^2}$, on cadata (20640 samples), shuttle (43500 samples), w8a (49749 samples), and connect-4 (67557 samples) datasets[2], where the optimal $h \in 2^{[-2:0.5:5]}$ and $\lambda \in 2^{[-16:3:-4]}$ are selected via 5-fold cross-validation. The datasets are normalized with 70% samples used for training and the rest for testing. The experiments use RMSE and classification error for regression and classification problems and are repeated 5 times with a server of 32 cores (2.40GHz) and 32 GB of RAM.

Figure 1 compare DKRR-RS with Subgauss, ROS[3], DKRR, the classical Nyström[4] (Li, Kwok, and Lu 2010), and DKRR-NY which is a combination of Nyström (Li, Kwok, and Lu 2010) and DKRR, with $m = 600 > \sqrt{N}$. From Figure 1, one can find that DKRR-RS keeps approximate optimal error with the least time. And the larger the

[2]They are from https://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/

[3]The code is from the author of (Yang, Pilanci, and Wainwright 2017)

[4]The code is from (Li, Kwok, and Lu 2010)



(a) cadata (b) cadata (c) connect-4 (d) connect-4

Figure 1: The testing error and training time about the number of partitions p of various algorithms on cadata and connect-4 datasets. For each algorithm, if necessary, $m = 600$.

Table 2: Comparison of average training time (left) in seconds and average testing error (right) in solving KRR between Nyström, Subgauss, ROS and DKRR-RS algorithms on cadata, shuttle, and w8a datasets, with $m = 900$ and 1500, the number of partitions $p = 3$. We bold the numbers of the best algorithm.

Dataset	Nyström(m = 900)		Subgauss(m = 900)		ROS(m = 900)		DKRR-RS(m = 900)	
	time	error	time	error	time	error	time	error
cadata	0.55	1.08 ± 0.133	2.19	**0.15±0.021**	3.95	1.20 ± 0.119	**0.21**	0.19±0.023
shuttle	1.86	0.25 ± 0.012	10.4	**0.05±0.006**	21.1	0.26 ± 0.022	**0.30**	**0.05±0.001**
w8a	3.27	0.04 ± 0.001	13.0	**0.02±0.002**	36.0	0.05 ± 0.003	**0.39**	**0.02±0.004**
Dataset	Nyström(m = 1500)		Subgauss(m = 1500)		ROS(m = 1500)		DKRR-RS(m = 1500)	
	time	error	time	error	time	error	time	error
cadata	1.22	0.97 ± 0.038	3.20	**0.13±0.046**	4.44	0.99 ± 0.023	**0.29**	0.14±0.056
shuttle	4.52	0.25 ± 0.013	15.9	**0.04±0.002**	23.5	0.25 ± 0.015	**0.50**	0.05±0.002
w8a	6.00	0.03 ± 0.0061	16.3	**0.02±0.001**	36.5	0.04 ± 0.002	**0.71**	**0.02±0.007**

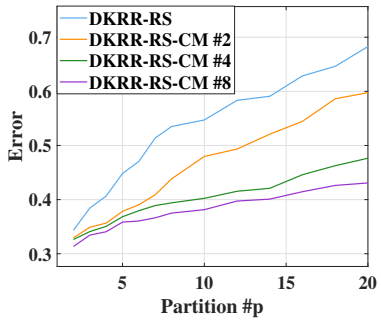


Figure 2: The testing error about the number of partitions p of DKRR-RS and DKRR-RS-CM on connect-4 datasets. 2, 4, and 8 represent the number of communications. $m = 600$.

partitions, the shorter the time of p-related algorithms. The gap between DKRR-RS and DKRR shows that the proposed sparse randomized sketching can greatly reduce the time consumption with a little error. They are consistent with our theoretical analysis and Theorem 1 and 2 that the proposed DKRR-RS can achieve satisfactory accuracy and less time with suitable p and m.

Table 2 compare the training time and testing error of DKRR-RS and m-related algorithms (Subgauss, ROS and Nyström) on cadata, shuttle and w8a datasets with $p = 3$, $m = 900$ and $m = 1500$. The experimental results show that the larger m is, the longer the training time is and the smaller the test error is. Under the same m, DKRR-RS is evidently superior to other algorithms in training time and keeps the (nearly) best testing error, which is consistent with the theoretical analysis. When $m = 1500$, DKRR-RS is even one order of magnitude faster than Nyström algorithm.

Figure 2 compares DKRR-RS-CM ($M = 2, 4, 8$) with DKRR-RS, with $m = 600 > \sqrt{N}$, which shows that: 1) With the increase of p, errors of distributed algorithms (DKRR-RS-CM and DKRR-RS) are gradually increasing. When p is bigger than some upper bounds, the errors are far from the starting point. This demonstrates Theorem 1, 2, and 3. 2) The upper bound p of DKRR-RS-CM is bigger than that of DKRR-RS, which verifies the power of communication strategy in enlarging the range of p. 3) As the number of communication increases, the upper bound of p is increasing. This is consistent with Theorem 3. More experiments and the details of datasets are given in Appendix.

7 Conclusions

In this paper, we propose DKRR-RS method by combining distributed learning and randomized sketching, and investigate its statistical and computational requirements in expectation and probability. Then, to further improve the learning performance, we construct an efficient communication strategy for DKRR-RS and demonstrate the power of communications via theoretical and empirical assessments.

Acknowledgments

This work was supported in part by the Special Research Assistant project of CAS (No.E0YY221-2020000702), the National Natural Science Foundation of China (No.62106259, No.62076234), and Beijing Outstanding Young Scientist Program NO.BJJWZYJH012019100020098. Thank Professor Weiping Wang for his help in this paper.

References

Caponnetto, A.; and Vito, E. D. 2007. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3): 331–368.

Chang, X.; Lin, S.-B.; and Zhou, D.-X. 2017. Distributed semi-supervised learning with kernel ridge regression. *The Journal of Machine Learning Research*, 18(1): 1493–1514.

Guo, Z.-C.; Lin, S.-B.; and Shi, L. 2019. Distributed learning with multi-penalty regularization. *Applied and Computational Harmonic Analysis*, 46(3): 478–499.

Li, J.; Liu, Y.; and Wang, W. 2019. Distributed learning with random features. *arXiv preprint arXiv:1906.03155*.

Li, J.; Liu, Y.; Yin, R.; Zhang, H.; Ding, L.; and Wang, W. 2018. Multi-class learning: From theory to algorithm. In *Advances in Neural Information Processing Systems*, 1593–1602.

Li, M.; Kwok, J. T.; and Lu, B. L. 2010. Making large-scale Nyström approximation possible. In *International Conference on Machine Learning*, 631–638.

Li, S.; and Liu, Y. 2021a. Sharper generalization bounds for clustering. In *International Conference on Machine Learning*, 6392–6402.

Li, S.; and Liu, Y. 2021b. Towards sharper generalization bounds for structured prediction. *Advances in Neural Information Processing Systems*.

Lian, H.; Liu, J.; and Fan, Z. 2021. Distributed learning for sketched kernel regression. *Neural Networks*, 143: 368–376.

Lin, J.; and Cevher, V. 2018. Optimal distributed learning with multi-pass stochastic gradient methods. In *International Conference on Machine Learning*, 3098–3107.

Lin, J.; and Cevher, V. 2020. Convergences of regularized algorithms and stochastic gradient methods with random projections. *The Journal of Machine Learning Research*, 21(20): 1–44.

Lin, S.-B.; Guo, X.; and Zhou, D.-X. 2017. Distributed learning with regularized least squares. *The Journal of Machine Learning Research*, 18(1): 3202–3232.

Lin, S. B.; Lei, Y.; and Zhou, D.-X. 2019. Boosted kernel ridge regression: Optimal learning rates and early stopping. *The Journal of Machine Learning Research*, 20(46): 1–36.

Lin, S.-B.; Wang, D.; and Zhou, D.-X. 2020. Distributed kernel ridge regression with communications. *The Journal of Machine Learning Research*, 21: 93:1–93:38.

Liu, M.; Shang, Z.; and Cheng, G. 2019. Sharp theoretical analysis for nonparametric testing under random projection. In *Conference on Learning Theory*, 2175–2209.

Liu, Y. 2021. Refined learning bounds for kernel and approximate k-means. In *Advances in Neural Information Processing Systems*.

Liu, Y.; Liu, J.; and Wang, S. 2021. Effective distributed learning with random features: Improved bounds and algorithms. In *International Conference on Learning Representations*.

Rahimi, A.; and Recht, B. 2007. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 1177–1184.

Rudi, A.; Camoriano, R.; and Rosasco, L. 2015. Less is more: Nyström computational regularization. In *Advances in Neural Information Processing Systems*, 1657–1665.

Rudi, A.; Camoriano, R.; and Rosasco, L. 2016. Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems*, 3215––3225.

Rudi, A.; Carratino, L.; and Rosasco, L. 2017. Falkon: An optimal large scale kernel method. In *Advances in Neural Information Processing Systems*, 3888–3898.

Saunders, C.; Gammerman, A.; and Vovk, V. 1998. Ridge regression learning algorithm in dual variables. In *International Conference on Machine Learning*, 515–521.

Schölkopf, B.; Herbrich, R.; and Smola, A. J. 2001. A generalized representer theorem. In *International Conference on Computational Learning Theory*, 416–426. Springer.

Shalev Shwartz, S.; Singer, Y.; Srebro, N.; and Cotter, A. 2011. Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical Programming*, 127(1): 3–30.

Smale, S.; and Zhou, D. X. 2007. Learning theory estimates via integral operators and their approximations. *Constructive Approximation*, 26(2): 153–172.

Steinwart, I.; and Christmann, A. 2008. *Support vector machines*. Springer Science & Business Media.

Wang, J.; Liu, J.; Liu, Y.; et al. 2021. Improved learning rates of a functional Lasso-type SVM with sparse multi-kernel representation. In *Advances in Neural Information Processing Systems*.

Wang, S. 2019. A sharper generalization bound for divide-and-conquer ridge regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 5305–5312.

Yang, Y.; Pilanci, M.; and Wainwright, M. J. 2017. Randomized sketches for kernels: Fast and optimal nonparametric regression. *The Annals of Statistics*, 45(3): 991–1023.

Yin, R.; Liu, Y.; Lu, L.; Wang, W.; and Meng, D. 2020a. Divide-and-conquer learning with Nyström: Optimal rate and algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6696–6703.

Yin, R.; Liu, Y.; Wang, W.; and Meng, D. 2020b. Extremely sparse Johnson-Lindenstrauss transform: From theory to algorithm. In *2020 IEEE International Conference on Data Mining*, 1376–1381. IEEE.

Yin, R.; Liu, Y.; Wang, W.; and Meng, D. 2020c. Sketch kernel ridge regression using circulant matrix: Algorithm and theory. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9): 3512–3524.

Yin, R.; Liu, Y.; Wang, W.; and Meng, D. 2021. Distributed Nyström kernel learning with communications. In *International Conference on Machine Learning*, 12019–12028.

Zhang, Y.; Duchi, J.; and Wainwright, M. 2013. Divide and conquer kernel ridge regression. In *Conference on Learning Theory*, 592–617.

Zhang, Y.; Duchi, J.; and Wainwright, M. 2015. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *The Journal of Machine Learning Research*, 16(1): 3299–3340.