# Distributed Nyström Kernel Learning with Communications

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

## Abstract

We study the statistical performance for distributed kernel ridge regression with Nyström (DKRR-NY) and with Nyström and iterative solvers (DKRR-NY-PCG) and successfully derive the optimal learning rates, which can improve the ranges of the number of local processors $p$ to the optimal in existing state-of-art bounds. More precisely, our theoretical analysis show that DKRR-NY and DKRR-NY-PCG achieve the same learning rates as the exact KRR requiring essentially $\mathcal{O}(|D|^{1.5})$ time and $\mathcal{O}(|D|)$ memory with relaxing the restriction on $p$ in expectation, where $|D|$ is the number of data, which exhibits the average effectiveness of multiple trials. Furthermore, for showing the generalization performance in a single trial, we deduce the learning rates for DKRR-NY and DKRR-NY-PCG in probability. Finally, we propose a novel algorithm DKRR-NY-CM based on DKRR-NY, which employs a communication strategy to further improve the learning performance, whose effectiveness of communications is validated in theoretical and experimental analysis.

## 1. Introduction

In nonparametric statistical learning, Kernel ridge regression (KRR) has made a remarkable achievements (Trevor et al., 2009; Taylor & Cristianini, 2004; Yin et al., 2019). However, due to the high computational requirements, KRR does not scale well in large scale settings.

To address the scalability issues, a series of large scale techniques are widely used: Nyström methods (Yin et al., 2020a; Rudi et al., 2015; 2017; Li et al., 2010), random features (Liu et al., 2021; Li et al., 2019; Rudi et al., 2016; Avron et al., 2017), random projections (Lin & Cevher, 2020; Liu et al., 2019; Yang et al., 2017; Williams & Seeger, 2001; Yin et al., 2020b), iterative optimization (Carratino et al., 2018; Lo et al., 2008; Shalev-Shwartz et al., 2011; Gonen et al., 2016; Cutajar et al., 2016; Ma & Belkin, 2017; Rudi et al., 2017), distributed learning (Liu et al., 2021; Lin et al., 2020; Lin & Cevher, 2018; Zhang et al., 2013; Wang, 2019; Chang et al., 2017b; Guo et al., 2019; Zhang et al., 2015; Lin et al., 2017), and combination of the above methods which includes the combination of distributed learning, Nyström and iterative optimization (Yin et al., 2020a), distributed learning and random features (Liu et al., 2021; Li et al., 2019), Nyström and iterative optimization (Rudi et al., 2017), etc. Recent statistical learning works demonstrate that the combination of distributed learning and Nyström (Yin et al., 2020a) can achieve great computational gains and guarantee the optimal theoretical properties. However, the main theoretical bottleneck is that there is a strict restriction on the number of local processors. More specifically, under the basic setting, the upper bound of the local processors is restricted to be a constant with the optimal learning rate, which cannot meet the demand in practical applications.

In this paper, we focus on enlarging the number of local processors and considering the communication strategy between local processors while preserving the optimal learning rates. Firstly, we improve the existing state-of-art performances of the distributed learning together with Nyström (DKRR-NY) and with Nyström and iterative optimization (DKRR-NY-PCG) in expectation. In particular, to guarantee the optimal learning rates, we theoretically derive their upper bounds $\mathcal{O}(\sqrt{|D|})$ of partitions, while it is limited to a constant $\mathcal{O}(1)$ under the basic setting in the existing state-of-art bounds, where $|D|$ is the number of data. The expectation demonstrates the average effectiveness of multiple trials but may fail to capture the generalization performance for a single trial. Therefore, we further deduce the optimal learning rates for DKRR-NY and DKRR-NY-PCG in probability, which can support numerical observations that cannot be seen from the estimates in expectation. Finally, we propose a novel algorithm (DKRR-NY-CM) based on DKRR-NY, which utilizes a communication strategy to further improve the performance and protect the privacy of data in each local processor. Both theoretical analysis and

[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. Correspondence to: Weiping Wang <wangweiping@iie.ac.cn>.

*Proceedings of the $38^{th}$ International Conference on Machine Learning*, PMLR 139, 2021. 

numerical results are conducted to verify the power of the proposed communications.

The rest of the paper is organized as follows. In section 2, we introduce the related work. Section 3 is the background about KRR, Nyström, PCG (preconditioning and conjugate gradient), and divide-and-conquer methods. Section 4 introduces the proposed algorithm (DKRR-NY-CM). In section 5, we mainly show the improved theoretical analysis of DKRR-NY and DKRR-NY-PCG in expectation and probability, and show the optimal learning rate for the proposed DKRR-NY-CM in probability. The following sections are about the numerical experiments and conclusions.

## 2. Related Work

In this section, we mainly introduce the related Nyström and distributed learning in approximate KRR.

The key techniques of Nyström (Li et al., 2010; Rudi et al., 2015; Tu et al., 2016; Camoriano et al., 2016; Rudi et al., 2017; Yin et al., 2020a) are to construct the approximate kernel matrix with a few Nyström centers, which are obtained by different strategies, so as to characterize statistical and computational trade-offs, that is if, or under which conditions, computational gains come at the expense of statistical accuracy. The paper (Rudi et al., 2015) is one of the representative Nyrström method, which utilized both uniform and leverage score based sampling strategies to achieve the same optimal learning rates as the exact KRR with dramatically reducing the computational requirements. Subsequently, for substantially improving computations with preserving the optimal theoretical accuracy, Nyström-PCG method was proposed in (Rudi et al., 2017) by combining Nyström methods (Rudi et al., 2015) with preconditioning and conjugate gradient (PCG) (Cutajar et al., 2016), whose time complexity and space complexity are $\mathcal{O}(|D|m + m^3)$ and $\mathcal{O}(|D|m)$ with $m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}})$ and the optimal learning rate, where $m$ is the sampling scale. Further, Yin et al. (Yin et al., 2020a) proposed DKRR-NY-PCG, which combined Nyström-PCG (Rudi et al., 2017) and divide-and-conquer method, to scale up KRR. Its time complexity and space complexity are $\mathcal{O}(\max(\frac{|D|m}{p}, m^3))$ and $\mathcal{O}(\frac{|D|m}{p})$ with $p \leq \mathcal{O}(|D|^{\frac{2r-1}{2r+\gamma}})$ and $m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}})$ while preserving the optimal learning rate in expectation, where $p$ is the number of partition. Compared to Nyström-PCG, DKRR-NY-PCG (Yin et al., 2020a) reduces the time complexity and space complexity by factors of $\min(|D|^{\frac{2r-1}{2r+\gamma}} + |D|^{\frac{1-\gamma}{2r+\gamma}}, 1 + |D|^{\frac{2r+\gamma-2}{2r+\gamma}})$ and $|D|^{\frac{2r-1}{2r+\gamma}}$ with the optimal learning rate, where $\frac{2r-1}{2r+\gamma} \geq 0$ and $\frac{1-\gamma}{2r+\gamma} \geq 0$. However, at the basic setting ($r = 1/2$ and $\gamma = 1$), the upper bound of partition is $\mathcal{O}(1)$ in DKRR-NY-PCG, which is not practical in the large scale scenarios.

Distributed KRR (Zhang et al., 2013; 2015; Lin et al., 2017; Guo et al., 2019; Li et al., 2019; Lin et al., 2020; Liu et al., 2021) applies KRR to tackle the data subset in each local processor, then communicates exclusive information such as the data (Bellet et al., 2015), gradients (Zeng & Yin, 2018) and local estimator (Huang & Huo, 2015) between different local processors, finally produces a global estimator by combining local estimators and communicated information on the global processor, typical strategies of which are the majority voting (Mann et al., 2009), weighted average (Chang et al., 2017a) and gradient-based algorithms (Bellet et al., 2015). Divide-and-conquer is one of the most popular distributed methods, whose optimal learning rates for KRR in expectation were established (Zhang et al., 2013; 2015; Lin et al., 2017). The theoretical analysis shows that divide-and-conquer KRR can achieve the same learning rates as the exact KRR, however, there is a strict restriction on the number of local machines (Zhang et al., 2015; Guo et al., 2019). Specifically, to reach the optimal learning rate, $p$ should be restrict to a constant $\mathcal{O}(1)$ in (Lin et al., 2017). Subsequently, in (Lin et al., 2020; Liu et al., 2021), they considered the communications among different local machines to enlarge the number of local machines. However, the communication strategy in (Lin et al., 2020) requires communicating the input data between each local processor, which cannot protect the data privacy of each local processor. Furthermore, for each iteration, the communication complexity of each local processor is $\mathcal{O}(d|D|)$, where $d$ is the dimension, which is infeasible in practice for large scale setting.

## 3. Background

In the supervised learning, given dataset $D = \{(x_i, y_i)_{i=1}^N\}$ be sampled identically and independently from $\mathbf{X} \times \mathbb{R}$ with respect $\rho$, where $\rho$ is a probability measure on $\mathbf{X} \times \mathbb{R}$, which is fixed but unknown. $D = \cup_{j=1}^p D_j$ with $p$ disjoint subsets $\{D_j\}_{j=1}^p$. $N = |D|$. For simplicity, denote with $n$ the number of data in $D_j$. Let $\mathcal{H}$ be a separable reproducing kernel Hibert space (RKHS) with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. The reproducing kernel $K : \mathbf{X} \times \mathbf{X} \to \mathbb{R}$ is a positive definite kernel, measurable and uniformly bounded. We denote with $K_x$ the function $K(x, \cdot)$ and have $(\mathbf{K}_N)_{ij} = K(x_i, x_j)$ for all $x_1, \ldots, x_N \in \mathbf{X}$. We denote with $C_\lambda$ the operator $C + \lambda I$ for $\lambda > 0$, where $I$ is the identity operator. For clarity, we define some linear operators: For any $f, g \in \mathcal{H}$, we have $Z_m : \mathcal{H} \to \mathbb{R}^m$, $Z_m^* : \mathbb{R}^m \to \mathcal{H}$; $S_n = \frac{1}{\sqrt{n}} Z_m, S_n^* = \frac{1}{\sqrt{n}} Z_m^*$; $C_n : \mathcal{H} \to \mathcal{H}$, $\langle f, C_n g \rangle_{\mathcal{H}} = \frac{1}{n} \sum_{i=1}^n f(x_i) g(x_i)$, $C_n = S_n^* S_n$, $K_n = n S_n S_n^*$.

The performance of estimating a function is usually measured by the expected risk in the supervised learning,

$$\inf_{f \in \mathcal{H}} \mathcal{E}(f), \quad \mathcal{E}(f) = \int (f(x) - y)^2 d\rho(x, y), \quad (1)$$

where $\mathcal{H}$ is a space of candidate solutions.

### 3.1. Kernel Ridge Regression (KRR)

Kernel Ridge Regression (KRR) (Schölkopf et al., 2002) considers a space $\mathcal{H}$ of functions

$$\hat{f}_{D,\lambda}(x) = \sum_{i=1}^{|D|} \hat{\alpha}_i K(x_i, x). \tag{2}$$

The coefficients $\hat{\alpha}_1, \ldots, \hat{\alpha}_{|D|}$ are deduced from the square loss problem:

$$\hat{f}_{D,\lambda} = \arg\min_{f\in\mathcal{H}} \frac{1}{|D|} \sum_{i=1}^{|D|} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2, \lambda > 0, \tag{3}$$

where $x_1, \ldots, x_{|D|}$ are the data points and $\mathbf{y} = \mathbf{y}_D = [y_1, \ldots, y_{|D|}]^T$ are the corresponding labels. KRR can be transferred into a linear system

$$\hat{\boldsymbol{\alpha}} = (\mathbf{K}_N + \lambda |D| \mathbf{I})^{-1} \mathbf{y}, \tag{4}$$

where $\mathbf{K}_N$ is the kernel matrix.

To solve the linear system, the time complexity is $\mathcal{O}(|D|^3)$ in the inverse operation of $\mathbf{K}_N + \lambda|D|\mathbf{I}$ and the space complexity is $\mathcal{O}(|D|^2)$ in storing the kernel matrix $\mathbf{K}_N$, which are prohibitive for the large scale setting.

### 3.2. KRR with Nyström (KRR-NY)

For reducing computational requirements, Nyström samples the training set to approximate the empirical kernel matrix. The key of Nyström in (Rudi et al., 2015) is to obtain Nyström centers $\{\tilde{x}_1, \ldots, \tilde{x}_m\}$ by uniformly sampling the data points at random without replacement from the training set. Thus, a smaller hypothesis space $\mathcal{H}_m$ is introduced

$$\mathcal{H}_m = \{f | f = \sum_{i=1}^{m} \alpha_i K(\tilde{x}_i, \cdot), \boldsymbol{\alpha} \in \mathbb{R}^m\},$$

where sampling scale $m \leq |D|$. Considering a space $\mathcal{H}_m$ of functions

$$\tilde{f}_{m,\lambda}(x) = \sum_{i=1}^{m} \tilde{\alpha}_i K(\tilde{x}_i, x), \tag{5}$$

the square loss problem can be transferred into the following

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

The solution of Eq.(6) is characterized by the following equation (Rudi et al., 2015)

$$(P_m C_N P_m + \lambda I) \tilde{f}_{m,\lambda} = \frac{1}{\sqrt{|D|}} P_m S_N^* \mathbf{y}, \tag{7}$$

with $P_m$ the projection operator with range $\mathcal{H}_m$.

The corresponding coefficient $\tilde{\boldsymbol{\alpha}}$ is in the form:

$$\tilde{\boldsymbol{\alpha}} = (\mathbf{K}_{Nm}^T \mathbf{K}_{Nm} + \lambda |D| \mathbf{K}_{mm})^{\dagger} \mathbf{K}_{Nm}^T \mathbf{y}, \tag{8}$$

where $\mathbf{H}^{\dagger}$ denotes the Moore-Penorse pseudoinverse of a matrix $\mathbf{H}$, $(\mathbf{K}_{Nm})_{ij} = K(x_i, \tilde{x}_j)$ with $i \in \{1, \ldots, |D|\}$ and $j \in \{1, \ldots, m\}$, and $(\mathbf{K}_{mm})_{kj} = K(\tilde{x}_k, \tilde{x}_j)$ with $k, j \in \{1, \ldots, m\}$.

### 3.3. KRR with Nyström and PCG

To quickly compute the coefficients $\tilde{\boldsymbol{\alpha}}$ in Eq.(8), PCG (preconditioning and conjugate gradient), one of the most popular gradient methods (Saad, 1996), is introduced, whose speed of convergence can benefit from preconditioning (Rudi et al., 2017).

The key idea behind preconditioning is to use a suitable matrix $\mathbf{P}$ to define an equivalent linear system:

$$\mathbf{P} = \frac{1}{\sqrt{|D|}} \mathbf{T}^{-1} \mathbf{A}^{-1}, \tag{9}$$

where $\mathbf{T} = \text{chol}(\mathbf{K}_{mm})$ and $\mathbf{A} = \text{chol}(\frac{1}{m}\mathbf{T}\mathbf{T}^T + \lambda \mathbf{I})$. chol() represents the Cholesky decomposition.

Then, KRR with Nyström and PCG can be seen as solving the following system

$$\mathbf{P}^T \mathbf{H} \hat{\boldsymbol{\alpha}} = \mathbf{P}^T \mathbf{z}, \text{with } \hat{f}_{m,\lambda}(x) = \sum_{i=1}^{m} \hat{\alpha}_i K(\tilde{x}_i, x), \tag{10}$$

where $\hat{\boldsymbol{\alpha}}$ is solved via $t$-step conjugate gradient algorithm and $t \in \mathbb{N}$. Note that, when $t \to \infty$, $\hat{f}_{m,\lambda}$ in Eq.(10) is equal to $\tilde{f}_{m,\lambda}$ in Eq.(5) (Rudi et al., 2017).

### 3.4. Distributed KRR with Nyström (DKRR-NY) and with Nyström and PCG (DKRR-NY-PCG)

Distributed KRR with Nyström and PCG (DKRR-NY-PCG) is defined as

$$\bar{f}_{D,m,t}^0 = \sum_{j=1}^{p} \frac{|D_j|}{|D|} f_{D_j,m,t}, \tag{11}$$

where $f_{D_j,m,t}$ is the solver in Eq.(10). When $t \to \infty$, Eq.(11) can be seen as distributed KRR with Nyström and without PCG (DKRR-NY). $\bar{f}_{D,m,t}^0$ is rewritten as $\bar{f}_{D,m,\lambda}^0$ and $f_{D_j,m,t}$ is rewritten as $f_{D_j,m,\lambda}$ which is the solver in Eq.(5). In each local processor, the time complexity, space complexity, and communication complexity of DKRR-NY are $\mathcal{O}(m^2|D_j|)$, $\mathcal{O}(m|D_j|)$, and $\mathcal{O}(m)$, respectively. And the corresponding complexity of DKRR-NY-PCG are $\mathcal{O}(m|D_j| + m^3)$, $\mathcal{O}(m|D_j|)$, and $\mathcal{O}(m)$, respectively.

If we utilize Nyström to approximate KRR without distributed method on dataset $D$, its time $\mathcal{O}(m^2|D|)$ and memory $\mathcal{O}(m|D|)$ are high, which is not suitable for large scale setting. Distributed learning is one of the most popular methods to reduce the size of dataset. Therefore, it is significant for Nyström. However, the weighted averaging in Eq.(11) is not good enough to compensate for the loss of samples in each local processor (Lin et al., 2020; Liu et al., 2018; Shang & Cheng, 2017), that is, the weighted average cannot improve the approximation ability of KRR in each local processor, and its approximation ability becomes worse when the number of local processors $p$ increases. Thus, efficient communication strategies and synthetic methods are required to enlarge the range of $p$ to guarantee the best generalization performance of distributed Nyström learning.

## 4. DKRR-NY with Communications (DKRR-NY-CM)

In this section, we present a novel communication strategy for DKRR-NY to further enlarge the number of local processors, which is called DKRR-NY-CM. DKRR-NY-CM communicates the gradients instead of data between local processors, which can protect the privacy of datasets in each local processor. The proposed communication strategy is adaptation from (Lin et al., 2020) to avoid data communication between local processors.

### 4.1. Motivation

According to Eq.(7) about Nyström, we know

$$f_{D,m,\lambda} = \frac{1}{\sqrt{|D|}}(P_m C_N P_m + \lambda I)^{-1} P_m S_N^* \mathbf{y}_D, \quad (12)$$

and

$$\bar{f}^0_{D,m,\lambda} = \sum_{j=1}^{p} \frac{|D_j|}{|D|} \frac{1}{\sqrt{|D_j|}}(P_m C_n P_m + \lambda I)^{-1} P_m S_n^* \mathbf{y}_{D_j}. \quad (13)$$

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

$$f_{D,m,\lambda} = f - (P_m C_N P_m + \lambda I)^{-1} * \left[ (P_m C_N P_m + \lambda I) f - \frac{1}{\sqrt{|D|}} P_m S_N^* \mathbf{y}_D \right], \quad (14)$$

and

$$\bar{f}^0_{D,m,\lambda} = f - \sum_{j=1}^{p} \frac{|D_j|}{|D|}(P_m C_n P_m + \lambda I)^{-1} * \left[ (P_m C_n P_m + \lambda I) f - \frac{1}{\sqrt{|D_j|}} P_m S_n^* \mathbf{y}_{D_j} \right]. \quad (15)$$

The above Eq.(14) and Eq.(15) can be seen as the well known Newton-Raphson iteration.

The half gradient of the empirical risk in Eq.(6) over $\mathcal{H}_m$ on $f$ is

$$G_{D,m,\lambda}(f) = (P_m C_N P_m + \lambda I) f - \frac{1}{\sqrt{|D|}} P_m S_N^* \mathbf{y}_D. \quad (16)$$

Noting that the global gradient $G_{D,m,\lambda}$ can be achieved via the communications of each local gradient

$$G_{D,m,\lambda}(f) = \sum_{j=1}^{p} \frac{|D_j|}{|D|} G_{D_j,m,\lambda}(f). \quad (17)$$

For $l > 0$, let

$$\beta_j^{l-1} = (P_m C_n P_m + \lambda I)^{-1} G_{D,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda}). \quad (18)$$

Comparing Eq.(14) and Eq.(15), we can use the Newton-Raphson iteration to design a communication strategy formed as

$$\bar{f}^l_{D,m,\lambda} = \bar{f}^{l-1}_{D,m,\lambda} - \sum_{j=1}^{p} \frac{|D_j|}{|D|} \beta_j^{l-1}. \quad (19)$$

In the following, we introduce the detail flows of the proposed DKRR-NY-CM.

### 4.2. DKRR-NY with Communications (DKRR-NY-CM)

Based on DKRR-NY and Eq.(19), we propose an iteration procedure to implement the communication strategy of DKRR-NY-CM. The detail of DKRR-NY-CM is shown in Algorithm 1. Denote with $M$ the number of communication. For $l$ from 0 to $M$, if $l = 0$: We compute $f_{D_j,m,\lambda}$ according to Eq.(5) in each local processor and communicate them back to the global processor; Then, we compute $\bar{f}^0_{D,m,\lambda}$ in Eq.(11) by the back values of the local processors and communicate it to each local processor. If $l > 0$: We have four steps. In the first step, we compute the local gradient $G_{D_j,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda})$ in Eq.(16) and communicate them back to the global processor; Then we compute the global gradient $G_{D,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda})$ in Eq.(17) and communicate them to each local processor; Thirdly, we compute $\beta_j^{l-1}$ in Eq.(18) in local processors and communicate back to the global processor; Finally, we compute $\bar{f}^l_{D,m,\lambda}$ according to Eq.(19) in global processor and communicate to each local processor. Loop the above operations until that $l$ is equal to the number of communication $M$ and finally output $\bar{f}^M_{D,m,\lambda}$.

The testing flows are shown in Appendix.

**Algorithm 1** DKRR-NY with Communications (DKRR-NY-CM)

**Input**: $p$ disjoint subsets $\{D_j\}_{j=1}^p$ with $D = \cup_{j=1}^p D_j$, kernel parameter, regularization parameter $\lambda$, sampling scale $m$.

**For** $l = 0$ **to** $M$ **do**

- **If** $l = 0$
  **Local processor**: compute $f_{D_j,m,\lambda}$ in Eq.(5), and communicate back to global processor.
  **Global processor**: compute $\bar{f}^0_{D,m,\lambda}$ in Eq.(11), and communicate to each local processor.
- **Else**
  **Local processor**: compute local gradient $G_{D_j,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda})$ in Eq.(16) and communicate back to the global processor.
  **Global processor**: compute global gradient $G_{D,m,\lambda}(\bar{f}^{l-1}_{D,m,\lambda})$ in Eq.(17), and communicate to each local processor.
  **Local processor**: compute $\beta_j^{l-1}$ in Eq.(18) and communicate back to the global processor.
  **Global processor**: compute $\bar{f}^l_{D,m,\lambda}$ in Eq.(19), and communicate to each local processor.
- **End If**

**End For**

### 4.3. Complexity Analysis

(1) Time complexity: In each local processor, we need to compute the matrices multiplication $\mathbf{K}_{nm}^T\mathbf{K}_{nm}$ and the inverse of $\mathbf{K}_{nm}^T\mathbf{K}_{nm} + \lambda|D_j|\mathbf{K}_{mm}$ once. In each iteration except for $l = 0$, we need to compute local gradient $G_{D_j,m,\lambda}$ and $\beta_j$ for each local processor. Thus the total time complexity is $\mathcal{O}(m^2|D_j| + Mm|D_j|)$ in each local processor.

(2) Space complexity: In each local processor, the decisive element is the scale of matrix $K_{nm}$, whose space complexity is $\mathcal{O}(m|D_j|)$.

(3) Communication complexity: The global processor needs receive local gradient $G_{D_j,m,\lambda}$ and $\beta_j$ from the local processors, and distribute $G_{D,m,\lambda}$ and $\bar{f}^l_{D,m,\lambda}$ to local processors in each iteration except for $l = 0$. In $l = 0$, the global processor and local processors need to communicate $\bar{f}^0_{D,m,\lambda}$ and $f_{D_j,m,\lambda}$. Therefore the total communication complexity is $\mathcal{O}(Mm)$.

## 5. Theoretical Analysis

In this section, we first introduce some basic assumptions which are widely used in statistical learning of squared loss (Smale & Zhou, 2007; Caponnetto & Vito, 2007; Rudi et al., 2017; Li et al., 2019). Then we analyze the generation performance of DKRR-NY, DKRR-NY-PCG and DKRR-NY-CM.

### 5.1. Basic Assumptions

The first Assumption describes that the problem in Eq.(1) has at least a solution (Smale & Zhou, 2007; Caponnetto & Vito, 2007).

**Assumption 1.** *There exists an* $f_{\mathcal{H}} \in \mathcal{H}$ *such that* $\mathcal{E}(f_{\mathcal{H}}) = \min_{f\in\mathcal{H}} \mathcal{E}(f)$.

Secondly, we show a basic assumption on data distribution to derive probabilistic results.

**Assumption 2.** *Let* $z_x$ *be the random variable* $z_x = y - f_{\mathcal{H}}(x)$, *with* $x \in X$, *and* $y$ *distributed according to* $\rho(y|x)$. *Then, there exists* $b, \sigma > 0$ *such that* $\mathbb{E}|z_x|^e \le \frac{1}{2}e!b^{e-2}\sigma^2$ *for any* $e \ge 2$, *almost everywhere on* $X$.

The above assumption (Yin et al., 2020a) holds the bounded $y$ and is satisfied with $\sigma = b$, when $|y| \le b, \forall b > 1$.

In the following, we show an assumption that controls the variance of the estimator (Rudi et al., 2015).

**Assumption 3.** *Let* $C$ *be the covariance operator as* $C : \mathcal{H} \to \mathcal{H}, \langle f, Cg\rangle_{\mathcal{H}} = \int_{\mathbf{X}} f(x)g(x)d\rho_X(x), \forall f, g \in \mathcal{H}$. *For* $\lambda > 0$, *define the random variable* $\mathcal{N}_x(\lambda) = \langle K_x, (C + \lambda I)^{-1}K_x\rangle_{\mathcal{H}}$ *with* $x \in \mathbf{X}$ *distributed according to* $\rho_{\mathbf{X}}$ *and let* $\mathcal{N}(\lambda) = \mathbb{E}\mathcal{N}_x(\lambda)$, $\mathcal{N}_\infty(\lambda) = \sup_{x\in\mathbf{X}} \mathcal{N}_x(\lambda)$. *The kernel* $K$ *is measurable,* $C$ *is bounded. Moreover, for all* $\lambda > 0$ *and a* $Q > 0$,

$$\mathcal{N}_\infty(\lambda) < \infty, \tag{20}$$

$$\mathcal{N}(\lambda) \le Q\lambda^{-\gamma}, \quad 0 < \gamma \le 1. \tag{21}$$

In the above assumption, $\gamma$ inflects the size of RKHS $\mathcal{H}$, namely it quantifies the capacity assumption (Yin et al., 2020a). The more benign situation with smaller $\mathcal{H}$ is obtained when $\gamma \to 0$. If the kernel satisfied $\sup_{x\in X} K(x, x) = \kappa^2 < \infty$, we have $\mathcal{N}_\infty(\lambda) \le \kappa^2/\lambda$ for all $\lambda > 0$. The assumption ensures that the covariance operator is a well defined linear, continuous, self-adjoint, positive operator. Because the operator $C$ is trace class, Eq.(21) always holds for $\gamma = 1$.

**Assumption 4.** *There exists* $s \ge 0, 1 \le R < \infty$, *such that*

$$\left\|C^{-s} f_{\mathcal{H}}\right\|_{\mathcal{H}} < R. \tag{22}$$

The above assumption (Rudi et al., 2015) can quantify the degree that $f_{\mathcal{H}}$ can be well approximated by functions in the RKHS $\mathcal{H}$, and can be seen as regularity of $f_{\mathcal{H}}$. For more

details of the four assumptions, please refer to the cited references.

### 5.2. Optimal Learning Rate for DKRR-NY and DKRR-NY-PCG in Expectation

In the following, we analyze the optimal learning rate of DKRR-NY in expectation. Let $r = 1/2 + \min(s, 1/2)$.

**Theorem 1.** *Under Assumptions 1, 2, 3, and 4, let* $\delta \in (0,1]$, $r \in [1/2, 1]$, $\gamma \in (0,1]$, $\lambda = \Omega(|D|^{-\frac{1}{2r+\gamma}})$, $|D_1| = \ldots = |D_p|$, *and* $\bar{f}^0_{D,m,\lambda}$ *be the estimator. With probability* $1-\delta$, *when*

$$p \leq \mathcal{O}(|D|^{\frac{2r+\gamma-1}{2r+\gamma}}) \text{ and } m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}}),$$

*we have* $\mathbb{E}[\mathcal{E}(\bar{f}^0_{D,m,\lambda})] - \mathcal{E}(f_{\mathcal{H}}) = \mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$.

*Proof.* The proof is given in Appendix. □

The following is the optimal learning rate of DKRR-NY-PCG in expectation.

**Corollary 1.** *Under Assumptions 1, 2, 3, and 4, let* $\delta \in (0,1]$, $r \in [1/2, 1]$ $\gamma \in (0,1]$, $\lambda = \Omega(|D|^{-\frac{1}{2r+\gamma}})$, $|D_1| = \ldots = |D_p|$, *and* $\bar{f}^0_{D,m,t}$ *be the estimator. With probability* $1-\delta$, *when* $t \geq \mathcal{O}(\log(|D|))$,

$$p \leq \mathcal{O}(|D|^{\frac{2r+\gamma-1}{2r+\gamma}}), \text{ and } m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}}),$$

*we have* $\mathbb{E}[\mathcal{E}(\bar{f}^0_{D,m,t})] - \mathcal{E}(f_{\mathcal{H}}) = \mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$.

*Proof.* The proof is given in Appendix. □

Note that $\mathbb{E}[\mathcal{E}(\bar{f}^0_{D,m,\lambda})] - \mathcal{E}(f_{\mathcal{H}}) = \mathbb{E}[\|\bar{f}^0_{D,m,\lambda} - f_{\mathcal{H}}\|^2_\rho]$ and $\mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$ [1] is the optimal learning rate of KRR (Caponnetto & Vito, 2007; Yin et al., 2020a). Theorem 1 and Corollary 1 show that if $p \leq |D|^{\frac{2r+\gamma-1}{2r+\gamma}}$ and $m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}})$, the learning rates of the proposed DKRR-NY and DKRR-NY-PCG can reach $\mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$ which are the same statistical accuracy as the exact KRR. The proposed DKRR-NY and DKRR-NY-PCG derive the same learning rate with the number of iteration $t \geq \mathcal{O}(\log(|D|))$ in expectation, which verifies that the error bound caused by PCG is small (Rudi et al., 2017). Let $\lambda = 1/\sqrt{|D|}$, with the optimal learning rate, the proposed DKRR-NY and DKRR-NY-PCG both achieve $\mathcal{O}(|D|^{1.5})$ time complexity and $\mathcal{O}(|D|)$ space complexity.

Theoretical analysis show that divide-and-conquer KRR (Lin et al., 2017; Guo et al., 2019), DKRR-NY-PCG (Yin et al., 2020a), and DKRR-RF (Li et al., 2019) also obtain the same learning rates as the exact KRR in expectation. However, they have a strict limitation to the number of local processors $p$. In particular, at the basic setting, to guarantee the optimal generalization properties, the upper bounds of $p$ in them are restricted to $\mathcal{O}(1)$, but our results in DKRR-NY-PCG and DKRR-NY are both $\mathcal{O}(\sqrt{|D|})$. DKRR-RF (Liu et al., 2021) obtains the optimal learning rate with $p \geq \mathcal{O}(|D|^{\frac{2r+\gamma-1}{2r+\gamma}})$ and $m \leq \mathcal{O}(|D|^{\frac{(2r-1)\gamma+1}{2r+\gamma}})$ in expectation. Compared to DKRR-RF (Liu et al., 2021) in expectation, the proposed DKRR-NY-PCG reduces the time complexity and space complexity by factors of $|D|^{\frac{2(2r-1)\gamma}{2r+\gamma}}$ and $|D|^{\frac{(2r-1)\gamma}{2r+\gamma}}$ with the optimal learning rate, where $(2r-1)\gamma \geq 0$. Compared to Nyström-PCG (Rudi et al., 2017), DKRR-NY-PCG proposed by this paper reduces the time complexity and space complexity by factors of $\min(|D|^{\frac{2r+\gamma-1}{2r+\gamma}} + |D|^{\frac{1}{2r+\gamma}}, 1 + |D|^{\frac{2r+\gamma-2}{2r+\gamma}})$ and $|D|^{\frac{2r+\gamma-1}{2r+\gamma}}$ with the optimal learning rate, where $2r+\gamma-1 > 0$.

### 5.3. Optimal Learning Rate for DKRR-NY and DKRR-NY-PCG in Probability

Theorem 1 and Corollary 1 describe the optimal learning rates for DKRR-NY and DKRR-NY-PCG in expectation. The expectation demonstrates the average effectiveness of multiple trials, but may fail to capture the learning performance for a single trial. Therefore, in the following, we deduce the learning rates for DKRR-NY and DKRR-NY-PCG in probability.

The below is the learning rate of DKRR-NY in probability.

**Theorem 2.** *Under Assumptions 1, 2, 3, and 4, let* $\delta \in (0,1]$, $r \in [1/2, 1]$, $\gamma \in (0,1]$, $\lambda = \Omega(|D|^{-\frac{1}{2r+\gamma}})$, $|D_1| = \ldots = |D_p|$, *and* $\bar{f}^0_{D,m,\lambda}$ *be the estimator. With probability* $1-\delta$, *when*

$$p \leq \mathcal{O}(|D|^{\frac{2r+\gamma-1}{4r+2\gamma}}) \text{ and } m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}}),$$

*we have* $\|\bar{f}^0_{D,m,\lambda} - f_{\mathcal{H}}\|^2_\rho = \mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$.

*Proof.* The proof is given in Appendix. □

Here is the learning rate of DKRR-NY-PCG in probability.

**Corollary 2.** *Under Assumptions 1, 2, 3, and 4, let* $\delta \in (0,1]$, $r \in [1/2, 1]$, $\gamma \in (0,1]$, $\lambda = \Omega(|D|^{-\frac{1}{2r+\gamma}})$, $|D_1| = \ldots = |D_p|$, *and* $\bar{f}^0_{D,m,t}$ *be the estimator. With probability* $1-\delta$, *when* $t \geq \mathcal{O}(\log(|D|))$,

$$p \leq \mathcal{O}(|D|^{\frac{2r+\gamma-1}{4r+2\gamma}}), \text{ and } m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}}),$$

*we have* $\|\bar{f}^0_{D,m,t} - f_{\mathcal{H}}\|^2_\rho = \mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$.

[1] Logarithmic terms of learning rates and complexity are hidden in this paper.

*Table 1.* Computational complexity of the classical approximation algorithms in KRR estimates with the optimal learning rate and $\lambda = 1/\sqrt{|D|}$. The columns from second to last correspond to the time complexity, space complexity, communication complexity, the number of partitions $p$, $m$, and types, respectively. $m$ denotes the sampling scale in Nyström and the number of random features in random features methods. $|D|$ denotes the number of training data, $M$ is the number of communication, $d > 0$, $\Delta_1 = \frac{(1-\gamma)\gamma}{2} \geq 0$, $\Delta_2 = \frac{\gamma}{2} > 0$, and $\gamma \in (0, 1]$. Logarithmic terms are not showed.

| Algorithms | Time | Space | Comm | $p$ | $m$ | Types |
|---|---|---|---|---|---|---|
| KRR (Caponnetto & Vito, 2007) | $\|D\|^3$ | $\|D\|^2$ | / | / | / | In probability |
| Nyström (Rudi et al., 2015) | $\|D\|^2$ | $\|D\|^{1.5}$ | / | / | $\|D\|^{0.5}$ | In probability |
| Nyström-PCG (Rudi et al., 2017) | $\|D\|^{1.5}$ | $\|D\|^{1.5}$ | / | / | $\|D\|^{0.5}$ | In probability |
| Random Features (Rudi et al., 2016) | $\|D\|^{2+2\Delta_1}$ | $\|D\|^{1.5+\Delta_1}$ | / | / | $\|D\|^{0.5+\Delta_1}$ | In probability |
| DKRR-RF (Li et al., 2019) | $\|D\|^{1.5+2\Delta_1+\Delta_2}$ | $\|D\|^{1+\Delta_1+\Delta_2}$ | $\|D\|^{0.5+\Delta_1}$ | $\|D\|^{0.5-\Delta_2}$ | $\|D\|^{0.5+\Delta_1}$ | In expectation |
| DKRR-RF (Liu et al., 2021) | $\|D\|^{1.5+2\Delta_1}$ | $\|D\|^{1+\Delta_1}$ | $\|D\|^{0.5+\Delta_1}$ | $\|D\|^{0.5}$ | $\|D\|^{0.5+\Delta_1}$ | In expectation |
| DKRR-RF (Liu et al., 2021) | $\|D\|^{1.75+2\Delta_1}$ | $\|D\|^{1.25+\Delta_1}$ | $\|D\|^{0.5+\Delta_1}$ | $\|D\|^{0.25}$ | $\|D\|^{0.5+\Delta_1}$ | In probability |
| DKRR-RF-CM (Liu et al., 2021) | $\|D\|^{\frac{3M+7}{2M+4}+2\Delta_1}$ | $\|D\|^{\frac{2M+5}{2M+4}+\Delta_1}$ | $M\|D\|^{0.5+\Delta_1}$ | $\|D\|^{\frac{M+1}{2(M+2)}}$ | $\|D\|^{0.5+\Delta_1}$ | In probability |
| DKRR (Chang et al., 2017b) | $\|D\|^2$ | $\|D\|$ | $\|D\|^{0.5}$ | $\|D\|^{0.5}$ | / | In expectation |
| DKRR (Lin et al., 2020) | $\|D\|^{2.25}$ | $\|D\|^{1.5}$ | $\|D\|^{0.75}$ | $\|D\|^{0.25}$ | / | In probability |
| DKRR-CM (Lin et al., 2020) | $\|D\|^{\frac{3(M+3)}{2(M+2)}}$ | $\|D\|^{\frac{M+3}{M+2}}$ | $Md\|D\|$ | $\|D\|^{\frac{M+1}{2(M+2)}}$ | / | In probability |
| DKRR-NY-PCG (Yin et al., 2020a) | $\|D\|^{1.5}$ | $\|D\|^{1+\Delta_2}$ | $\|D\|^{0.5}$ | $\|D\|^{0.5-\Delta_2}$ | $\|D\|^{0.5}$ | In expectation |
| **DKRR-NY-PCG (Corollary 1)** | $\|D\|^{1.5}$ | $\|D\|$ | $\|D\|^{0.5}$ | $\|D\|^{0.5}$ | $\|D\|^{0.5}$ | In expectation |
| **DKRR-NY-PCG (Corollary 2)** | $\|D\|^{1.75}$ | $\|D\|^{1.25}$ | $\|D\|^{0.5}$ | $\|D\|^{0.25}$ | $\|D\|^{0.5}$ | In probability |
| **DKRR-NY (Theorem 1)** | $\|D\|^{1.5}$ | $\|D\|$ | $\|D\|^{0.5}$ | $\|D\|^{0.5}$ | $\|D\|^{0.5}$ | In expectation |
| **DKRR-NY (Theorem 2)** | $\|D\|^{1.75}$ | $\|D\|^{1.25}$ | $\|D\|^{0.5}$ | $\|D\|^{0.25}$ | $\|D\|^{0.5}$ | In probability |
| **DKRR-NY-CM (Theorem 3)** | $\|D\|^{\frac{3M+7}{2M+4}}$ | $\|D\|^{\frac{2M+5}{2M+4}}$ | $M\|D\|^{0.5}$ | $\|D\|^{\frac{M+1}{2(M+2)}}$ | $\|D\|^{0.5}$ | In probability |

*Proof.* The proof is given in Appendix. □

Note that, DKRR-NY and DKRR-NY-PCG can also achieve the optimal learning rates in probability. The upper bound of $p$ in Theorem 2 and Corollary 2 are $\mathcal{O}(|D|^{\frac{2r+\gamma-1}{4r+2\gamma}})$, which is stricter than $\mathcal{O}(|D|^{\frac{2r+\gamma-1}{2r+\gamma}})$ in Theorem 1 and Corollary 1. The reason is that, compared to the error decomposition in expectation, the error decomposition in probability is not easy to separate a distributed error to control the number of local processors. To derive the optimal learning rate, we provide a novel decomposition, which is shown in Appendix.

### 5.4. Optimal Learning Rate for DKRR-NY-CM in Probability

We demonstrate that DKRR-NY-CM can derive the optimal learning rate and further enlarge the number of partition $p$ compared to DKRR-NY and DKRR-NY-PCG in probability.

**Theorem 3.** *Under Assumptions 1, 2, 3, and 4, let* $\delta \in (0, 1]$, $r \in [1/2, 1]$, $\gamma \in (0, 1]$, $\lambda = \Omega(|D|^{-\frac{1}{2r+\gamma}})$, $|D_1| = \ldots = |D_p|$, *and* $\bar{f}^M_{D,m,\lambda}$ *be the estimator. With probability* $1 - \delta$, *when*

$$p \leq \mathcal{O}(|D|^{\frac{(2r+\gamma-1)(M+1)}{(2r+\gamma)(M+2)}}) \text{ and } m \geq \mathcal{O}(|D|^{\frac{1}{2r+\gamma}}), \quad (23)$$

*we have* $\|\bar{f}^M_{D,m,\lambda} - f_{\mathcal{H}}\|^2_\rho = \mathcal{O}(|D|^{-\frac{2r}{2r+\gamma}})$.

*Proof.* The proof is given in Appendix. □

Comparing Theorem 2 with Theorem 3, we know that, with the same optimal learning rates and $m$, the upper bound of partition $\mathcal{O}(|D|^{\frac{(2r+\gamma-1)(M+1)}{(2r+\gamma)(M+2)}})$ in DKRR-NY-CM is larger than $\mathcal{O}(|D|^{\frac{2r+\gamma-1}{4r+2\gamma}})$ of DKRR-NY in probability, where $M \geq 1$. This means that the proposed communication strategy can relax the restriction on $p$, namely improve the performance of DKRR-NY. Furthermore, the upper bound of $p$ is monotonically increasing with the number of communications $M$, showing the power of communications. DKRR-NY, DKRR-NY-PCG, and DKRR-NY-CM can achieve the rate $\mathcal{O}(1/\sqrt{|D|})$ at the basic setting and the rate $\mathcal{O}(1/|D|)$ under the best case ($r = 1$ and $\gamma = 0$). PCG can also be used to accelerate the calculation in DKRR-NY-CM. In this part, we mainly verify the effectiveness of communication, so PCG is not used.

### 5.5. Compared with the Related Work

Table 1 shows the computational complexity of the classical KRR estimators with the optimal learning rate and $\lambda = 1/\sqrt{|D|}$. By comparison, we know that the proposed DKRR-NY and DKRR-NY-PCG require only $|D|^{1.5}$ time and $|D|^{0.5}$ memory with the optimal learning rates in expectation, which keep the least at the same time and are more effective than other algorithms. For DKRR-NY-CM, with

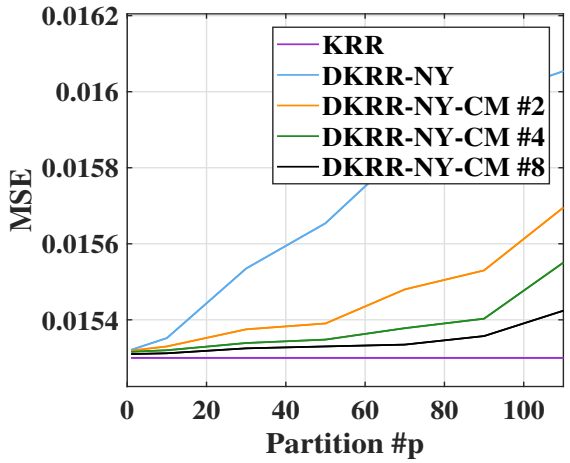


*Figure 1.* The mean square error on testing sampling with different partitions on KRR, DKRR-NY, and our DKRR-NY-CM. The numbers 2, 4 and 8 represent the number of communications.

the optimal learning rates, it keeps the less time complexity, space complexity, and communication complexity in probability compared to the communication-based algorithms of DKRR-RF-CM (Liu et al., 2021) and DKRR-CM (Lin et al., 2020). Meanwhile, the proposed DKRR-NY, DKRR-NY-PCG, and DKRR-NY-CM keep the best upper bound of $p$ and the best lower bound of $m$ at the same condition.

The proof in this paper is non-trivial extensions of (Yin et al., 2020a; Lin et al., 2020; Liu et al., 2021). We provide a novel error decomposition compared to DKRR-NY-PCG (Yin et al., 2020a) so that the improved bounds can be obtained in expectation. If we use the same way of error decomposition in (Yin et al., 2020a), this paper cannot relax the restriction on the number of local processors. Furthermore, we provide the bounds of DKRR-NY-PCG and DKRR-NY in probability and consider communication strategy to further improve the bounds of DKRR-NY in probability which are not obtained in (Yin et al., 2020a; Lin et al., 2020; Liu et al., 2021).

The paper (Lin et al., 2020) provides the communication strategy in DKRR, but it requires communicating the data among local processors, which cannot protect the privacy of data in local processors and increases the communication complexity of each local processor. In this paper, we communicate the gradients, model parameters, and model estimators, which are better to protect data privacy and reduce the communication complexity.

## 6. Experiments

In this section, we report numerical results to verify the theoretical statements about the power of communications in DKRR-NY-CM on simulated dataset.

The way of generating the synthetic data is as below. The training samples $\{\mathbf{x}_i\}_{i=1}^N$ and the testing samples $\{\mathbf{x}'_i\}_{i=1}^{N'}$ are independently drawn according to the uniform distribution on the (hyper-)cube $[0,1]$. The outputs of training samples $\{y_i\}_{i=1}^N$ are generated from the regression models $y_i = g(\mathbf{x}_i) + \varepsilon_i$ for $i = 1, 2, \ldots, N$, where $\varepsilon_i$ is the independent Gaussian noise $\mathcal{N}(0, 0.01)$, if $0 < x \leq 0.5$, $g(x) = x$, otherwise $g(x) = 1 - x$. The outputs of testing samples $\{y'_i\}_{i=1}^{N'}$ are generated by $y'_i = g(\mathbf{x}'_i)$. Define the reproducing kernels $K(x, x') = 1 + \min(x, x')$. Obviously, $g \in \mathcal{H}_K$ (Wu, 1995). The way of generating dataset is the same as (Lin et al., 2020). The criterion is the mean square error (MSE). According to the proposed Theorem, we set sampling scale $m = \sqrt{N}$ and $\lambda = \frac{1}{2\sqrt{N}}$. In the training process of distributed algorithms, we uniformly distribute $N$ training samples to $p$ local processors.

Generating 20000 training samples and 2000 testing samples. Using the exact KRR as a baseline, which trains all samples in a batch. We compare our proposed DKRR-NY-CM with DKRR-NY and KRR, repeat the training 5 times, and estimate the averaged error on testing samples. The testing results are shown in Figure 1, which can be summarized as follows: 1) The larger the $p$ is, the larger the gaps between the distributed algorithms (DKRR-NY and DKRR-NY-CM) and KRR are. When $p$ is larger than an upper bound, MSE of distributed algorithms is far from the exact KRR. This verifies the statement about $p$ in Theorem 1, 2, and 3. 2) The upper bound of $p$ in our DKRR-NY-CM is much larger than that of DKRR-NY. This result verifies Theorem 3 that the communication strategy can relax the restriction on $p$. 3) The upper bound of $p$ is increasing with the number of communication increasing, which shows the effectiveness of communication and is consistent with our theoretical analysis in Eq.(23) of Theorem 3.

## 7. Conclusions

This paper studies the statistical performance of DKRR-NY and DKRR-NY-PCG, and successfully derives the optimal learning rates with enlarging the ranges of $p$ to the optimal in existing state-of-art bounds. Specifically, DKRR-NY and DKRR-NY-PCG achieve the same learning rates as the exact KRR requiring essentially $\mathcal{O}(|D|^{1.5})$ time and $\mathcal{O}(|D|)$ memory with relaxing the restriction on $p$ in expectation. Furthermore, for showing the generalization performance in a single trial, we deduce the learning rates for DKRR-NY and DKRR-NY-PCG in probability. Finally, we utilize a communication strategy to further improve the performance of DKRR-NY and protect the privacy of data in each local processor. Theoretical and experimental analysis verify the effectiveness of the communications.

## Acknowledgements

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.62076234), and the Beijing Outstanding Young Scientist Program (No. BJJWZYJH012019100020098).

## References

Avron, H., Clarkson, K. L., and Woodruff, D. P. Faster kernel ridge regression using sketching and preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 38 (4):1116–1138, 2017.

Bellet, A., Liang, Y., Garakani, A. B., Balcan, M.-F., and Sha, F. A distributed frank-wolfe algorithm for communication-efficient sparse learning. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pp. 478–486. SIAM, 2015.

Camoriano, R., Angles, T., Rudi, A., and Rosasco, L. Nytro: When subsampling meets early stopping. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pp. 1403–1411, 2016.

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

Carratino, L., Rudi, A., and Rosasco, L. Learning with sgd and random features. In *Advances in Neural Information Processing Systems*, pp. 10192–10203, 2018.

Chang, X., Lin, S.-B., Wang, Y., et al. Divide and conquer local average regression. *Electronic Journal of Statistics*, 11(1):1326–1350, 2017a.

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

Cutajar, K., Osborne, M., Cunningham, J., and Filippone, M. Preconditioning kernel matrices. In *International Conference on Machine Learning*, pp. 2529–2538, 2016.

Gonen, A., Orabona, F., and Shalev-Shwartz, S. Solving ridge regression using sketched preconditioned svrg. In *International Conference on Machine Learning*, pp. 1397–1405. PMLR, 2016.

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

Huang, C. and Huo, X. A distributed one-step estimator. *arXiv preprint arXiv:1511.01443*, 2015.

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

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

Lin, J. and Cevher, V. Optimal distributed learning with multi-pass stochastic gradient methods. In *International Conference on Machine Learning*, pp. 3092–3101. PMLR, 2018.

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

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

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

Liu, M., Shang, Z., and Cheng, G. How many machines can we use in parallel computing for kernel ridge regression? *arXiv preprint arXiv:1805.09948*, 2018.

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

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

Lo, G. L., Rosasco, L., Odone, F., De, V. E., and Verri, A. Spectral algorithms for supervised learning. *Neural Computation*, 20(7):1873–1897, 2008.

Ma, S. and Belkin, M. Diving into the shallows: A computational perspective on large-scale shallow learning. *arXiv preprint arXiv:1703.10622*, 2017.

Mann, G., McDonald, R., Mohri, M., Silberman, N., and Walker, D. D. Efficient large-scale distributed training of conditional maximum entropy models. In *Advances in Neural Information Processing Systems*, pp. 1231–1239, 2009.

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

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

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

Saad, Y. *Iterative methods for sparse linear systems*. SIAM, 1996.

Schölkopf, B., Smola, A. J., Bach, F., et al. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

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

Shang, Z. and Cheng, G. Computational limits of a distributed algorithm for smoothing spline. *Journal of Machine Learning Research*, 18:3809–3845, 2017.

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

Taylor, J. S. and Cristianini, N. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

Trevor, H., Robert, T., and Jerome, F. *The elements of statistical learning: Data mining, inference, and prediction*. Springer, New York, 2009.

Tu, S., Roelofs, R., Venkataraman, S., and Recht, B. Large scale kernel learning using block coordinate descent. *arXiv preprint arXiv:1602.05310*, 2016.

Wang, S. A sharper generalization bound for divide-and-conquer ridge regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5305–5312, 2019.

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

Wu, Z. Compactly supported positive definite radial functions. *Advances in computational mathematics*, 4(1): 283–292, 1995.

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

Yin, R., Liu, Y., Wang, W., and Meng, D. Sketch kernel ridge regression using circulant matrix: Algorithm and theory. *IEEE transactions on neural networks and learning systems*, 31(9):3512–3524, 2019.

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

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

Zeng, J. and Yin, W. On nonconvex decentralized gradient descent. *IEEE Transactions on signal processing*, 66(11): 2834–2848, 2018.

Zhang, Y., Duchi, J. C., and Wainwright, M. J. Divide and conquer kernel ridge regression. *Proceeding of the 26th Annual Conference on Learning Theory*, 30(1):592–617, 2013.

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