Ridgeless Regression with Random Features

Jian Li\textsuperscript{1}, Yong Liu\textsuperscript{2*}, Yingying Zhang\textsuperscript{3}

\textsuperscript{1}Institute of Information Engineering, Chinese Academy of Sciences
\textsuperscript{2}Gaoling School of Artificial Intelligence, Renmin University of China
\textsuperscript{3}School of Mathematics and Information Sciences, Yantai University
lijian9026@iie.ac.cn, liuyonggsai@ruc.edu.cn, yingyingzhang239@gmail.com

Abstract
Recent theoretical studies illustrated that kernel ridgeless regression can guarantee good generalization ability without an explicit regularization. In this paper, we investigate the statistical properties of ridgeless regression with random features and stochastic gradient descent. We explore the effect of factors in the stochastic gradient and random features, respectively. Specifically, random features error exhibits the double-descent curve. Motivated by the theoretical findings, we propose a tunable kernel algorithm that optimizes the spectral density of kernel during training. Our work bridges the interpolation theory and practical algorithm.

1 Introduction
In the view of traditional statistical learning, an explicit regularization should be added to the nonparametric learning objective [Caponnetto and De Vito, 2007; Li et al., 2018], i.e. an 2-norm penalty for the kernelized least-squares problems, known as kernel ridge regression (KRR). To ensure good generalization (out-of-sample) performance, the modern models should choose the regularization hyperparameter \( \lambda \) to balancing the bias and variance, and thus avoid overfitting. However, recent studies empirically observed that neural networks still can interpolate the training data and generalize well when \( \lambda = 0 \) [Zhang et al., 2021; Belkin et al., 2018]. But also, for many modern models including neural networks, random forests and random features, the test error captures a 'double-descent' curve as the increase of features dimensional [Mei and Montanari, 2019; Advani et al., 2020; Nakkiran et al., 2021]. Recent empirical successes of neural networks prompted a surge of theoretical results to understand the mechanism for good generalization performance of interpolation methods without penalty where researchers started with the statistical properties of ridgeless regression based on random matrix theory [Liang and Rakhlin, 2020; Bartlett et al., 2020; Jacot et al., 2020; Ghorbani et al., 2021].

However, there are still some open problems to settle down: 1) In the theoretical front, current studies focused on the direct bias-variance decomposition for the ridgeless regression, but ignore the influence from the optimization problem, i.e. stochastic gradient descent (SGD). Meanwhile, the connection between kernel regression and ridgeless regression with random features are still not well established. Further, while asymptotic behavior in the overparameterized regime is well studied, ridgeless models with a finite number of features are much less understood. 2) In the algorithmic, there is still a great gap between statistical learning for ridgeless regression and algorithms. Although the theories explain the double-descent phenomenon for ridgeless methods well, a natural question is whether the theoretical studies helps to improve the mainstream algorithms or design new ones.

In this paper, we consider the Random Features (RF) model [Rahimi and Recht, 2007] that was proposed to approximate kernel methods for easing the computational burdens. In this paper, we investigate the generalization ability of ridgeless random features, of which we explore the effects from stochastic gradient algorithm and random features. And then, motivated by the theoretical findings, we propose a tunable kernel algorithm that optimizes the spectral density of kernel during training, reducing multiple trials for kernel selection to just training once. Our contributions are summarized as:

1) Stochastic gradients error. We first investigate the stochastic gradients error influenced the factors from SGD, i.e. the batch size \( b \), the learning rate \( \gamma \) and the iterations \( t \). The theoretical results illustrate the tradeoffs among these factors to achieve better performance, such that it can guide the set of these factors in practice.

2) Random features error. We then explore the difference between ridgeless kernel predictor and ridgeless RF predictor. In the overparameterized setting \( M > n \), the ridgeless RF converges to the ridgeless kernel as the increase number of random features, where \( M \) is the number of random features and \( n \) is the number of examples. In the underparameterized regime \( M \leq n \), the error of ridgeless RF also exhibit an interesting ”double-descent” curve in Figure 2 (a), because the variance term explores near the transition point \( M = n \).

3) Random features algorithm with a tunable kernel. Theoretical results illustrate the errors depends on the trace of kernel matrix, motivating us to design a kernel learning algorithm which asynchronously optimizes the spectral density and model weights. The algorithm is friend to random initial-
ization, and thus easing the problem of kernel selection.

2 Related Work

Statistical properties of random features. The generalization efforts for random features are mainly in reducing the number of random features to achieve the good performance. [Rahimi and Recht, 2007] derived the appropriate error bound between kernel function and the inner product of random features. And then, the authors proved $O(\sqrt{n})$ features to obtain the error bounds with convergence rate $O(1/\sqrt{n})$ [Rahimi and Recht, 2008]. Rademacher complexity based error bounds have been proved in [Li et al., 2019; Li et al., 2020]. Using the integral operator theory, [Rudi and Rosasco, 2017; Li and Liu, 2022] proved the minimax optimal rates for random features based KRR. In contrast, recent [Hastie et al., 2019] or linear models [Bartlett et al., 2020] efforts on the overparameterized case for random features to compute the asymptotic risk and revealed the double-descent curve for random ReLU [Mei and Montanari, 2019] features and random Fourier features [Jacot et al., 2020].

Double-descent in ridgeless regression. The double-descent phenomenon was first observed in multilayer networks on MNIST dataset for ridgeless regression [Advani et al., 2020]. It then been observed in random Fourier features and decision trees [Belkin et al., 2018]. [Nakkiran et al., 2021] extended the double-descent curve to various models on more complicated tasks. The connection between double-descent curve and random initialization of the Neural Tangent Kernel (NTK) has been established [Geiger et al., 2020]. Recent theoretical work studied the asymptotic error for ridgeless regression with kernel models [Liang and Rakhlin, 2020; Jacot et al., 2020] or linear models [Bartlett et al., 2020; Ghorbani et al., 2021].

3 Problem Setup

In the context of nonparametric supervised learning, given a probability space $\mathcal{X} \times \mathcal{Y}$ with an unknown distribution $\mu(x, y)$, the regression problem with squared loss is to solve

$$
\min_{f} \mathcal{E}(f), \quad \mathcal{E}(f) = \int_{\mathcal{X} \times \mathcal{Y}} (f(x) - y)^2 d\mu(x, y).
$$

(1)

However, one can only observe the training set $(x_i, y_i)^n_{i=1}$ that drawn i.i.d. from $\mathcal{X} \times \mathcal{Y}$ according to $\mu(x, y)$, where $x_i \in \mathbb{R}^d$ are inputs and $y_i \in \mathbb{R}$ are the corresponding labels.

3.1 Kernel Ridgeless Regression.

Suppose the target regression $f^*(x) = \mathbb{E}(y|x = x)$ lie in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}_K$, endowed with the norm $\| \cdot \|_K$ and Mercer kernel $K(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Denote $X = [x_1, \cdots, x_n] \in \mathbb{R}^{n \times d}$ the input matrix and $y = [y_1, \cdots, y_n]^T$ the response vector. We then let $K(X, X) = [K(x_i, x_j)]_{i,j=1}^n \in \mathbb{R}^{n \times n}$ be the kernel matrix and $K(x, X) = [K(x, x_1), \cdots, K(x, x_n)] \in \mathbb{R}^{1 \times n}$.

Given the data $(X, y)$, the empirical solution to (1) admits a closed-form solution:

$$
\hat{f}(x) = K(x, X)K(X, X)^\dagger y,
$$

(2)

where $\dagger$ is the Moore-Penrose pseudo inverse. The above solution is known as kernel ridgeless regression.

3.2 Ridgeless Regression with Random Features

The Mercer kernel is the inner product of feature mapping in $\mathcal{H}_K$, stated as $K(x, x') = \langle K(x, \cdot), K(x', \cdot) \rangle$, $\forall x, x' \in X$, where $K_{\pi} = K(x, \cdot)$ in $\mathcal{H}_K$ is high or infinite dimensional.

The integral representation for kernel is $K(x, x') = \int_{\mathcal{X}} \int_{\mathcal{X}} \psi(x, \omega) \psi(x', \omega) d\pi(\omega)$, where $\pi(\omega)$ is the spectral density and $\psi : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is continuous and bounded function. Random features technique is proposed to approximate the kernel by a finite dimensional feature mapping $K(x, x') \approx \langle \phi(x), \phi(x') \rangle$, with

$$
\phi(x) = \frac{1}{\sqrt{M}} (\psi(x, \omega_1), \cdots, \psi(x, \omega_M)),
$$

(3)

where $\phi : \mathcal{X} \to \mathbb{R}^M$ and $\omega_1, \cdots, \omega_M$ are sampled independently according to $\pi$. The solution of ridgeless regression with random features can be written as

$$
\hat{f}_M(x) = \langle \phi(x), \phi(X) \rangle \hat{\theta} \phi(x),
$$

(4)

where $\phi(X) \in \mathbb{R}^{n \times M}$ is the feature mapping matrix over $X$ and $\phi(x) \in \mathbb{R}^{1 \times M}$ is the feature mapping over $x$.

3.3 Random Features with Stochastic Gradients

To further accelerate the computational efficiency, we consider the stochastic gradient descent method as bellow

$$
\tilde{f}_{M,b,t}(x) = \langle w_t, \phi(x) \rangle,
$$

(5)

$$
\hat{w}_{t+1} = w_t - \gamma_t \sum_{i=(t-1)+1}^t \langle w_t, \phi(x_i) \rangle - y_i \phi(x_i),
$$

(6)

where $w_t \in \mathbb{R}^M$, $w_0 = 0$, $b$ is the mini-batch size and $\gamma_t$ is the learning rate. When $b = 1$ the algorithm reduces to SGD and $b > 1$ is the mini-batch version. We assume the examples are drawn uniformly with replacement, by which one pass over the data requires $\lceil n/b \rceil$ iterations. Before the iteration, the compute of $\phi(X)$ consumes $O(nM)$ time. The time complexity is $O(Mb)$ for each iteration and the total complexity is $O(nMT)$ after $T$ passes on the train set.

4 Main Results

In this section, we study the statistical properties of estimators $\hat{f}_{M,b,t}$ (5), $\hat{f}_M$ (3) and $\hat{f}$ (2). We only present the main theoretical results in this part and the proofs can be find in the full version.\footnote{https://arxiv.org/abs/2205.00477} Denote $\mathbb{E}_{\mu}[]$ the expectation w.r.t. the marginal $x \sim \mu$ and

$$
\|g\|_{L_2(\mu)}^2 = \int g^2(x) d\mu(x) = \mathbb{E}_{\mu}[g^2(x)], \quad \forall g \in L^2(\mu).
$$

The squared integral norm over the space $L^2(\mu) = \{ g : \mathcal{X} \to \mathbb{R} | \int g^2(x) d\mu(x) < \infty \}$ and $f^* \in L^2(\mu)$. Combining the above equation with (1), one can prove that $\mathcal{E}(f) - \mathcal{E}(f^*) = \|f - f^*\|_{L_2(\mu)} \forall f \in L^2(\mu)$. Therefore we can decompose the excess risk of $\mathcal{E}(\hat{f}_{M,b,t}) - \mathcal{E}(f^*)$ as bellow

$$
\mathcal{E}(\hat{f}_{M,b,t}) - \mathcal{E}(f^*) \\
\leq \|\hat{f}_{M,b,t} - \hat{f}_M\| + \|\hat{f}_M - \hat{f}\| + \|\hat{f} - f^*\|.
$$

(6)
The excess risk bound includes three terms: stochastic gradient error $\| \hat{f}_{M,b,t} - \hat{f}_M \|$, random feature error $\| \hat{f}_M - \hat{f} \|$, and excess risk of kernel ridgeless regression $\| \hat{f} - f^* \|$, which admits the bias-variance form. In this paper, since the ridgeless excess risk $\| \hat{f} - f^* \|$ has been well-studied [Liang and Rakhlin, 2020], we focus on the first two bounds and explore the factors in them, respectively. Throughout this paper, we assume the true regression $f^*(x) = (f^*, K_x)$ lies in the RKHS of the kernel $K$, i.e., $f^* \in \mathcal{H}_K$.

**Assumption 1** (Random features are continuous and bounded). Assume that $\psi$ is continuous and there is a $\kappa \in [1, \infty)$, such that $|\psi(x, \omega)| \leq \kappa, \forall x \in \mathcal{X}, \omega \in \Omega$.

**Assumption 2** (Moment assumption). Assume there exists $B > 0$ and $\sigma > 0$, such that for all $p \geq 2$ with $p \in \mathbb{N}$,

$$\int_{\mathbb{R}} |y|^p dP(y|x) \leq \frac{1}{2} p! B^{p-2} \sigma^2. \quad (7)$$

The above two assumptions are standard in statistical learning theory [Smale and Zhou, 2007; Caponnetto and De Vito, 2007; Rudi and Rosasco, 2017]. According to Assumption 1, the kernel $K$ is bounded by $K(x, x) \leq \kappa^2$. The moment assumption on the output $y$ holds when $y$ is bounded, subgaussian or sub-exponential. Assumptions 1 and 2 are standard in the generalization analysis of KRR, always leading to the learning rate $O(1/\sqrt{N})$ [Smale and Zhou, 2007].

### 4.1 Stochastic Gradients Error

We first investigate the approximation ability of stochastic gradient by measuring $\| \hat{f}_{M,b,t} - \hat{f}_M \|_{\mathcal{L}^2(\mu)}$, and explore the effect of the mini-batch size $b$, the learning rate $\gamma$ and the number of iterations $t$.

**Theorem 1** (Stochastic gradient error). Under Assumptions 1, 2, let $t \in [T], \gamma \leq \frac{n}{\sqrt{T} \log n} \wedge \frac{1}{8(1+\log T)}$ and $n \geq 32 \log \frac{2}{\gamma}$, the following bounded holds with high probability

$$\| \hat{f}_{M,b,t} - \hat{f}_M \| \lesssim \frac{\gamma}{b} + \frac{\| f^* \|_K}{\gamma^t}. \quad \quad (8)$$

The first term in the above bound measures the similarity between mini-batch gradient descent estimator and full gradient descent estimator $\| \hat{f}_{M,b,t} - \hat{f}_M \|$, which depends on the mini-batch size $b$ and the learning rate $\gamma$. The second term reflects the approximation between the gradient estimator and the random feature estimator $\| \hat{f}_{M,t} - \hat{f}_{M,b,t} \|$, which is determined by the number of iterations $t$ and the step-size $\gamma$, leading to a sublinear convergence $O(1/t)$.

**Corollary 1.** Under the same assumptions of Theorem 1, one of the following cases and the time complexities

1) $b = 1, \gamma \simeq \frac{1}{\sqrt{n}}$ and $T = n$ $\Rightarrow$ $O(nM)$

2) $b = \sqrt{n}, \gamma \simeq 1$ and $T = \sqrt{n}$ $\Rightarrow$ $O(nM)$

3) $b = n, \gamma \simeq 1$ and $T = \sqrt{n}$ $\Rightarrow$ $O(n\sqrt{n}M)$

is sufficient to guarantee with high probability that

$$\| \hat{f}_{M,b,t} - \hat{f}_M \| \lesssim \frac{1}{\sqrt{n}}. \quad \quad (9)$$

In the above corollary, we give examples for SGD, mini-batch gradient, and full gradient, respectively. It shows the computational efficiency of full gradient descent is usually worse than stochastic gradient methods. The computational complexities $O(nM)$ are much smaller than that of random features $O(nM^2 + M^3)$. All these cases achieve the same learning rate $O(1/\sqrt{m})$ as the exact KRR. With source condition and capacity assumption in integral operator literature [Caponnetto and De Vito, 2007; Rudi and Rosasco, 2017], the above bound can achieve faster convergence rates.

**Remark 1.** [Carratino et al., 2018] also studied the approximation of mini-batch gradient descent algorithm, but the random features estimator is defined with noise-free labels $f^*(X)$ and with ridge regularization, which failed to directly capture the effect of stochastic gradients. Note that this work provides empirical estimators $\hat{f}_{M,b,t}$, $\hat{f}_M$, and $\hat{f}$ with noise labels $y$ and ridgeless, which makes the proof techniques quite different from [Carratino et al., 2018]. The technical difference can be found by comparing the proofs of Theorem 1 in this paper and Lemma 9 in [Carratino et al., 2018].

### 4.2 Random Features Error

The ridgeless RF predictor (3) characterizes different behaviors depending on the relationship between the number of random features $M$ and the number of samples $n$.

**Theorem 2** (Overparameterized regime $M \geq n$). When $M \geq n$, the random features error can be bounded by

$$\mathbb{E} \mathcal{E}(\hat{f}_M) - \mathcal{E}(\hat{f}) \lesssim \frac{(\alpha + c_1)\| f^* \|^2_K}{M},$$

where the constant $\alpha \propto \frac{M}{n-M}$.

In the overparameterized case, the ridgeless RF predictor is an unbiased estimator of the ridgeless kernel predictor and RF prediction can interpolate the dataset. The error term in the above bound is the variance of the ridgeless RF predictor. As shown in Figure 1 (a), the variance of ridgeless RF estimator explodes near the interpolation as $M \to n$, leading to the double descent curve that has been well studied in [Mei and Montanari, 2019; Jacot et al., 2020].

Obviously, a large number of random features in the overparameterized domain can approximate the kernel method well [Rahimi and Recht, 2007; Rudi and Rosasco, 2017], but it introduces computational challenges, i.e. $O(nM)$ time complexity for $M > n$. To reach good tradeoffs between statistical properties and computational cost, it is rather important to investigate the approximation of ridgeless RF predictor in the underparameterized regime.

**Theorem 3** (Underparameterized regime $M < n$). When $M < n$, under Assumption 1, the ridgeless RF estimator $\hat{f}_M$ approximates a kernel ridge regression estimator $\hat{f}_\lambda = K(x, X)(K + \lambda I)^{-1} y$ by

$$\mathbb{E} \mathcal{E}(\hat{f}_M) - \mathcal{E}(\hat{f}_\gamma) \lesssim \frac{\text{Tr}(K)^4\| f^* \|^2_K}{M^6} + \frac{(\alpha + c_1)\| f^* \|^2_K}{M},$$

where $K = K(X, X)$ is the kernel matrix, $\alpha \propto \frac{n}{n-M}$ and the regularization parameter $\lambda$ is the unique positive number satisfying

$$\text{Tr} [K(K + \lambda I)^{-1}] = M/n.$$
The above bound estimates the approximation between the ridgeless RF estimator \( \tilde{f}_M \) and the ridge regression \( f_\lambda \), such that the ridgeless RF essentially imposes an implicit regularization. For the shift-invariant kernels \( K(x, x') = h(||x - x'||) \), i.e. Gaussian kernel \( K(x, x') = \exp(-||x - x'||^2/(2\sigma^2)) \), the trace of matrix is a constant \( \text{Tr}(K) = n \). The number of random features needs \( M \geq \Omega(n^{0.75}) \) to achieve the convergence rate \( O(1/\sqrt{n}) \) for Theorem 3. Note that \( \tilde{N}(\lambda) := \text{Tr} \left[ (K + \lambda I)^{-1} \right] \) is known as the empirical effective dimension, which has been used to control the capacity of hypothesis space [Caponnetto and De Vito, 2007] and sample points via leverage scores [Rudi et al., 2018]. Theorem 3 states the implicit regularization effect of random features, of which the regularizer parameter is related to the features dimensional \( M \).

Together with Theorem 2, Theorem 3 and Figure 1(a), we discuss the influence from different feature dimensions \( M \) for ridgeless RF predictor \( f_M \):

1) In the underparameterized regime \( M < n \), the regularization parameter is inversely proportional to the feature dimensional \( \lambda \sim 1/M \). The effect of implicit regularity becomes greater as we reduce the features dimensional, while the implicit regularity reduces to zero as \( M \to n \). As the increase of \( M \), the test error drops at first and then rises due to overfitting (or explored variance).

2) At the interpolation threshold \( M = n \), the condition \( \text{Tr} \left[ (K + \lambda I)^{-1} \right] = M \) leads to \( \lambda = 0 \). Thus, \( M = n \) not only split the underparameterized regime and the underparameterized regime, but also is the start of implicit regularization for ridgeless RF. At this point, the variance of ridgeless RF predictor explodes, leading to double descent curve.

3) In the overparameterized case \( M > n \), the ridgeless RF predictor is an unbiased estimator ridgeless kernel predictor, but the effective ridge goes to zero. As the number of random features increases, the test error of ridgeless RF drops again.

**Remark 2 (Excess risk bounds).** From (6), one can derive the entire excess risk bound for ridgeless RF-SGD \( \|\tilde{f}_{M,t} - f^*\| \) by using the existing ridge regression bound \( \|f_\lambda - f^*\| \) in [Bartlett et al., 2005; Caponnetto and De Vito, 2007] for Theorem 3, and ridgeless regression bound \( \|\tilde{f} - f^*\| \) in [Liang and Rakhlin, 2020] for Theorem 2. These risk bounds usually characterized the bias-variance decomposition with \( \|f - f^*\|_2^2 \) \( L^2(\mu) \), where the bias usually can be bounded by \( B \leq \frac{1}{2} \text{Tr}(K) \). For the conventional kernel methods, the kernel matrix is fixed and thus \( \text{Tr}(K) \) is a constant.

## 5 Random Features with Tunable Kernels

By noting that \( \text{Tr}(K) \) play a dominate role in random features error in Theorem 3 and bias, we thus design a tunable kernel learning algorithm by reducing the trace \( \text{Tr}(K) \). We first transfer the trace to trainable form by random features

\[
\text{Tr}(K) \approx \text{Tr} \left[ (\phi(X)\phi(X)^\top) \right] = ||\phi(X)||_F^2,
\]

where \( \phi : \mathcal{X} \to \mathbb{R}^M \) depends on the spectral density \( \pi \) according to (3) and \( ||\cdot||_F \) is the squared Frobenius norm for a matrix. Since \( ||\cdot||_F \) is differentiable w.r.t. \( \Omega \), we thus can optimize the kernel density with backpropagation. For example, considering the random Fourier features [Rahimi and Recht, 2007], the feature mappings can be written as

\[
\phi(x) = \frac{1}{\sqrt{M}} \cos(\Omega^\top x + b),
\]

where the frequency matrix \( \Omega = [\omega_1, \cdots, \omega_M] \in \mathbb{R}^{d \times M} \) composed \( M \) vectors drawn i.i.d. from a Gaussian distribution \( \mathcal{N}(0, \frac{1}{\sqrt{M}} I) \in \mathbb{R}^d \). The phase vectors \( b = [b_1, \cdots, b_M] \in \mathbb{R}^M \) are drawn uniformly from \([0, 2\pi]\).

Theoretical findings illustrate that smaller trace of kernel matrix \( \text{Tr}(K) \) can lead to better performance. We first propose a bi-level optimization learning problem

\[
\begin{align*}
\min_{w} \quad & \frac{1}{n} \sum_{i=1}^{n} (w^\top \phi(x_i) - y_i)^2 \\
\text{s.t.} \quad & \Omega^* = \arg \min_{\Omega} \|\phi(X)\|_F^2.
\end{align*}
\]

The above objective includes two steps: 1) given a certain kernel, i.e. the frequency matrix \( \Omega = [\omega_1, \cdots, \omega_M] \sim \pi(w) \), the algorithm train the ridgeless RF model \( w \); 2) given a trained RF model \( w \), the algorithm optimize the spectral density (the kernel) by updating the frequency matrix \( \Omega \).

To accelerate the solve of (9), we optimize \( w \) and \( \Omega \) jointly by minimize the following objective

\[
\mathcal{L}(w, \Omega) = \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \beta ||\phi(X)||_F^2,
\]

where \( \beta \) is a hyperparameter to balance the effect between empirical loss and the trace of kernel matrix. Here, the update of \( w \) is still dependent on the squared loss (thus it is still ridgeless), but the update of \( \Omega \) is related to both the squared loss and Frobenius norm.

Therefore, the time complexity for update \( w \) once is \( O(Mn) \). However, since the trace is defined on all data, the update of \( \Omega \) is relevant to all data and time complexity is \( O(nMd) \), which is infeasible to update in every iterations. Meanwhile, the kernel needn’t to update frequently, and thus we apply an asynchronous strategy for optimizing the spectral density. As shown in Algorithm 1, the frequency matrix \( \Omega \) is updated after every \( s \) iterations for the update of \( w \). The total complexity for the asynchronous strategy is \( O(nMdT/s) \).

![Algorithm 1 Ridgeless RF with Tunable Kernels (RFTK)](image-url)
Figure 1: (a) The variance factor $\alpha$ w.r.t. $M/n$, which make the variance explores near the transition point $M = n$. (b) Empirical stochastic gradient errors $\Delta_t = \frac{1}{M} \sum_{i=1}^{M} |\hat{f}_{M,b,t}(x_i) - \hat{f}_b(x)|$ of the ridgeless RF-SGD predictors on $n = 10000$ synthetic examples. (c) The test MSE of all ridgeless RF-SGD estimators.

Figure 2: (a) Test errors v.s. regulerizations. (b) Test accuracies of compared methods. (c) Training loss and the kernel trace.

6 Experiments

We use random Fourier feature defined in (8) to approximate the Gaussian kernel $K(x, x') = \exp(-\sigma^2 \|x - x'\|^2/2)$. Note that, the corresponding random Fourier feature (8) is with the frequency matrix $\Omega \sim \mathcal{N}(0, \sigma^2)$. We implement all code on Pytorch $^2$ and tune hyperparameters over $\sigma^2 \in \{0.01, \cdots, 1000\}$, $\lambda = \{0.1, \cdots, 10^{-5}\}$ by grid search.

6.1 Numerical Validations
Factors of Stochastic Gradient Methods
We start with a nonlinear problem $y = \min(-w^T x, w^T x) + \epsilon$, where $\epsilon \sim \mathcal{N}(0, 0.2)$ is the label noise, and $x \sim \mathcal{N}(0, I)$. Setting $d = 10$, we generate $n = 10000$ samples for training and 2500 samples for testing. The optimal kernel hyperparameter is $\sigma^2 = 100$ after grid search.

As stated in Theorem 1, the stochastic gradients error is determined by the batch size $b$, learning rate $\gamma$ and the iterations $t$. To explore effects of these factors, we evaluate the approximation between the ridgeless RF-SGD estimator $\hat{f}_{M,b,t}$ and the ridgeless RF estimator $\hat{f}_M$ on the simulated data. Given a batch size $b$, we tune the learning rate $\gamma$ w.r.t. the MSE over $\gamma \in \{10^3, 10^0, \cdots, 10^{-4}\}$. As shown in Figure 1 (b), (c), give the $b, \gamma$, we estimate the ideal iterations $t = \# \text{Epoch} + 10000/b$ after which the error drops slowly. As the step size $b$ increases, the learning rates $\gamma$ become larger while the iterations reduces. This coincides with the tradeoffs among these factors $b/\gamma + 1/(\gamma t)$ in Theorem 1, where the balance between $b/\gamma$ and $1/\gamma t$ leads to better performance.

Comparing Figure 1 (b) and (c), we find that: 1) After the same passes over the data (same epoch), the stochastic gradient error and MSE of the algorithms with smaller batch sizes are smaller with faster drop speeds. 2) The stochastic error still decreases after the MSE converges, where the other error terms dominates the excess risk, i.e. bias of predictors.

Double Descent Curve in Random Features
To characterize different behaviors of the random features error w.r.t. $M$, we fixed the number of training examples $n$ and changes the random features dimensional $M$. Figure 2 (a) reports test errors in terms of different ratios $M/n$, illustrating that: 1) Kernel methods with smaller regularization lead to smaller test errors, and the reason may be regularization hurts the performance when the training data is “clean”. 2) Test errors of RF predictors converges to the corresponding kernel predictors, because a larger $M$ have better approximation to the kernel. This coincides with Theorem 2 and 3 that a larger $M$ reduces the random features error. 3) When the regularization term $\lambda$ is small, the predictors leads to double descent curves and $M = n$ is the transition point dividing the underparamterized and overparameterized regions. The smaller regularity, the more obvious curve the RF predictor exhibits.

In the overparameterized case, the error of RF predictor
Converges to corresponding kernel predictor as $M$ increases, verifying the results in Theorem 2 that the variance term reduces given more features when $M > n$. In the underparameterized regime, the errors of RF predictors drop first and then increase, and finally explodes at $M = n$. These empirical results validates theoretical findings in Theorem 3 that the variance term dominates the random features near $M = n$ and it is significantly large as shown in Figure 1 (a).

Benefits from Tunable Kernel
Motivated by the theoretical findings that the trace of kernel matrix $\text{Tr}(K)$ influences the performance, we design a tunable kernel method $RFTK$ that adaptively optimizes the spectral density in the training. To explore the influence of factors upon convergence, we evaluate both test accuracy and training loss on the MNIST dataset. Compared with the exact random features (RF) and random features with stochastic gradients (RF-SGD), we conduct experiments on the entire MNIST datasets. From Figure 2 (b), we find there is a significant accuracy gap between RF-SGD and $RFTK$. Figure 2 (c) indicates the trace of kernel matrix term takes affects after the current model fully trained near 100 epochs, and it decrease fast. Specifically, since the kernel is continuously optimized, more iterations can still improve the its generalization performance, while the accuracy of RF-SGD decreases after 100 epochs because of the overfitting to the given hypothesis.

Figure 2 (b) and (c) explain the benefits from the penalty of $\text{Tr}(K)$ that optimizes the kernel during the training, avoiding kernel selection. Empirical studies shows that smaller $\text{Tr}(K)$ guarantees better performance, coinciding with the theoretical results in Theorem 1 and Theorem 3 that both stochastic gradients error and random features error depends on $\text{Tr}(K)$.

## 6.2 Empirical Results

Compared with related algorithms listed in Table 2, we evaluate the empirical behavior of our proposed algorithm $RFTK$ on several benchmark datasets. Only the kernel ridge regression makes uses of the ridge penalty $\|w\|_2^2$, while the others are ridgeless. For the sake of presentation, we perform regression algorithms on classification datasets with one-hot encoding and cross-entropy loss. We set $b = 32$ and 100 epochs for the training, and thus the stop iteration is $T = 100n/32$. Before the training, we tune the hyperparameters $\sigma^2, \lambda, \gamma, \beta$ via grid search for algorithms on each dataset. To obtain stable results, we run methods on each dataset 10 times with randomly partition such that 80% data for training and 20% data for testing. Further, those multiple test errors allow the estimation of the statistical significance among methods.

Table 1 reports the test accuracies for compared methods over classification tasks. It demonstrates that: 1) In some cases, the proposed $RFTK$ remarkably outperforms the compared methods, for example dna, segment and Sensorless. That means $RFTK$ can optimize the kernel even with a bad initialization, which makes it more flexible than the schema (kernel selection + learning). 2) For many tasks, the accuracies of $RFTK$ are also significantly higher than kernel predictors, i.e. dna, because the learned kernel is more suitable for these tasks. 3) Compared to kernel predictors, $RFTK$ leads to similar or worse results in some cases. The reason is that kernel hyperparameter $\sigma^2$ has been tuned and in these cases they are near the optimal one, and thus the spectral density is changed a little by $RFTK$.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Time</th>
<th>Density</th>
<th>Regularizer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel Ridge</td>
<td>$O(n^2)$</td>
<td>Assigned</td>
<td>Ridge</td>
</tr>
<tr>
<td>Kernel Ridgeless</td>
<td>$O(n^3)$</td>
<td>Assigned</td>
<td>Ridgeless</td>
</tr>
<tr>
<td>RF</td>
<td>$O(nM^2 + M^3)$</td>
<td>Assigned</td>
<td>Ridgeless</td>
</tr>
<tr>
<td>RF-SGD</td>
<td>$O(MbT)$</td>
<td>Assigned</td>
<td>Ridgeless</td>
</tr>
<tr>
<td>$RFTK$</td>
<td>$O(nMbT/s)$</td>
<td>Learned</td>
<td>$|\phi(X)|_F^2$</td>
</tr>
</tbody>
</table>

Table 2: Compared algorithms.
Acknowledgments
This work was supported in part by the Excellent Talents Program of Institute of Information Engineering, CAS, the Special Research Assistant Project of CAS, the Beijing Outstanding Young Scientist Program (No. BJWZYJH012019100020098), Beijing Natural Science Foundation (No. 4222029), and National Natural Science Foundation of China (No. 62076234, No. 62106257).

References


