The Annals of Statistics
2022, Vol. 50, No. 5, 2848–2871
https://doi.org/10.1214/22-AOS2212

A SIEVE STOCHASTIC GRADIENT DESCENT ESTIMATOR FOR ONLINE NONPARAMETRIC REGRESSION IN SOBOLEV ELLIPSOIDS

BY TIANYU ZHANG[a] AND NOAH SIMON[b]

Department of Biostatistics, University of Washington, [a]zty@uw.edu, [b]nrsimon@uw.edu

The goal of regression is to recover an unknown underlying function that best links a set of predictors to an outcome from noisy observations. In nonparametric regression, one assumes that the regression function belongs to a prespecified infinite-dimensional function space (the hypothesis space). In the online setting, when the observations come in a stream, it is computationally-preferable to iteratively update an estimate rather than refitting an entire model repeatedly. Inspired by nonparametric sieve estimation and stochastic approximation methods, we propose a sieve stochastic gradient descent estimator (Sieve-SGD) when the hypothesis space is a Sobolev ellipsoid. We show that Sieve-SGD has rate-optimal mean squared error (MSE) under a set of simple and direct conditions. The proposed estimator can be constructed with a low computational (time and space) expense: We also formally show that Sieve-SGD requires almost minimal memory usage among all statistically rate-optimal estimators.

1. Introduction. It is commonly of interest to understand the association between a number of features (or predictors) and a quantitative outcome. To this end, one often estimates an underlying regression function that best links these two quantities from noisy observations. More formally, suppose we obtain n samples, (X_i, Y_i), where $X_i \in \mathcal{X} \subset \mathbb{R}^p$ denotes a p-vector of features from the ith sample we observe, and $Y_i \in \mathbb{R}$ denotes the ith outcome. Further suppose that each pair (X_i, Y_i) is independently and identically distributed (i.i.d.) from a fixed but unknown distribution ρ over $\mathcal{X} \times \mathbb{R} \subset \mathbb{R}^p \times \mathbb{R}$. A common target of estimation is the conditional mean $f_\rho(X) := E_\rho[Y|X]$. Under extremely mild conditions, this conditional mean is the optimal function for predicting Y from X with regard to mean squared error. More formally,

$$f_\rho = \operatorname*{argmin}_{f \in L^2_{\rho_X}} E_\rho\big[\big(Y - f(X)\big)^2\big], \tag{1}$$

where $L^2_{\rho_X}$ is the collection of all ρ_X-mean square integrable functions and ρ_X is the marginal distribution of X. Our goal is to estimate f_ρ from our collection of observed data.

In order to make a tractable estimation of f_ρ from data, we need to make additional assumptions on its smoothness/structure: The entire $L^2_{\rho_X}$ space is too big to search within [4, 31]. We often formally assume that f_ρ belongs to a prespecified function space $\mathcal{F} \subsetneq L^2_{\rho_X}$. This $\mathcal{F}$ is known as the *hypothesis space* of the regression problem.

If $\mathcal{F}$ can be indexed by a finite-dimensional parameter set $\Theta \subset \mathbb{R}^d$, $d \in \mathbb{N}^+$, we refer to $\mathcal{F}$ as a *parametric function space* or a *parametric class*. One common parametric class is $\mathcal{F} = \{X^\top \beta \mid \beta \in \mathbb{R}^d\}$, the set of all linear functions of X. Parametric classes can impose overly restrictive assumptions on the form of the regression function that may not be realistic in practice. As such, it has become popular to assume less restrictive structure: It is common to define the hypothesis space based on constraints on derivatives, monotonicity or other

Received March 2022.
MSC2020 subject classifications. 62G08.
Key words and phrases. Stochastic gradient descent, nonparametric regression, online learning.

shape-related properties. Such an $\mathcal{F}$ is most naturally written as an infinite-dimensional subset of $L^2_{\rho_X}$. Commonly used examples of $\mathcal{F}$ in the statistics community include Hölder balls, Sobolev spaces [22, 37, 51], reproducing kernel Hilbert spaces (RKHS) [5, 11] and Besov spaces [24]. These are known as *nonparametric function spaces*, as they cannot naturally be parametrized using a finite length vector. The Sobolev ellipsoid, in particular, is a simple and useful abstraction of many important function spaces [51]. Therefore, we focus on them exclusively as the hypothesis spaces in this paper.

In this paper, we propose an estimator for *online* nonparametric regression. In online estimation, the data are seen sequentially, one sample at a time. After each sample is observed, our estimate of f_ρ must be updated, as a prediction may be required at any point in time before all the available samples are processed. In an online problem with n observations, we must sequentially construct n estimates. This is in contrast to the classical batch learning setting where we collect all the data initially and perform estimation only once. In the online setting, it is generally computationally infeasible to repeatedly refit the whole model from scratch for each new observation. Thus, online algorithms are generally carefully developed to permit more tractable *updates* after each new observation [14, 30].

An ideal estimator in online settings should be: (i) statistically rate-optimal, that is, achieve the minimax rate for estimating f_ρ over $\mathcal{F}$; and (ii) computationally inexpensive to construct/update. In this paper, we present such an online nonparametric estimator for use when the hypothesis space is a Sobolev ellipsoid, which we term the *Sieve Stochastic Gradient Descent estimator (Sieve-SGD)*. This method can be thought of as an online version of the classical projection estimator [49], where the latter is a specific example of sieve estimators [21, 42]. We use the more general term "sieve" in naming our method to emphasize its nonparametric nature and avoid confusion with the term "stochastic projection" [50]. We will show that Sieve-SGD can achieve rate-optimal estimation error for $\mathcal{F}$, a Sobolev ellipsoid, and asymptotically uses minimal memory (up to a log factor) among all rate-optimal estimators. In addition, our estimator has the same computational cost (up to a constant) as merely examining each allocated memory location every time a new sample X_i is collected. This intimates that in scenarios when our estimator has near optimal space complexity, it may also have near optimal time complexity (though formal investigation of lower bounds for time complexity in this problem is beyond the scope of the current manuscript).

The structure of our paper continues as follows. In Section 2, we briefly cover classical results for batch, nonparametric estimation in Sobolev ellipsoids, focusing on projection estimators (which motivate our method). In Section 3, we return to the online setting and explore intuition for how one might combine projection estimation and stochastic gradient descent (SGD) [7]. The latter is a well-studied method that has been applied fruitfully to online parametric regression problems. This will help motivate our proposed method, which as we will see, can be thought of as an SGD estimator with a parameter space of increasing dimension. In Section 4, we discuss existing nonparametric SGD estimators, and identify some notable drawbacks of current methods. In Section 5, we introduce the formal construction of Sieve-SGD and analyze its computational expense. From there, we show that our estimator has a dramatically smaller "dimension" than existing methods and discuss how this helps to reduce the computational expense. In Section 6, we give a theoretical analysis of the statistical properties of Sieve-SGD. In constructing our estimator, we need to decide how quickly to grow the dimension it projects onto. Under minimal assumptions, we characterize the required growth rate and learning rate for our estimator to be statistically and computationally (near) optimal. We will also investigate under what conditions such an optimality result is adaptive/insensitive to our choice of the "dimension-specific learning rate." Section 7 provides simulation studies to illustrate our theoretical results. Finally, in Section 8, we have some further discussion of Sieve-SGD and possible future research directions.

Notation. In this paper, we use C to denote a generic constant that does not depend on sample size n (The value of C may be different in different parts of the manuscript). Additionally, the notation $a_n = \Theta(b_n)$ means $a_n = O(b_n)$ and $b_n = O(a_n)$. The function $\lfloor x \rfloor$ maps x to the largest integer smaller than x. For a vector $x \in \mathbb{R}^p$, $x^{(i)}$ is the ith component of x. The notation $x \vee y$ (resp., $x \wedge y$) is shorthand for $\max\{x, y\}$ (resp., $\min\{x, y\}$). The $\|\cdot\|_\infty$ norm of a continuous function f is defined as $\|f\|_\infty = \sup_{x\in\mathcal{X}} |f(x)|$, where $\mathcal{X}$ is the domain of f.

2. Batch learning and the projection estimator. In this section, we consider estimation in the classical batch setting where our estimate is constructed once after all n samples are observed. We will begin by formally introducing a Sobolev ellipsoid: This is the hypothesis space we will use throughout this manuscript. This will be followed by presenting the classical projection estimator [49].

Consider a user-specified measure ν whose support contains $\mathcal{X}$, and the corresponding square-integrable function space L^2_ν. In many interesting cases, ν can be simply taken as Lebesgue measure over $\mathcal{X}$ but it is not necessary in the general form of our theory. To define a Sobolev ellipsoid in L^2_ν, suppose we have a complete orthonormal basis $\{\psi_j, j = 1, 2, \ldots\} \subset L^2_\nu$ of L^2_ν [26]. This means:

(i) For any $f \in L^2_\nu$, there exists a unique sequence $(\theta_j)_{j=1}^\infty \in \ell^2$ such that

$$\lim_{N\to\infty} \int \left| f(z) - \sum_{j=1}^{N} \theta_j \psi_j(z) \right|^2 d\nu(z) = 0 \quad \text{(completeness)}, \tag{2}$$

where ℓ^2 is the space of square convergent series.

(ii) $\{\psi_j\}$ is an orthonormal system,

$$\int \psi_i(z)\psi_j(z)\, d\nu(z) = \delta_{ij} \quad \text{(orthonormality)}, \tag{3}$$

where δ_{ij} is the Kronecker delta.

We define the *Sobolev ellipsoid* $W(s, Q, \{\psi_j\})$ as

$$W(s, Q, \{\psi_j\}) := \left\{ f = \sum_{j=1}^{\infty} \theta_j \psi_j \,\middle|\, \sum_{j=1}^{\infty} (\theta_j j^s)^2 \le Q^2 \right\}. \tag{4}$$

We refer to $(\theta_j)_{j=1}^\infty$ as the (general) *Fourier coefficients* of a function f. Throughout this manuscript, we assume the measure ν, basis functions ψ_j and the regularity parameter s are all known. When it is clear which ψ_j we are using, we will denote a Sobolev ellipsoid simply by $W(s, Q)$. We may also use the further simplified notation $W(s)$ because the diameter Q usually plays a secondary role in our theoretical analysis and the proposed method is adaptive to it. Intuitively, by saying a function f belongs to a Sobolev ellipsoid, we are requiring its coefficients $\{\theta_j\}$ to converge to zero faster than $j^{-(s+1/2)}$ (if not, the sum $\sum_{j=1}^\infty (\theta_j j^s)^2$ would diverge to infinity). The larger s is, the faster the decay of θ_j will be, and thus the stronger our assumption is.

Sobolev ellipsoids are popular spaces to study for two reasons: (1) They impose a useful structure for theory and computations, especially as a basic example of hypothesis spaces with finite metric entropy; and (2) Many natural spaces of regular functions are Sobolev ellipsoids. For example, if $\mathcal{X} = [0, 1]$ with ν as Lebesgue measure, then for any $s > 0$, the periodic Sobolev space

$$\mathcal{F} = \left\{ f \in L^2_\nu \,\middle|\, \int (f^{(s)}(x))^2\, dx < Q^2, f^{(k)}(0) = f^{(k)}(1), k = 0, 1, \ldots, s-1 \right\} \tag{5}$$

can be written as a Sobolev ellipsoid, using an orthogonal basis of trigonometric functions [51], Chapter 2. More generally, for many important RKHSs $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}})$, it is possible to find a set of ψ_j such that $W(s, Q, \{\psi_j\}) = \{f \in \mathcal{H} | \|f\|_{\mathcal{H}} \le Q\}$, that is, a ball in an RKHS is a Sobolev ellipsoid (see [12, 47]): Under mild conditions [44], a Mercer kernel $K(s,t) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ has the following Mercer representation:

$$K(s,t) = \sum_{j\in\mathcal{J}} \lambda_j \psi_j(s)\psi_j(t), \tag{6}$$

where $\lambda_j > 0$, $\mathcal{J}$ is at most countably infinite. And $\{\psi_j\}$ is an orthonormal system (in L^2_ν) w.r.t. some measure ν on $\mathcal{X}$, and any function $f \in \mathcal{H}$ can be written as $f = \sum_{j\in\mathcal{J}} \theta_j \psi_j$. It is also known that the RKHS-norm can be identified as $\|f\|^2_{\mathcal{H}} = \sum_{j\in\mathcal{J}} \theta_j^2 \lambda_j^{-1}$. So, a ball in the RKHS, that is, $\{f \in \mathcal{H} | \|f\|_{\mathcal{H}} \le Q\}$ is the same as a Sobolev ellipsoid spanned by $\{\psi_j\}$ when $\mathcal{J} = \mathbb{N}^+$ and $\lambda_j = j^{-2s}$. This is the case for many Sobolev-type kernels (e.g., p. 454 in [16]). When $\mathcal{J}$ is finite-dimensional (polynomial kernels) or λ_j decays exponentially fast in j (Gaussian kernel, p. 455 in [16]), a ball in the RKHS can be characterized as some “generalized” Sobolev ellipsoid.

In everything that follows, we will assume that f_ρ, our target of estimation, lives in a known Sobolev ellipsoid $W(s, Q, \{\psi_j\})$; with $\{\psi_j\}$ specified, and orthonormal w.r.t. a specified measure ν (not necessarily equal to ρ_X); and s known (we allow Q to be unknown).

The *projection estimator* is a classical estimator naturally associated with a Sobolev ellipsoid. We can treat it as a special case of general sieve estimation [21], Chapter 10: The estimates can be characterized by a sequence of finite-dimensional linear spaces of increasing dimension (the dimension increases with sample size). For any given $f \in W(s, Q)$, the magnitude of its Fourier coefficients must asymptotically decrease with j fast enough. Thus, it might be sensible to consider an estimator that discards the basis functions far into the tail. This is precisely what the projection estimator does. More formally, for a user-specified truncation level J_n, the projection estimator is given by

$$\hat{f}_{n,J_n} = \sum_{j=1}^{J_n} \hat{\theta}_j \psi_j, \tag{7}$$

where $\hat{\theta} = (\hat{\theta}_1, \dots, \hat{\theta}_{J_n})^\top$ is the solution of the least square problem:

$$\min_{\theta\in\mathbb{R}^{J_n}} \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{J_n} \theta_j \psi_j(X_i) \right)^2. \tag{8}$$

It has been shown (e.g., [49], Theorem 1.9) that when we choose $J_n = \Theta(n^{1/(2s+1)})$ the projection estimator is a rate-optimal estimator over $W(s, Q)$, that is,

$$\limsup_{n\to\infty} \sup_{f_\rho \in W(s,Q)} E\big[\|\hat{f}_{n,J_n} - f_\rho\|_2^2\big] = O\big(n^{-\frac{2s}{2s+1}}\big). \tag{9}$$

This result is usually shown in the literature for X_i equally spaced, or drawn from a uniform distribution. But in our theoretical analysis (Section 6), we allow ρ_X to be a much more general distribution.

Sieve-SGD is inspired by this (batch) projection estimator. The key here is that the number of basis functions we need to use can be dramatically smaller than the sample size, and their analytical forms do not depend on the data (usually reproducing kernel methods use basis functions “centered” at the feature vectors X_i). This possibility has been rarely explored [58] by existing nonparametric online estimation research.

3. Online learning and stochastic approximation.

We now move to the online learning setting where observations are collected sequentially from a data stream, and an estimate of our function is required after each sample. Such an infinite data stream may really exist, for example, with simulated samples as in reinforcement learning, or the stream may serve as an abstraction used with large-scale data sets where it is not favorable to handle all the samples at once. It is generally computationally prohibitive to use a method developed for the "batch" setting and completely refit it after each observation. Instead methods that iteratively update are preferred. For example, fitting a single projection estimator (solving (8)) with n observations using $J_n = n^{1/(2s+1)}$ requires computation of $\Theta(n^{1+2/(2s+1)})$. Refitting a projection estimator (from scratch) after each observation $i = 1, \dots, n$ with $J_i = \lfloor i^{1/(2s+1)} \rfloor$ would require an accumulated computation of $\sum_{i=1}^{n} i^{1+2/(2s+1)} = \Theta(n^{2+2/(2s+1)})$. This scales worse than quadratically in n. Our goal in the online nonparametric setting is to find a statistically rate-optimal estimator whose computation scales only slightly worse than linearly in n.

Online learning has been thoroughly studied for parametric $\mathcal{F}$. Many proposed methods are based on the concept of *stochastic approximation* [30]. One of the most popular methods in stochastic approximation is Stochastic Gradient Descent (SGD) [7]. In the parametric setting, SGD gives a statistically rate-optimal estimator $\hat{f}_n$ whose population mean-squared error $E\|\hat{f}_n - f_\rho\|^2_{L^2_{\rho_X}}$ is of order $O(n^{-1})$ [2, 3, 17]. Both vanilla SGD and its variants have been applied to general convex loss functions and are shown to be statistically rate-optimal under mild conditions [14, 38].

3.1. *Parametric SGD.*

To motivate stochastic optimization in the nonparametric setting, we first give more details on SGD for parametric classes. Here, we consider a specific class of functions $\mathcal{F} = \{f = \sum_{j=1}^{d} \beta^{(j)} \psi_j, \beta \in \mathbb{R}^d\}$ for a set of prespecified basis functions $\psi_j : \mathbb{R}^p \to \mathbb{R}$, $j = 1, \dots, d$. We use this example to illustrate the principle of (parametric) SGD. Solving $\operatorname{argmin}_{f \in \mathcal{F}} E[(Y - f(X))^2]$ reduces to solving

$$\min_{\beta \in \mathbb{R}^d} \ell(\beta) := \min_{\beta \in \mathbb{R}^d} E\left[\left(Y - \sum_{j=1}^{d} \beta^{(j)} \psi_j(X)\right)^2\right]. \tag{10}$$

We assume the minimizer of $\ell(\beta)$ exists and denote it as β^*.

If we knew the true joint distribution ρ of (X, Y) (which never happens in practice), then equation (10) is just a numerical optimization problem, which does not involve data. We could use gradient descent to solve it. The gradient of ℓ at any point β is

$$\nabla \ell(\beta) = -2E\left[\left(Y - \sum_{j=1}^{d} \beta^{(j)} \psi_j(X)\right)(\psi_1(X), \dots, \psi_d(X))^\top\right]. \tag{11}$$

Thus, the gradient descent updating rule one could use is

$$\begin{aligned} &\hat{\beta}_0 = 0, \\ &\hat{\beta}_n = \hat{\beta}_{n-1} - \gamma_n \nabla \ell(\hat{\beta}_{n-1}), \end{aligned} \tag{12}$$

where $\{\gamma_n\}$ is a prespecified sequence of step-sizes (or learning rate) and $\hat{\beta}_n \in \mathbb{R}^d$ is the sequence of approximations of β^*.

In practice, we do not know the joint distribution ρ: we must use data to estimate β^*. In the framework of SGD, this is done by using the data to get unbiased estimates of the gradients and substituting the estimates into our updating rule (12). In particular, we note that

$\widehat{\nabla\ell(\beta)} := -2(Y_i - \sum_{j=1}^{d} \beta^{(j)}\psi_j(X_i))(\psi_1(X_i), \dots, \psi_d(X_i))^\top$ is an unbiased estimator of the gradient $\nabla\ell(\beta)$ based on one sample. This results in the SGD updating rule:

$$\begin{aligned}
&\hat{\beta}_0 = 0, \\
&\hat{\beta}_n = \hat{\beta}_{n-1} - \gamma_n \widehat{\nabla\ell(\hat{\beta}_{n-1})} \\
&\quad = \hat{\beta}_{n-1} + 2\gamma_n \Bigg(Y_n - \sum_{j=1}^{d} \hat{\beta}_{n-1}^{(j)} \psi_j(X_n) \Bigg) \big(\psi_1(X_n), \dots, \psi_d(X_n)\big)^\top .
\end{aligned} \tag{13}$$

So, our estimator $\hat{f}_n$ of f_ρ has the following functional update rule, derived from (13):

$$\hat{f}_n = \hat{f}_{n-1} + 2\gamma_n\big(Y_n - \hat{f}_{n-1}(X_n)\big) \sum_{j=1}^{d} \psi_j(X_n)\psi_j . \tag{14}$$

Here, we have shifted to considering our estimator $\hat{f}_n$ as a function, rather than thinking about $\hat{\beta}_n$ a vector of coefficients. This will be important in the nonparametric setting.

3.2. *From parametric SGD to nonparametric SGD.* In this subsection, we discuss the intuition in moving from SGD in a finite-dimensional parametric space to an infinite-dimensional space.

We assume $f_\rho \in W(s, Q, \{\psi_j\}) \subset L^2_\nu$. Since ψ_j is a complete basis of L^2_ν, we can always find an expansion of f_ρ w.r.t. $\{\psi_j\}$:

$$f = \sum_{j=1}^{\infty} \theta_j \psi_j . \tag{15}$$

In Section 3.1, we already discussed the SGD updating rule for a d-dimensional model $f(X) = \sum_{j=1}^{d} \beta^{(j)}\psi_j(X)$. In the nonparametric scenario, the number of basis function is increased from d to infinity: This causes problems if care is not taken.

One might naturally consider applying a direct analog to the finite-dimensional SGD rule (14) here (we omit the constant 2):

$$\hat{f}_n = \hat{f}_{n-1} + \gamma_n\big(Y_n - \hat{f}_{n-1}(X_n)\big) \sum_{j=1}^{\infty} \psi_j(X_n)\psi_j . \tag{16}$$

Unfortunately, we run into a severe problem: The series $\sum_{j=1}^{\infty} \psi_j(X_n)\psi_j$ does not converge even if all ψ_j are bounded (it is direct to check when $X_n = 0$ and ψ_j are trigonometric functions). However, as we assume $f_\rho \in W(s)$, we know that those higher order components, ψ_j, $j \gg 1$ should have very small coefficients. Thus, one natural solution is to use a different step-size per component that decreases as j increases. By doing "less fitting" for larger j, we can stabilize our update (smaller variance), and yet might still appropriately fit the overall regression function. In particular, one might modify (16) to

$$\hat{f}_n = \hat{f}_{n-1} + \gamma_n\big(Y_n - \hat{f}_{n-1}(X_n)\big) \sum_{j=1}^{\infty} t_j \psi_j(X_n)\psi_j , \tag{17}$$

where the component-specific (or dimension-specific) learning rates $t_j > 0$ are monotonically decreasing with j. For t_j decreasing fast enough and uniformly bounded ψ_j, the function series $\sum_{j=1}^{\infty} t_j\psi_j(X_n)\psi_j$ is absolutely convergent. Now (17) becomes a sensible nonparametric SGD updating rule when the hypothesis space is a Sobolev ellipsoid. In addition, sometimes $\sum_{j=1}^{\infty} t_j\psi_j(X_n)\psi_j$ actually has a simply characterized closed form (in particular, for many

RKHS). In such cases, (17) results in a relatively straightforward algorithm. More specifically, one can show that when $t_j = j^{-2s}$ and $\gamma_n = \Theta(n^{-1/(2s+1)})$, the average

$$\bar{f}_n := \frac{1}{n}\sum_{i=1}^{n} \hat{f}_i \tag{18}$$

is a rate-optimal estimator of $f_\rho \in W(s)$. This was recently proposed (though motivated quite differently) in the context of RKHS hypothesis spaces [13]. The authors there engage directly with the *kernel function* for the RKHS (though their updating rule is equivalent to equation (17)). This will be discussed in more detail in Section 4. Our work engages and extends these ideas (in combination with sieve estimation) to form a statistically rate-optimal online estimator with greatly reduced computational and memory complexity.

4. Related work. Nonparametric online learning is a relatively new area. A few remarkable functional stochastic approximation algorithms have been proposed in the last two decades [9, 13, 34, 48, 55]. The key ideas in that body of work are intimately related to those mentioned in Section 3.2, however, they engage those ideas from a different direction: They assume that the hypothesis function space $\mathcal{F}$ is an RKHS, and then leverage the kernel in that space. In particular, the RKHS structure makes it possible to take the gradient of the evaluation functional $L_x(f) := f(x)$, with respect to the RKHS inner product $\langle\cdot,\cdot\rangle_K$, that is,

$$L_x(f + \epsilon g) = f(x) + \epsilon g(x) = L_x(f) + \epsilon\langle g, K_x\rangle_K. \tag{19}$$

Thus, $K_x(\cdot) := K(x,\cdot) \in \mathcal{F}$ is the gradient of functional L_x at f. However, one cannot do this in the general $L^2_{\rho_X}$ space where the evaluation functional is no longer a bounded operator.

Thus, when $\mathcal{F}$ is an RKHS associated with kernel K, there is a simple nonparametric SGD updating rule for minimizing $E[(Y - f(X))^2]$ over $\mathcal{F}$:

$$\begin{aligned} &\hat{f}_0 = 0, \\ &\hat{f}_n = \hat{f}_{n-1} + \gamma_n\big(Y_n - \hat{f}_{n-1}(X_n)\big)K(X_n,\cdot). \end{aligned} \tag{20}$$

Here, because the gradient is taken with respect to the RKHS inner product, we do not have the issue encountered in (16) where our series representation of the "gradient" actually did not converge. In fact, by working with the RKHS inner product, we implicitly carry out the proposal of Section 3.2 and decrease the component-specific learning rate of higher order terms. More specifically, we usually have the Mercer expansion of the kernel function,

$$K(x,z) = \sum_{j=1}^{\infty} t_j \psi_j(x)\psi_j(z), \tag{21}$$

with respect to an orthonormal basis $\{\psi_j\}$ of L^2_ν. For many common RKHSs, we have $t_j = \Theta(j^{-u})$ for some $u > 1$ [16], Appendix A. Thus, (20) corresponds precisely to the previously discussed update (17). Most popular RKHS have a kernel $K(x,z)$ with a closed-form representation, and thus, rather than having to store an infinite number of coefficients, after n steps the estimate from (20) would take the form of a weighted linear combination of n kernel functions [13]:

$$\hat{f}_n = \sum_{i=1}^{n} b_i K(X_i,\cdot). \tag{22}$$

Although such estimators (with one more Polyak averaging step (18)) have been shown to give rate-optimal MSE [13], updating them with a new observation (X_{n+1}, Y_{n+1}) usually involves evaluating n kernel functions at X_{n+1}, with computational expense of order $\Theta(n)$.

This is in contrast with the constant update cost of $\Theta(d)$ in parametric SGD, where d is the dimension of the parameter space. Thus, the time expense of nonparametric kernel SGD will accumulate at order $\Theta(n^2)$. Also, one is required to store the n feature-values $\{X_i\}_{i=1}^n$ to evaluate the estimator, which results in $\Theta(n)$ space expense. This relatively large time and space complexity indicates that those kernel-based SGD estimators are not ideal as methods that are nominally designed to deal with large data sets.

There has been some work in the literature aimed at improving the computational aspects of kernel SGD methods [29, 32, 43]. These methods select a subset of the n kernel functions centered at the feature vectors and use them as basis functions to construct estimators (which is also related to Nyström projection). Neither the statistical performance nor the computational expense of the aforementioned work is guaranteed to be optimal. Also, the theoretical analysis in that work typically requires the noise variable to have extremely light tails.

There has also been recent work [9, 34] aimed at improving kernel SGD algorithms by leveraging approximate second-order information (SGD only uses the first-order information). The estimator in [34] is shown to give rate-optimal MSE and has better (theoretical) computational efficiency than the vanilla kernel SGD mentioned above. However, these algorithms are usually dramatically more complicated and have a couple of hyperparameters that need to be tuned.

There is another branch of research also called "online nonparametric regression" that engages with a different but related setting [18, 40]. They do not aim to directly minimize the (population) generalization error. Their definition of "regret" is based on comparing a running average of prediction error and the empirical risk minimizer's training error. Formally, it is defined as $\sum_{i=1}^n l(\widehat{Y}_i, Y_i) - \inf_{f\in\mathcal{F}} \sum_{i=1}^n l(f(X_i), Y_i)$, where $\hat{Y}_i$ is the prediction of the algorithm based on the first $i-1$ observations, l is a convex loss and $\mathcal{F}$ is the hypothesis function space. While this is an interesting area of research, and might be used to engage with population generalization error (using online-to-batch techniques), it is a less direct treatment than what we are considering in this work.

5. Online learning and the projection estimator: Sieve-SGD.

In this section, we combine ideas from the projection estimator (in the batch learning setting), and stochastic gradient descent to develop an estimator that is suitable for online nonparametric regression. The estimator we will propose achieves the minimax rate for MSE over a Sobolev ellipsoid, and is much more computationally efficient than standard kernel SGD methods.

As a reminder, the kernel SGD estimator based on (20) has minimax rate optimal MSE. When $\sum_{j=1}^\infty t_j\psi_i(s)\psi_j(t)$ has an available closed form, it requires $\Theta(n)$ memory and has $\Theta(n^2)$ time expense for sequentially processing n observations. We aim to improve this, and furthermore, to propose an effective estimator appropriate for cases where $\sum_{j=1}^\infty t_j\psi_i(s)\psi_j(t)$ has no closed form.

Motivated by the projection estimator, we opt to use truncated series in the updating rule, modifying (17) (or equivalently (20)) to get

$$\hat{f}_n = \hat{f}_{n-1} + \gamma_n\big(Y_n - \hat{f}_{n-1}(X_n)\big)\sum_{j=1}^{J_n} t_j\psi_j(X_n)\psi_j. \tag{23}$$

Here, J_n is an increasing sequence of integers that grows as we collect more observations. When J_n is larger, the updating rule (23) is closer to our original form (17), however, a smaller J_n is desirable because it results in a lower computational expense. Part of our task is identifying a "minimal" J_n that still maintains favorable statistical properties.

It turns out there are 2 ways to control the bias-variance tradeoff. One can use the truncation level J_n, or the component specific step-sizes t_j. If the truncation level is used, then the methodology is more analogous to a projection estimator. In this case, so long as t_j is not too large (controlling the variance in the dynamics of SGD) or too small (controlling the bias term), we would get (near) optimal statistical performance for a relatively wide range of choices for t_j. We give formal results related to this in Section 6.3. If, instead, we control the bias-variance tradeoff using t_j then our estimator is more analogous to kernel-SGD. In this case, the first-order terms for bias and variance are determined by the sequence $\{t_j\}$ and J_n just needs to be sufficiently large (such that we do not induce excess bias). We give formal results for this in Section 6.2. This second way to control the tradeoff is similar to using a truncated basis for penalized regression in the batch learning setting. For example, in [23] and [54], Section 5.2, the authors propose to estimate f_ρ by solving a penalized regression spline problem, where they use a reduced spline basis for improved computation (rather than including a knot at every point). The bias/variance trade-off there is controlled via the penalty: They are careful to include enough basis elements so that the use of a reduced basis only contributes a second-order term to the bias.

We will next give details of our proposal. For this proposal, we are assuming that $f_\rho \in W(s, Q, \{\psi_j\}) \subset L^2_\nu$, and that s is known. Based on this, we choose our component specific step-sizes as $t_j = j^{-2\omega}$ (for some $1/2 < \omega \leq s$). We also define

$$K^{\omega}_{x,J_n}(\cdot) = \sum_{j=1}^{J_n} j^{-2\omega}\psi_j(x)\psi_j(\cdot). \tag{24}$$

In addition to simplifying exposition, this notation relates our method to (21). The function $K^{\omega}_{x,J_n}(\cdot)$ can be seen as a truncated approximation of the kernel function

$$K^{\omega}_{x,\infty}(\cdot) = \sum_{j=1}^{\infty} j^{-2\omega}\psi_j(x)\psi_j(\cdot) \tag{25}$$

that drops all the ψ_j with index $j > J_n$.

5.1. *Sieve stochastic gradient descent.* We now explicitly give our Sieve Stochastic Gradient Descent algorithm (Sieve-SGD) for estimation of f_ρ in a Sobolev ellipsoid $W(s, Q, \{\psi_j\})$.

Let $J_n = \lfloor n^\alpha \rfloor$ for some specified $\alpha > 0$ and $\omega \in (1/2, s]$. The parameter α is usually taken between $(2s+1)^{-1}$ and 1. We use γ_i to denote the step-size (learning rate) of the ith update and typically choose $\gamma_i = \Theta(i^{-1/(2s+1)})$. The construction of Sieve-SGD estimators is formally described in Algorithm 1.

We refer to the function $\bar{f}_i$ as the *Sieve-SGD estimate* of f_ρ. We will later show that $\bar{f}_i$ has rate-optimal MSE for estimating any $f_\rho \in W(s)$. In Algorithm 1, we use the language of "updating a function," but in practice one would update the coefficient vector corresponding to the functions $\{\psi_j\}_{j=1}^{J_n}$. In Appendix A [60], we attach a presentation of the algorithm that works directly with the coefficients. This estimator is quite simple, though it does require apriori selection/knowledge of $\{\psi_j\}$ and s (which can be done using a held-out validation set in practice). Unfortunately, showing its favorable statistical properties (in Section 6) is somewhat more complex.

5.2. *Computational expense.* After examining the updating rule above, one can see that $\hat{f}_i$ has the form:

$$\hat{f}_i(x) = \sum_{j=1}^{J_i} b_j\psi_j(x). \tag{28}$$

Proposed Algorithm 1: Sieve Stochastic Gradient Descent (Sieve-SGD)

```
    Set $\alpha, \omega > 0$, step-size $\{\gamma_i\}$ and basis functions $\{\psi_j\}$. Initialize $\bar{f}_0 = \hat{f}_0 = 0$.
    For $i = 1, 2, \ldots$:
1. Calculate $J_i = \lfloor i^{\alpha} \rfloor$, collect data pair $(X_i, Y_i)$.
2. Update $\hat{f}_i$:
   (26)  $\hat{f}_i = \hat{f}_{i-1} + \gamma_i(Y_i - \hat{f}_{i-1}(X_i)) \sum_{j=1}^{J_i} j^{-2\omega} \psi_j(X_i) \psi_j$
               $= \hat{f}_{i-1} + \gamma_i(Y_i - \hat{f}_{i-1}(X_i)) K^{\omega}_{X_i, J_i}$
3. Polyak averaging: Update $\bar{f}_i$ by
   (27)  $\bar{f}_i = \frac{1}{i+1} \sum_{k=0}^{i} \hat{f}_k$
               $\left(= \frac{i}{i+1} \bar{f}_{i-1} + \frac{1}{i+1} \hat{f}_i\right)$
```

This requires storing the coefficients $\{b_j\}_{j=1}^{J_i}$ in memory. The main computational burden of each update step is calculating $\hat{f}_{i-1}(X_i)$ and $K^{\omega}_{X_i,J_i}$. Both require evaluating J_i basis functions at X_i. Thus, the computational expense of the "Update $\hat{f}_i$" step above is of order $J_i = \Theta(i^{\alpha})$ when we take evaluating one basis function at one point as $O(1)$. And the total expense of processing n samples is of order $\Theta(n^{1+\alpha})$. The space expense is of the same order $\Theta(i^{\alpha})$: We need only store coefficients of J_i basis functions. In Section 6.4, we will show that, under mild conditions, this memory complexity is near optimal among all estimators with rate-optimal MSE.

This compares favorably with standard kernel SGD (22), which uses i basis functions at step i: Our estimator uses fewer when $\alpha < 1$; as we will show later, α can be taken as small as $(2s+1)^{-1}$, which is a substantial improvement. In practice, the parameter α can either be selected based on our assumptions about s (belief on the smoothness of f_{ρ}) or heuristically tuned for empirical performance.

5.3. *General convex loss.* Although the main focus of this paper is regression with squared-error loss, our algorithm has a straightforward extension to general convex loss. Suppose we want to minimize the population loss,

$$E[\ell(Y, f(X))], \tag{29}$$

over all functions $f \in W(s, Q, \{\psi_j\})$ and the loss function $\ell(Y, \cdot)$ is convex for each Y. In this case, we need only modify step 2 of the Sieve-SGD estimator in Section 5.1. Given loss $\ell(\cdot, \cdot)$, the updating rule for $\hat{f}_i$ takes the general form:

2') Update $\hat{f}_i$:

$$\hat{f}_i = \hat{f}_{i-1} + \gamma_i \frac{\partial}{\partial v} \ell(u, v)\bigg|_{(Y_i, \hat{f}_{i-1}(X_i))} K^{\omega}_{X_i, J_i}. \tag{30}$$

For example, with $Y = \{1, -1\}$ considering nonparametric logistic regression, the loss function one would use is $\ell(Y, f(X)) = \log(1 + \exp(-Yf(X)))$. In this case, we have

$$\frac{\partial}{\partial v}\ell(u, v)\bigg|_{(Y_i, \hat{f}_{i-1}(X_i))} = \big(1 + \exp\big(Y_i \hat{f}_{i-1}(X_i)\big)\big)^{-1} Y_i \in \mathbb{R}. \tag{31}$$

Theoretical guarantees for Sieve-SGD using general convex loss are beyond the scope of this paper. However, in Section 7 we provide simulated experiments that show the empirical performance of Sieve-SGD for nonparametric logistic regression. These empirical results intimate that perhaps similar theoretical guarantees to those shown for squared-error-loss hold in a more general setting.

5.4. *Choice of basis functions and multivariate problems.* In practice, there are many available choices of univariate ψ_j that in general lead to interesting (Sobolev-type) hypothesis spaces. For example,

$$\psi_1(x) = 1, \qquad \psi_j = \sqrt{2}\cos\big((j-1)\pi x\big), \quad \text{for } j \geq 2. \tag{32}$$

This set of basis functions are the "eigenfunctions" of Sobolev spaces over $[0, 1]$ (Appendix A.2 in [39]), which means they are orthogonal w.r.t. to the Lebesgue inner product and the Sobolev inner product simultaneously. The corresponding Sobolev ellipsoid does not impose periodicity assumptions of f_ρ and is very convenient to use in practice. Among many other choices, we can also use algebraic polynomials, or a combination of algebraic polynomials and (periodic) Fourier basis [15].

In most applications, the covariate X_i's take value in $\mathbb{R}^p$ where $p > 1$. In some situations, there are some "canonical" choices of basis function $\psi(x) : \mathbb{R}^p \to \mathbb{R}$ that people might use for identifying their (multivariate) Sobolev ellipsoid. For example, when considering estimating a function on a sphere $\mathbb{S}^2$, ψ_j could be taken as the orthonormal spherical harmonics ([27, 36]).

In many situations, the basis functions ψ_j can conveniently be taken as a tensor product of a one-dimensional complete basis, and Sieve-SGD can be directly applied in this multivariate setting. If we are using a univariate Sobolev ellipsoid to represent a ball in an RKHS, then the ellipsoid defined by the tensor product basis will correspond to a ball in the RKHS spanned by the tensor product kernel (though care needs to be taken with the ordering of the basis vectors [59]). Some technical details and numerical examples on this can be found in Appendix B and the references therein. In all of these cases, our theoretical results will hold (so long as the function f_ρ belongs to the specified space).

A common alternative approach in multivariate problems is to impose some additional structure on the hypothesis space to make estimation more tractable. This is particularly true when the feature dimension p is large. One popular model is the nonparametric additive model [25, 46, 57], which is thought to effectively balance model flexibility and interpretability. For $x \in \mathbb{R}^p$, we might consider assuming/imposing an additive structure on the regression function:

$$f_\rho(x) = \sum_{k=1}^{p} f_{\rho,k}\big(x^{(k)}\big), \tag{33}$$

where each of the component functions $f_{\rho,k}$ belong to a Sobolev ellipsoid $W_k(s_k, Q_k, \{\psi_{jk}\})$. For ease of exposition, in (33), we assume $E[Y] = 0$ to avoid the need for a common intercept term. For a more complete version with common intercept, see Appendix B. For a fixed dimension p, when all $W_k = W^*$ (for some Sobolev ellipsoid W^*), the minimax rate for estimating such an additive model is identical (up to a multiplicative constant p) to the minimax rate in the analogous one-dimension nonparametric regression problem with the same hypothesis space W^* [41, 46]. For the additive model (33), the updating rule (26) of Sieve-SGD could be replaced by

$$\hat{f}_i = \hat{f}_{i-1} + \gamma_i \left(Y_i - \sum_{k=1}^{p} \hat{f}_{i-1,k}\big(X_i^{(k)}\big) \right) \sum_{k=1}^{p} \sum_{j=1}^{J_{ik}} j^{-2\omega_k} \psi_{jk}\big(X_i^{(k)}\big) \psi_{jk}. \tag{34}$$

Here, J_{ik} is the truncation level of kth dimension when the sample size $= i$ and $\hat{f}_{i-1,k}$ is the estimate of $f_{\rho,k}$. Most of the theory that we develop in Section 6 could apply here.

6. Generalization guarantees of Sieve-SGD.

In this section, we show Sieve-SGD achieves the minimax rate for nonparametric estimation in Sobolev ellipsoids under mild assumptions. We also show that Sieve-SGD has near minimal memory complexity among all estimators that are minimax rate-optimal for estimation in a Sobolev ellipsoid. The conditions on the hyperparameters can be used as theoretical guidance when applying Sieve-SGD to real data problems.

6.1. *Model assumptions.*

We begin by listing the conditions we will require in our proof. They reflect different aspects of the problem: independent observations (A1), distribution of X (A2), the hypothesis space assumed for f_ρ (A3) and tail behavior of the noise (A4). These conditions ensure the MSE rate-optimality of Sieve-SGD.

A1 (i.i.d. data) The data points $(X_n, Y_n)_{n\in\mathbb{N}} \in \mathcal{X} \times \mathbb{R}$ are independently, identically sampled from a distribution $\rho(X, Y)$.

A2 (feature distribution) Let ν be a user-specified measure that is strictly positive on $\mathcal{X}$. Assume the distribution of feature X, ρ_X, is absolutely continuous w.r.t. ν. Let $p_X = d\rho_X/d\nu$ denote its Radon–Nikodym derivative. We assume for some u, ℓ such that $0 < \ell < u < \infty$:

$$\ell \le p_X(x) \le u \quad \text{for all } x \in \mathcal{X}$$

A3 (Sobolev ellipsoid) Let $\{\psi_j\}_{j=1}^{\infty}$ be a set of uniformly bounded ($\|\psi_j\|_\infty \le M$), continuous, orthonormal basis of L_ν^2. We assume the regression function f_ρ falls in a Sobolev ellipsoid, with basis functions given by $\{\psi_j\}$, that is, for some $s > 1/2$, $Q < \infty$,

$$f_\rho \in W\big(s, Q, \{\psi_j\}\big) \tag{35}$$

A4 (noise) One of the following two assumptions is satisfied by the noise variable $\epsilon = Y - f_\rho(X)$:

- ϵ is bounded by some C_ϵ almost surely.
- ϵ is independent of the features, X, and has a finite second moment $E[\epsilon^2] = C_\epsilon^2$.

Note 1: The lower bound requirement of p_X in A2 may be due to artifacts in our proof. In reality, especially when the dimension of our feature-space X is large, such a requirement may be hard to satisfy. According to our simulation results, Sieve-SGD still achieves the minimax rate even when ρ_X has a strictly smaller support than ν. As compared with other work in nonparametric online learning [13, 48, 55], our assumptions are more direct. We discuss this in detail later in this section.

Note 2: In Assumption A3, we do not require ψ_j to be orthonormal w.r.t. ρ_X (and it is in general not true), but only require them to be orthonormal w.r.t. the known measure ν. In many cases, ν might be taken to be Lesbesgue (or uniform) measure over a domain containing $\mathcal{X}$, as this is the canonical measure under which function spaces such as Sobolev spaces and Besov spaces are defined. As long as the density function p_X satisfies A2, using the nonorthonormal (w.r.t. ρ_X) basis functions, ψ_j, does not prevent Sieve-SGD from having rate-optimal MSE.

Note 3: It is a common convention to think about a hypothesis space where the Sobolev (type) norm of the regression function is bounded by a constant Q (A3), rather than just $< \infty$. Such a bounded space has a finite minimax rate: the exponent is determined by s, and the constant is proportional to Q (see also note 5 under Theorem 6.1). We also would like to note that the proposed algorithm does not use radius Q at any point and the theoretical guarantee is adaptive w.r.t. Q. (More specifically, the final bounds given in Appendix D.3 and Lemma D.1 have $\|f_\rho\|_K$, which can be thought of as the effective value for Q, on the right-hand side.)

6.2. *Rate optimality when $t_j = j^{-2s}$.* In this section, we present the rate-optimality results of Sieve-SGD when we choosing the component-specific learning rate to be $t_j = j^{-2s}$ (or $\omega = s$ in (26)). In this setting, our theoretical analysis treats Sieve-SGD as a truncated-version (in the basis expansion domain) of a "correct" kernel SGD procedure (we will discuss the "incorrect" version very soon in Section 6.3, and show that it can actually still have favorable statistical and computational properties). Here is the main result in this setting.

THEOREM 6.1. *Assume* A1–A4. *Set the component-specific learning rate as $t_j = j^{-2s}$. Also, set the overall learning rate to be $\gamma_n = \gamma_0 n^{-1/(2s+1)}$ with $\gamma_0 \leq (2M^2\zeta(2s))^{-1}$, where $\zeta(k) = \sum_{i=1}^{\infty} i^{-k}$. Choose the number of basis functions to be $J_n \geq n^\alpha \log^2 n \vee 1$ for an arbitrary $\alpha \geq (2s+1)^{-1}$.*

Then the MSE of Sieve-SGD (27) *converges at the following rate*:

$$E\|\bar{f}_n - f_\rho\|^2_{L^2_{\rho_X}} = O\big(n^{-\frac{2s}{2s+1}}\big). \tag{36}$$

This implies that Sieve-SGD is a minimax rate-optimal estimator of f_ρ over $W(s, Q, \{\psi_j\})$.

We now discuss our assumptions and results in more detail, and relate them to what is currently in the literature.

Note 1: In the analysis of many reproducing kernel methods for nonparametric estimation [13, 48, 56], the spectrum of the *covariance operator* plays an important role in controlling the statistical behavior of estimators. It is conventional in the community to make assumptions associated with this spectrum, which we find less natural than our related assumptions A2 and A3. The covariance operator is an analog of the covariance matrix in infinite-dimensional spaces. For our problem setting, one natural covariance operator T_X is defined as

$$\begin{aligned} &T_X : L^2_{\rho_X} \to L^2_{\rho_X}, \\ &g \mapsto \int_{\mathcal{X}} g(\tau)\left(\sum_{j=1}^{\infty} j^{-2s}\psi_j(\tau)\psi_j\right) d\rho_X(\tau). \end{aligned} \tag{37}$$

A direct analysis of the spectrum of T_X is hard. However, there is a simpler operator that we have in hand, which we can relate T_X to

$$\begin{aligned} &T_\nu : L^2_\nu \to L^2_\nu, \\ &g \mapsto \int_{\mathcal{X}} g(\tau)\left(\sum_{j=1}^{\infty} j^{-2s}\psi_j(\tau)\psi_j\right) d\nu(\tau). \end{aligned} \tag{38}$$

We know the eigensystem of T_ν: It is exactly (j^{-2s}, ψ_j) (eigenvalue, eigenfunction). It is direct to check because $\{\psi_j\}$'s are orthonormal w.r.t. ν, so $\int \psi_j(\tau) \sum_{j=1}^{\infty} j^{-2s} \psi_j(\tau)\psi_j \, d\nu(\tau) = j^{-2s}\psi_j$. As an additional contribution, our work shows that with the simple assumptions A2 and A3, we can get knowledge about T_X's eigenvalues from those of T_ν.

LEMMA 6.2. *Given assumptions* A2, A3, *the* j*th eigenvalue*, λ_j, (*sorted in a decreasing order*) *of* T_X *satisfies* $\lambda_j = \Theta(j^{-2s})$.

Moreover, the upper bound of the density in A2 ensures the upper bound in Lemma 6.2 ($\lambda_j = O(j^{-2s})$), and the lower bound of the density ensures the other half of the result. The proof of the above lemma uses the underlying connection between the eigenvalues of an operator and its metric entropy. For rigorous definitions and proof of Lemma 6.2, see Appendix C.

Although the exact result of Lemma 6.2 is not used in the proof of Theorem 6.1 (or Theorem 6.3), we still present it here since it may be of interest itself and the stated results are less technical and easier to comprehend. The proof of the more technical version (Lemma C.14) follows very closely to that of Lemma 6.2. In that more general version, we investigate the spectrum of covariance operators of form: $T^{\omega}_{X,J_n}(f) = \int f(\tau)(\sum_{j=1}^{J_n} j^{-2\omega}\psi_j(x)\psi_j)\, d\rho_X(\tau)$.

To prove Theorem 6.1, we need to engage with a series of RKHSs with kernels given by

$$K^s_{J_n} : \mathcal{X} \times \mathcal{X} \to \mathbb{R},$$

$$(s,t) \mapsto \sum_{j=1}^{J_n} j^{-2s}\psi_j(s)\psi_j(t). \tag{39}$$

While we discuss our work in the context of Sobolev ellipsoids, there is an equivalent formulation directly in RKHS. See Appendix C for more discussion. Although an explicit form for $K^s_{J_n}$ is not in general necessary or accessible for Sieve-SGD, the existence (i.e., the absolute convergence of the infinite sum) of $K^s_{J_n}$ is a direct consequence of A3. This is enough for theoretical analysis. For kernel SGD methods, a fixed kernel (with $J_n = \infty$) is used and there is only one relevant RKHS. This means, on average, kernel SGD is applying the same procedure each iteration; but for Sieve-SGD, we need to engage with a series of increasing RKHSs (on average, Sieve-SGD may not be doing the same thing between iterations). As a side contribution, we present how to handle such a more technically involved case.

Note 2: In contrast to our assumption A3, the hypothesis spaces in [13, 34, 48, 55] are described in terms of "T_X" and its eigendecomposition. This unfortunately obfuscates difficulties related to verifying those conditions when analyzing their statistical performance (though applying the learning algorithm in practice does not need the knowledge of the eigensystem). In particular, because ρ_X is involved in the definition of T_X (37), we need knowledge of (generally unknown) ρ_X to characterize T_X, and understand its eigenvalues and eigenfunctions.

More specifically, in the literature we mentioned above, it is often assumed that for some $r \in [1/2, 1]$ (Definition C.6),

$$\|T_X^{-r}(f_\rho)\|^2_{L^2_{\rho_X}} < \infty. \tag{40}$$

This can be related to a Sobolev ellipsoid-type condition

$$\|T_X^{-r}(f_\rho)\|^2_{L^2_{\rho_X}} = \sum_{j=1}^{\infty} \lambda_j^{-2r}\theta_j^2 < \infty \quad \text{where } f_\rho = \sum_{j=1}^{\infty} \theta_j \phi_j, \tag{41}$$

where $(\lambda_j, \phi_j)_{j=1}^{\infty}$ are the eigenvalue and eigenfunctions of operator T_X, and ϕ_j's are orthonormal w.r.t. $L^2_{\rho_X}$. Unfortunately, we cannot directly engage with $(\lambda_j, \phi_j)_{j=1}^{\infty}$, since calculating them requires knowledge of ρ_X. Thus, assumptions formulated in the language of

T_X^{-r} are difficult to directly understand. In contrast, our assumptions translate to analyzing the spectrum of T_ν, which has no dependence on ρ_X, and its spectrum can been directly calculated (as noted above).

Note 3: For parametric SGD methods, usually a bound on the second moment of the gradient vector is required to guarantee rate-optimal performance (both theoretically and in practice). Formally, for optimization problem (10), it is usually required that $E[\|\nabla \ell(\beta)\|^2] \leq R^2 < \infty$ for all $\beta \in \mathbb{R}^d$ [6, 14].

For nonparametric stochastic approximation, there is a similar regularity requirement for the "gradient." Assumptions A2–A3 are enough to ensure this for Sieve-SGD. In our proof, we show that there exists a number $R < \infty$ such that for all $x \in \mathcal{X}$ and any J_n, we have $\|K^s_{x,J_n}\|_K^2 \leq R^2$. This result is listed in Lemma D.1 where $R^2 = M^2\zeta(2s)$ and $\zeta(k) = \sum_{i=1}^\infty i^{-k}$. In Theorem 6.1, we required γ_0 to be smaller than $(2M^2\zeta(2s))^{-1}$ to ensure our theoretical guarantees.

Note 4: For completeness, here we state the minimax rate of our nonparametric regression problem over a Sobolev ellipsoid:

$$\liminf_{n\to\infty} \inf_{\hat{f}} \sup_{f_\rho \in W(s,Q,\{\psi_j\})} E\big[n^{\frac{2s}{2s+1}} \|\hat{f} - f_\rho\|^2_{L^2_{\rho_X}}\big] \geq C, \tag{42}$$

where the infimum ranges over all possible functions $\hat{f}$ that are sufficiently measurable. For a derivation of this lower bound, see [52], Chapter 15. Many other online methods we mentioned in Section 4 can achieve this lower bound, however, their computational expense is in general unfavorable compared with the proposed method.

Also, the bound (36) should not be understood as a dimension-free result. When the feature $X \in \mathbb{R}^p$ is a multivariate vector, the parameter s should be treated as $s = s^*/p$, where s^* is, for example, the order of derivative that we assume the regression function f_ρ has. Plugging this into the result presented in Theorem 6.1 gives a dimension-dependent bound of order $n^{-2s^*/(2s^*+p)}$ in which both the smoothness assumption and dimension show up in the exponent. Such a bound is minimax optimal when learning in a (large) homogeneous multivariate space [45]. In practice, one can usually achieve better performance by leveraging other low-dimensional structures (See Section 5.4 and Appendix B).

6.3. *Robustness of t_j for properly chosen J_n.*

In Section 6.2, we presented the optimality guarantees of Sieve-SGD in the case when the component-specific learning rate is chosen as the most "natural" value, that is, $t_j = j^{-2s}$. In that case, Sieve-SGD is statistically optimal so long as the number of basis functions does not increase too slowly, that is, $J_n \geq n^{1/(2s+1)} \log^2(n)$. Specifically, when $J_n = \infty$, the Sieve-SGD updating rule reduces to the kernel SGD updating rule (20) with kernel $K^s_\infty(X_n, \cdot) = \sum_{j=1}^\infty j^{-2s}\psi_j(X_n)\psi_j(\cdot)$. So long as we have access to the closed form of $K^s_\infty(X_n, \cdot)$, the corresponding kernel SGD estimator is also statistically optimal under the same conditions. In such a scenario, Sieve-SGD can be seen as a truncated-version of a "correct" kernel SGD method with much better computational properties.

Although truncating the kernel in the spectral domain may be seen as an extension of kernel SGD, it can alternatively be seen as related to projection estimators: In that case, however, two pieces of Theorem 6.1 may seem unnatural: (1) the strict requirement on t_j $(= j^{-2s})$; and (2) the fact that we only lower bound the truncation rate, rather than requiring a precise value for the growth of J_n. In the case of the original Theorem 6.1, the bias-variance tradeoff is actually not balanced via truncation. Instead, it is balanced directly using the t_j. The required lower bound on the truncation rate is just given to ensure that we do not accrue excess bias. Alternatively, to better parallel projection estimators, it might seem more natural to directly use the number of basis functions J_n to control the bias-variance tradeoff (there is nothing akin to t_j in (8)). In this section, we will explore this idea: that if we are more precise in specifying J_n, perhaps we can be more flexible with t_j.

More specifically, we are interested in milder conditions on the sequence (t_j) such that if we properly select the rate at which our "dimension" increases (i.e., the rate at which J_n grows), Sieve-SGD would still attain its favorable statistical and computational properties. Since we will be using J_n as the tuning parameter to balance bias and variance, one might expect kernel SGD, which sets $J_n = \infty$, would not always have optimal statistical performance for all sequences (t_j) satisfying the "milder" conditions. This is confirmed via the following theorem: For Sieve-SGD, one can actually use large componentwise step-sizes that need only satisfy $t_j < j^{-1}$ for any smoothness class $W(s)$, so long as the truncation level is appropriately set; while the largest t_j that can be used for kernel-SGD (without truncation) needs to ensure $t_j < j^{-(s+1/2)}$, which depends on the smoothness of f_ρ.

THEOREM 6.3. *Assume* A1–A4. *Set the component-specific learning rate to be* $t_j = j^{-2\omega}$ *with* $1/2 < \omega \le s$. *Choose the learning rate to be* $\gamma_n = \gamma_0 n^{-1/(2s+1)}$, *with* $\gamma_0 \le M^2\zeta(2\omega)/2$. *Choose the number of basis functions to be* $J_n = n^{1/(2s+1)}\log^2 n \vee 1$.

Then the MSE of Sieve-SGD (27) *converges at the following near optimal rate*:

$$E\|\bar{f}_n - f_\rho\|^2_{L^2_{\rho_X}} = O\big(n^{-\frac{2s}{2s+1}}\log^2 n\big). \tag{43}$$

Note 1: The requirement of $t_j < j^{-1}$ is to guarantee a finite "second moment" of the gradient (as in Note 3 under Theorem 6.1). In this theorem, once this minimal requirement is satisfied, the decay rate of t_j does not influence either the statistical guarantees or the computational expense of the estimators — both of these are determined entirely by the truncation level. As we will discuss very soon in Section 6.4, the choice of $J_n = n^{1/(2s+1)}\log^2 n$ in Theorem 6.3 and Theorem 6.1 would result in algorithms that are both statistically and computationally near optimal up to a polylog term.

Note 2: The most direct form of the projection estimator determines the basis functions' coefficients by solving a (unpenalized) least square problem (8) in which there are no learning rates involved. It is the truncation level J_n that determines the bias-variance trade-off and statistical performance. In Theorem 6.3, we present a stochastic approximation analog to that result. From a reproducing-kernel methodology perspective, Theorem 6.1 investigates the cases when the capacity of the kernel (ω) matches the source smoothness (s); in Theorem 6.3, we show under what conditions the mismatch between these two quantities do not affect the statistical (and computational) properties of Sieve-SGD. It is very common to discuss the generalization ability of a reproducing kernel method in the literature when the kernel capacity does not match the source smoothness. For example, in [13] the authors use a pair of parameters (r, α) to state the hypothesis space and the capacity of the kernels. The smoothness of the hypothesis space is determined by the product of the two parameters $r\alpha$. When $r \neq 1/2$, they are considering using a kernel whose capacity does not match the smoothness of f_ρ. Their proposed method must modify the learning rate properly to recover rate optimality (or it is impossible due to saturation).

Comparing their results with Theorem 6.3, there are ω such that the kernel SGD estimator, using kernel $K^\omega_\infty(X_n, \cdot) = \sum_{j=1}^\infty j^{-2\omega}\psi_j(X_n)\psi_j(\cdot)$, may not be optimal no matter how we modify the learning rate γ_n (described as "saturation" in [13]). Whereas for Sieve-SGD using the truncated "kernels" $K^\omega_{J_n}(X_n, \cdot) = \sum_{j=1}^{J_n} j^{-2\omega}\psi_j(X_n)\psi_j(\cdot)$, the statistical and computation performance can still be jointly near optimal. Theorem 6.3 is formally similar to such a "source-capacity" discussion, but the results are quite different in nature—in particular, it is the truncation level that "saves" us, and allows a much larger mismatch between kernel capacity and source smoothness.

Note 3: The overall proof structures of Theorem 6.1 and Theorem 6.3 are similar; the difference is in the proof of Theorem 6.1 we need Lemma D.4 and related technical results, but for Theorem 6.3 we use Lemma E.1 instead.

Note 4: We also provide some intuition for using a decreasing learning rate γ_n: For rate-optimal *parametric* SGD methods with averaging, the learning rate γ_n can be taken as a constant γ_0. However, the employed constant γ_0 is inversely proportional to the dimension of parameter (assuming each dimension of the feature has a bounded support) [3], which is, in some sense, consistent with our results (though we have seen no other results in the literature that engage with a dimension that increases as the learning process proceeds). We require the learning rate to be a decreasing sequence so that it can cancel out the effect of increasing the estimator dimension: The increasing dimension would have resulted in a noise variance that is increasing if care was not taken.

6.4. *Near optimal space expense.* In this section, we will show that Sieve-SGD is asymptotically (nearly) space-optimal for estimating f_ρ in a Sobolev ellipsoid under the conditions listed in Section 6.1. We will show that, even with computer round-off error, Sieve-SGD only needs $\Theta(n^{1/(2s+1)}\log^3 n)$ *bits* to achieve the minimax rate for MSE (or off by a $\log^2(n)$ term when $\omega \neq s$ as stated in Theorem 6.3), and further, that there is no estimator with $o(n^{1/(2s+1)})$ bits of space expense that can achieve the minimax rate for estimating $f_\rho \in W(s, Q)$. In our analysis, we note that computers cannot store decimals in infinite precision, and formally deal with a modified version of our algorithm that stores coefficients in fixed precision (that grows in n): This necessitates the extra $\log(n)$ term (compared with the number of basis function needed in Theorem 6.1 and 6.3). The modified algorithm with fixed, but growing precision still results in the same MSE when a round-off error is not considered.

We first give a more formal definition of the space expense of an estimator in our analysis. A regression estimator can be seen as a mapping M_n from the data $Z_1^n = \{(X_i, Y_i) | i = 1, 2, \dots, n\}$ to a function $\hat{f}_n \in \mathcal{F}$. For any such M_n that can be engaged by a computer, must be decomposable into an "encoder-decoder" pair (E_n, D_n). Here, E_n represents the "encoder" that compresses the information into computer memory. Formally, we define E_n to be a mapping from Z_1^n to a binary sequence of length b_n. And the corresponding D_n is the "decoder" of the binary sequence that translates the information saved in memory back to a mathematical object $\hat{f}_n$. Generally, the binary sequence length b_n will increase with n: As more information is contained in the data, we need more memory to store an increasingly accurate estimate of our regression function.

Given an estimator that can decomposed into a pair (E_n, D_n), one can see that the decomposition is not unique. There are, in fact, infinitely many pairs that are trivially different from each other for any such estimator. Moreover, E_n, D_n's can be random mappings if we allow random algorithms: For example, random forests include additional randomness due to bootstrapping/variable selection. In order to be more precise regarding memory complexity constraints, we introduce the following formalization.

DEFINITION 6.4 (b_n-sized estimator). *Given a sequence of integers $(b_i)_{i\in\mathbb{N}}$, we say an estimator $M_n : (\mathcal{X} \times \mathbb{R})^n \to \mathcal{F}$ is a b_n-sized estimator if it satisfies the following conditions*:

1. *For every n, there exists an encoder mapping $E_n : (\mathcal{X} \times \mathbb{R})^n \to \{0, 1\}^{b_n}$, and a decoder mapping $D_n : \{0, 1\}^{b_n} \to \mathcal{F}$ such that*

$$M_n = D_n \circ E_n \tag{44}$$

2. *The decoder D_n is a known, fixed mapping. E_n can be either a random or fixed mapping.*

We use the sample mean as a toy example to illustrate the above definition. In practice, the sample mean is usually a 64-sized estimator of the population mean. Here, 64 stands for the number of bits needed to represent a double-precision floating point number. In this case, the size $b_n = 64$ does not increase with sample size n. However, not every real number can be arbitrarily precisely specified by a fixed-length floating-point number, so a careful asymptotic analysis of estimation of the mean suggests that perhaps we should store a sample mean with growing levels of precision, that is, b_n would need to grow with n. A binary sequence of length s can specify 2^s real numbers, so to achieve the $O(n^{-1})$ statistically optimal bound for mean estimation, a $\log(n)$-sized version of sample mean is formally required. In practice, 64-bit precision is generally more than enough for mean estimation. Nevertheless, in this manuscript we aim to give a more formal and precise asymptotic analysis of our Sieve-SGD estimator.

Readers who are more familiar with computational complexity theory may find our definition similar to a (probabilistic) Turing machine. However, in our framework the machine does not use binary sequences on tapes as input and output; nor do we need to identify the basic operations on the "machine." We aimed to remove unnecessary complexity for readers with a more statistical background. Discussion of Turing machines using finite length working tape can be found in [1], Chapter 4.

To construct Sieve-SGD estimators that achieve (near) optimal MSE, we only need to store the coefficients of the $J_n = \Theta(n^{1/(2s+1)} \log^2 n)$ basis functions. However, as in our example with the sample mean, we need to be careful about the precision with which we store those coefficients. We need to determine: (i) how small we require the round-off error to be in order to maintain the statistical optimality of Sieve-SGD, and (ii) how much space expense is required to achieve such precision. In Appendix F.1, we identify how round-off error is introduced into the system and how it decreases as more bits are used to store each coefficient. In Corollary F.2, we show that by using $\Theta(\log n)$ bits per coefficient, a $O(n^{1/(2s+1)} \log^3 n)$-sized version of Sieve-SGD can achieve the same optimal convergence rate as in the infinite precision setting (or equivalently round-off error-free setting).

Combining the above result with the following theorem, we can conclude that no MSE rate-optimal estimator can require less memory by a polynomial factor than Sieve-SGD.

THEOREM 6.5. *Let b_n be a sequence of integers, and $b_n = o(n^{1/(2s+1)})$. Let $\mathcal{M}(b_n)$ be the collection of all b_n-sized estimators, then we have*

$$\lim_{n\to\infty} \inf_{M_n \in \mathcal{M}(b_n)} \sup_{f_\rho \in W(s,Q,\{\psi_j\})} E\big[n^{\frac{2s}{2s+1}} \|M_n(Z_1^n) - f_\rho\|^2_{L^2_{\rho_X}}\big] = \infty, \tag{45}$$

that is, no such b_n-sized estimators can be rate-optimal.

This theorem tells us we cannot find any minimax rate-optimal $o(n^{1/(2s+1)})$-sized estimator. Thus, the best rate-optimal estimator one can expect to find is a $\Theta(n^{1/(2s+1)})$-sized estimator: Sieve-SGD's space expense only misses this lower bound by a polylogarithmic factor.

We give the proof of the above theorem in Appendix F.2. Although here we focus on regression in Sobolev spaces, the technique used can be applied to other hypothesis spaces. The proof is based on the concept that metric-entropy is the minimal number of bits needed to represent an arbitrary function from a function space up to ϵ-error, which can be traced back to [28]. Also, following a very similar argument, one can prove that no constant-sized estimator can be rate-optimal (or even consistent) for parametric regression problems. We discuss this further in the Appendix F.2. We also include some discussion of the time expense in Section 8.

TABLE 1

Settings of simulation studies. $B_4(x) = x^4 - 2x^3 + x^2 - \frac{1}{30}$ is the 4*th Bernoulli polynomial.* $\{x\}$ *indicates the fractional part of x*

	Example 1	Example 2
True f_ρ	$B_4(x)$	$4\sqrt{2}\sum_{j=1}^{50}(-1)^{j+1}j^{-4}\sin((2j-1)\pi x/2)$
Ellipsoid para. s	2	3
J_n	$n^{0.21}$	$n^{0.10}$ and $n^{0.15}$ and $n^{0.43}$
t_j	$j^{-1.02}$ & j^{-4}	j^{-6}
$\psi_j(x)$	$\sin(2\pi\lceil j/2\rceil x)$, j is even $\cos(2\pi\lceil j/2\rceil x)$, j is odd	$\sqrt{2}\sin(\frac{(2j-1)\pi x}{2})$
Kernel $K(s,t)$	$-\frac{1}{24}B_4(\{s-t\})$	$\min(s,t)$
Noise	Unif$[-0.02, 0.02]$ or Unif$[-0.2, 0.2]$	Normal(0, 1)
γ_0	3	1

7. Simulation study.

7.1. *Sieve-SGD for online regression.* In this section, we illustrate both the statistical and computational properties of Sieve-SGD with simulated examples. The two examples we use have different f_ρ, $W(s, Q, \{\psi_j\})$ and ρ_X. The user-specified measure ν is taken as the uniform distribution over [0, 1] in both. We provide the details of our simulation settings in Table 1. These two examples are designed for verifying our theoretical guarantees: The f_ρ we use have known explicit series expansion or is constructed explicitly using the basis function ψ_j (to ensure the truth is hard enough to learn in the assumed Sobolev ellipsoid). In Appendix B, we provide more numerical examples to better mimic the practical application: We engage with multivariate features and compare Sieve-SGD with many popular machine learning methods.

EXAMPLE 1. In this example, we examine the empirical performance of Sieve-SGD and compare it with two other methods in batch or online nonparametric regression: kernel ridge regression (KRR) [52] and kernel SGD [13]. We will see that the relationship between generalization error $E\|\bar{f}_n - f_\rho\|_2^2$ and sample size corresponds well with our theoretical expectations presented in Theorem 6.1 (Figure 1).

The true regression function we chose for Example 1 is also used in the analysis of kernel SGD [13]. In that paper, kernel SGD with Polyak averaging is compared with other (kernel-based) nonparametric online estimators [48, 55], and has been shown to have similar or better performance, so we include only kernel SGD with averaging as the reference online estimator. We also note that although KRR performs slightly better than online methods, its time expense (which is of order $\Theta(n^3)$ per update) is dramatically more than online estimators (kernel SGD $\Theta(n)$, Sieve-SGD $\Theta(J_n)$, per update).

We compare the empirical performance of Sieve-SGD under two different distributions of X. In Figure 1, panel (A), X has an uniform distribution over [0, 1], and in panel (B) it has a distribution with a strictly smaller support (uniform over [0.25, 0.75]). The trigonometric basis functions we use are orthonormal w.r.t. ν, the Lebesgue measure over [0, 1] (panel (A)) but not w.r.t. the one in panel (B). Although the only the feature distribution in panel (A) satisfies the distribution assumption in A2, in both cases Sieve-SGD achieves the optimal rate. This is a heuristic evidence indicating the lower bound requirement in A2 may be due to some artifacts in the proof.

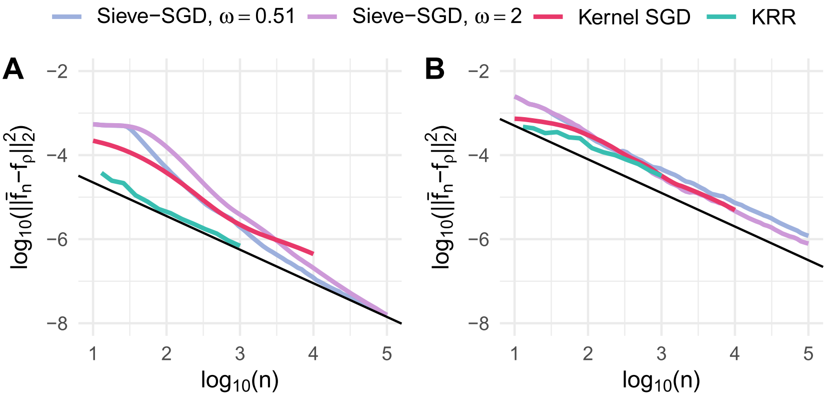

FIG. 1. *Example* 1, $\log_{10}\|\bar{f}_n - f_\rho\|_2^2$ *against* $\log_{10} n$. *The black line has slope* $= -4/5$, *which represents the optimal rate. Each curve is calculated as the average of* 100 *repetitions.* (*A*) *X is uniformly distributed over* [0, 1]. *In this setting, SNR* ~ 3. (*B*) *X has a distribution in which* ψ_j *are not orthonormal. We present the results with very large noise*, SNR ~ 0.03. *Due to different computational costs, we chose different maximum n for different methods.*

EXAMPLE 2. In this example, we consider the performance of Sieve-SGD under different $J_n = \lfloor n^\alpha \rfloor$ (number of basis functions). The f_ρ we use is explicitly constructed with basis functions $\psi_j(x) = \sqrt{2}\sin((2j-1)\pi x/2)$ and we tune the proposed method based on the (correct) assumption that it belongs to Sobolev ellipsoid $W(3, Q, \{\psi_j\})$ (see Theorem 4.1 of [26], Chapter 1, for completeness and orthonormality of $\{\psi_j\}$).

According to Theorem 6.1, in order to guarantee statistical optimality, we need $\alpha \geq (2s+1)^{-1} \sim 0.14$. We consider several values of α, one below this threshold, and two above it:

$$\mathbf{0.10} < \frac{1}{2s+1} \sim 0.14 < \mathbf{0.15} < \mathbf{0.43}. \tag{46}$$

As we can see from Figure 2(A), when $\alpha = 0.15$ and 0.43, Sieve-SGD is rate-optimal as expected. When $\alpha = 0.10$, we are using too few basis functions, which results in the suboptimal statistical performance. Such a suboptimality is because of the bias term: there are too few basis functions used. In fact, the parameter setting $\alpha = 0.1$ is so small that there are only 3 basis functions used when $n = 10^5$. To verify the above statement, we can briefly calculate

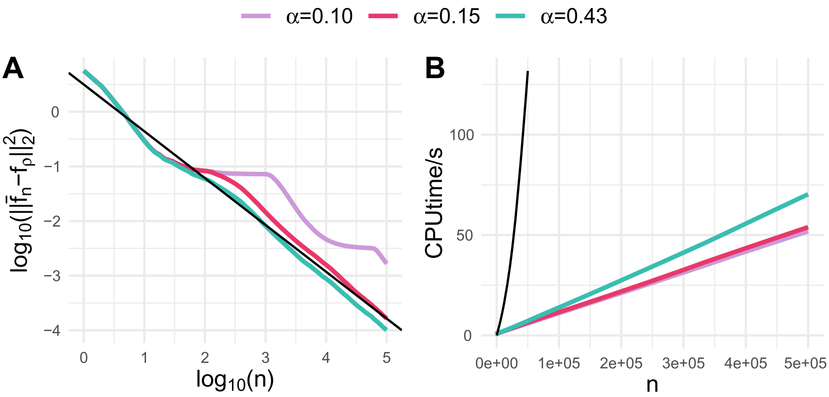

FIG. 2. *Example* 2, *effect of truncation exponents* $\alpha = 0.10, 0.15, 0.43$. (*A*) *Statistical performance*, $\log_{10}\|\bar{f}_n - f_\rho\|_2^2$ *against* $\log_{10} n$. *The black line has slope* $= -6/7$, *which represents the optimal rate. Each curve is calculated as the average of* 100 *repetitions.* (*B*) *The accumulated CPU time to process n observations. The black line is the CPU time of kernel SGD, included for a benchmark.*

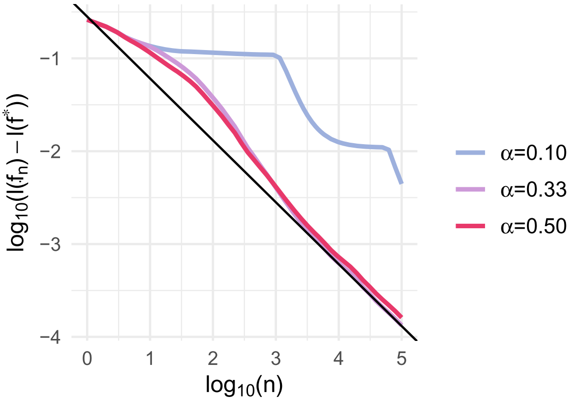

FIG. 3. *Example 3, empirical performance of Sieve-SGD in a nonparametric logistic regression problem. Plot* $\log_{10}(l(\bar{f}_n) - l(f^*))$ *against* $\log_{10} n$. *The black line has slope* $= -2/3$. *Each curve is calculated as the average of* 100 *repetitions*.

when the second and the third basis functions are added in: $(10^3)^{0.1} \sim 2$, this corresponds to the first acceleration of the learning rate around $\log_{10}(n) = 3$; similarly, $(10^{4.8})^{0.1} \sim 3$, which explains the second one.

In Figure 2(B), we show the CPU time for reference. For Sieve-SGD, the *accumulated* CPU time should be on the order of $\Theta(n^{1+\alpha})$: The larger α, the more basis functions required, the slower the algorithm. We also include the CPU time of kernel SGD with averaging as a benchmark, which has a cumulative computational expense of order $\Theta(n^2)$. The code is written in R (4.0.4), and runs on (the CPU of) a machine with 1 Intel Core m3 processor, 1.2 GHz, with 8 GB of RAM.

7.2. *Sieve-SGD for alternative convex losses.*

In this section, we provide the results of an experiment applying Sieve-SGD to online nonparametric logistic regression. Although this manuscript gives no theoretical guarantees in this setting, it is still of interest to see the empirical performance of Sieve-SGD for general convex loss.

EXAMPLE 3. In this setting, the distribution of class labels Y was generated by $Y \sim 2\,\mathrm{Ber}(g(X)) - 1$, where $(g(x))^{-1} = 1 + \exp(-5(1 - 2|x - 0.5|))$; and the distribution of X was uniform over $[0, 1]$. Thus, the minimizer f^* of loss $E[\ell(Y, f(X))] = E[\log(1 + \exp\{-Yf(X)\})]$ is $f^* = 5(1 - 2|x - 0.5|)$.

When we apply the Sieve-SGD estimator (30) to this problem, we assume

$$f^* \in W\bigl(1, Q, \{\sqrt{2}\sin\bigl((2j-1)\pi x/2\bigr)\}\bigr). \tag{47}$$

We try several $\alpha = 0.10, 0.33, 0.50$, all with $\gamma_0 = 6$. As we can see from Figure 3, the regret $E[\ell(\bar{f}_n) - \ell(f^*)]$ converges to zero at an apparent rate of $n^{-2/3}$ when $\alpha = 0.33, 0.50$ (which would agree with our result for squared error loss). When the number of basis functions increases too slowly (here is $\alpha = 0.10$), the regret decreases slowly after ~ 10 observations (for similar reason of overflowing bias term as we noted in Section 7.1).

8. Discussion.

In this paper, we considered online nonparametric regression in a Sobolev ellipsoid. We proposed the *Sieve Stochastic Gradient Descent estimator (Sieve-SGD)*, an online estimator inspired by both (a) the nonparametric projection estimator, which is a special realization of general sieve estimators; and (b) estimators constructed using stochastic gradient descent algorithms. By using an increasing number of basis functions, Sieve-SGD has a rate-optimal estimation error and is computationally very efficient.

For online learning problems with general convex losses, the optimal estimation rate depends on both the hypothesis space and loss function (e.g., whether it is Lipschitz or strongly convex). In this paper, we did not establish theoretical guarantees for Sieve-SGD when applied to a general convex loss, however, we gave some empirical evidence that it can perform well there. We believe our proof techniques might be extended beyond squared-error loss, perhaps using ideas in [3, 10, 34, 35].

We have seen a rich collection of work in the past decade targeting the optimality of estimators under computational (especially time expense) constraints. A lot of those results are established in the context of sparse PCA and related sparse-low-rank matrix problems, for example, [8, 19, 20, 33, 53, 61]. The main focus of these work is usually comparing the statistical performance of the best polynomial-time algorithm with that of the "optimal" algorithm without any computational restrictions. By relating their statistical problem with a known NP problem [1], they can usually show the suboptimality of polynomial-time algorithms under the famous conjecture $P \neq NP$. However, for the nonparametric regression problem in this paper, there is a polynomial-time estimator that can achieve the global minimax rate. It is of theoretical interest to know if there are any statistically rate-optimal online estimators that require less than $\Theta(n^{1+1/(2s+1)})$ time expense: We hypothesize that there are not.

Acknowledgments. The authors would like to thank the anonymous referees, an associate editor and the editor for their constructive comments that improved an early version of this paper.

Funding. N. Simon and T. Zhang were both supported by NIH grant R01HL137808.

SUPPLEMENTARY MATERIAL

A sieve stochastic gradient descent estimator for online nonparametric regression in Sobolev ellipsoids: Supplementary material (DOI: 10.1214/22-AOS2212SUPP; .pdf). We provide more discussion on the application of our proposed methods. The complete proof of the theoretical results is also provided.

REFERENCES

[1] Arora, S. and Barak, B. (2009). *Computational Complexity*: *A Modern Approach*. Cambridge Univ. Press, Cambridge. MR2500087 https://doi.org/10.1017/CBO9780511804090

[2] Babichev, D. and Bach, F. (2018). Constant step size stochastic gradient descent for probabilistic modeling. *Stat* **1050** 21.

[3] Bach, F. and Moulines, E. (2013). Non-strongly-convex smooth stochastic approximation with convergence rate $o(1/n)$. In *Advances in Neural Information Processing Systems* 773–781.

[4] Belkin, M., Hsu, D., Ma, S. and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias-variance trade-off. *Proc. Natl. Acad. Sci. USA* **116** 15849–15854. MR3997901 https://doi.org/10.1073/pnas.1903070116

[5] Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Springer, Berlin.

[6] Borkar, V. S. (2009). *Stochastic Approximation*: *A Dynamical Systems Viewpoint* **48**. Springer, Berlin.

[7] Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'*2010 177–186. Physica-Verlag/Springer, Heidelberg. MR3362066

[8] Cai, T. T., Liang, T. and Rakhlin, A. (2017). Computational and statistical boundaries for submatrix localization in a large noisy matrix. *Ann. Statist.* **45** 1403–1430. MR3670183 https://doi.org/10.1214/16-AOS1488

[9] Calandriello, D., Lazaric, A. and Valko, M. (2017). Efficient second-order online kernel learning with adaptive embedding. In *Advances in Neural Information Processing Systems* 6140–6150.

[10] Caponnetto, A. and De Vito, E. (2007). Optimal rates for the regularized least-squares algorithm. *Found. Comput. Math.* **7** 331–368. MR2335249 https://doi.org/10.1007/s10208-006-0196-8

[11] CHRISTMANN, A. and STEINWART, I. (2008). Support vector machines.

[12] CUCKER, F. and SMALE, S. (2002). On the mathematical foundations of learning. *Bull. Amer. Math. Soc.* (*N.S.*) **39** 1–49. MR1864085 https://doi.org/10.1090/S0273-0979-01-00923-5

[13] DIEULEVEUT, A. and BACH, F. (2016). Nonparametric stochastic approximation with large step-sizes. *Ann. Statist.* **44** 1363–1399. MR3519927 https://doi.org/10.1214/15-AOS1391

[14] DUCHI, J. C. (2014). Multiple Optimality Guarantees in Statistical Learning Ph.D. thesis UC Berkeley.

[15] EUBANK, R. L. and SPECKMAN, P. (1990). Curve fitting by polynomial-trigonometric regression. *Biometrika* **77** 1–9. MR1049403 https://doi.org/10.1093/biomet/77.1.1

[16] FASSHAUER, G. E. and MCCOURT, M. J. (2015). *Kernel-Based Approximation Methods Using Matlab* **19**. World Scientific, Singapore.

[17] FROSTIG, R., GE, R., KAKADE, S. M. and SIDFORD, A. (2015). Competing with the empirical risk minimizer in a single pass. In *Conference on Learning Theory* 728–763.

[18] GAILLARD, P. and GERCHINOVITZ, S. (2015). A chaining algorithm for online nonparametric regression. In *Conference on Learning Theory* 764–796.

[19] GAO, C., MA, Z., REN, Z. and ZHOU, H. H. (2015). Minimax estimation in sparse canonical correlation analysis. *Ann. Statist.* **43** 2168–2197. MR3396982 https://doi.org/10.1214/15-AOS1332

[20] GAO, C., MA, Z. and ZHOU, H. H. (2017). Sparse CCA: Adaptive estimation and computational barriers. *Ann. Statist.* **45** 2074–2101. MR3718162 https://doi.org/10.1214/16-AOS1519

[21] GEER, S. A. and VAN DE GEER, S. (2000). *Empirical Processes in M-Estimation* **6**. Cambridge Univ. Press, Cambridge.

[22] GYÖRFI, L., KOHLER, M., KRZYZAK, A. and WALK, H. (2006). *A Distribution-Free Theory of Nonparametric Regression*. Springer, Berlin.

[23] HALL, P. and OPSOMER, J. D. (2005). Theory for penalised spline regression. *Biometrika* **92** 105–118. MR2158613 https://doi.org/10.1093/biomet/92.1.105

[24] HÄRDLE, W., KERKYACHARIAN, G., PICARD, D. and TSYBAKOV, A. (2012). *Wavelets*, *Approximation*, *and Statistical Applications* **129**. Springer, Berlin.

[25] HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2009). *The Elements of Statistical Learning*: *Data Mining*, *Inference*, *and Prediction*, 2nd ed. *Springer Series in Statistics*. Springer, New York. MR2722294 https://doi.org/10.1007/978-0-387-84858-7

[26] HERNÁNDEZ, E. and WEISS, G. (1996). *A First Course on Wavelets*. *Studies in Advanced Mathematics*. CRC Press, Boca Raton, FL. MR1408902 https://doi.org/10.1201/9781420049985

[27] KENNEDY, R. A., SADEGHI, P., KHALID, Z. and MCEWEN, J. D. (2013). Classification and construction of closed-form kernels for signal representation on the 2-sphere. In *Wavelets and Sparsity XV* **8858** 88580M. International Society for Optics and Photonics.

[28] KOLMOGOROV, A. N. and TIHOMIROV, V. M. (1959). ε-entropy and ε-capacity of sets in function spaces. *Uspekhi Mat. Nauk* **14** 3–86. MR0112032

[29] KOPPEL, A., WARNELL, G., STUMP, E. and RIBEIRO, A. (2019). Parsimonious online learning with kernels via sparse projections in function space. *J. Mach. Learn. Res.* **20** 83–126.

[30] KUSHNER, H. J. and YIN, G. G. (2003). *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. *Applications of Mathematics* (*New York*) **35**. Springer, New York. MR1993642

[31] LIANG, T. and RAKHLIN, A. (2020). Just interpolate: Kernel "ridgeless" regression can generalize. *Ann. Statist.* **48** 1329–1347. MR4124325 https://doi.org/10.1214/19-AOS1849

[32] LU, J., HOI, S. C., WANG, J., ZHAO, P. and LIU, Z.-Y. (2016). Large scale online kernel learning. *J. Mach. Learn. Res.* **17** 1613–1655.

[33] MA, Z. and WU, Y. (2015). Computational barriers in minimax submatrix detection. *Ann. Statist.* **43** 1089–1116. MR3346698 https://doi.org/10.1214/14-AOS1300

[34] MARTEAU-FEREY, U., BACH, F. and RUDI, A. (2019). Globally convergent Newton methods for Ill-conditioned generalized self-concordant losses. In *Advances in Neural Information Processing Systems* 7634–7644.

[35] MARTEAU-FEREY, U., OSTROVSKII, D., BACH, F. and RUDI, A. (2019). Beyond least-squares: Fast rates for regularized empirical risk minimization through self-concordance. In *Conference on Learning Theory* 2294–2340. PMLR.

[36] MICHEL, V. (2012). *Lectures on Constructive Approximation*: *Fourier*, *Spline*, *and Wavelet Methods on the Real Line*, *the Sphere*, *and the Ball*. Springer, Berlin.

[37] NEMIROVSKI, A. (2000). Topics in non-parametric statistics. In *Lectures on Probability Theory and Statistics* (*Saint-Flour*, 1998). *Lecture Notes in Math.* **1738** 85–277. Springer, Berlin. MR1775640

[38] NEMIROVSKY, A. S. and YUDIN, D. B. (1983). *Problem Complexity and Method Efficiency in Optimization*. *Wiley-Interscience Series in Discrete Mathematics*. Wiley, New York. MR0702836

[39] NOVAK, E. and WOZNIAKOWSKI, H. (2008). *Tractability of Multivariate Problems*. *Vol.* 1: *Linear Information*.

[40] RAKHLIN, A. and SRIDHARAN, K. (2015). Online nonparametric regression with general loss functions. arXiv preprint. Available at arXiv:1501.06598.

[41] RASKUTTI, G., YU, B. and WAINWRIGHT, M. J. (2009). Lower bounds on minimax rates for nonparametric regression with additive sparsity and smoothness. *Adv. Neural Inf. Process. Syst.* **22** 1563–1570.

[42] SHEN, X. (1997). On methods of sieves and penalization. *Ann. Statist.* **25** 2555–2591. MR1604416 https://doi.org/10.1214/aos/1030741085

[43] SI, S., KUMAR, S. and LI, Y. (2018). Nonlinear online learning with adaptive Nyström approximation. arXiv preprint. Available at arXiv:1802.07887.

[44] STEINWART, I. and SCOVEL, C. (2012). Mercer's theorem on general domains: On the interaction between measures, kernels, and RKHSs. *Constr. Approx.* **35** 363–417. MR2914365 https://doi.org/10.1007/s00365-012-9153-3

[45] STONE, C. J. (1980). Optimal rates of convergence for nonparametric estimators. *Ann. Statist.* **8** 1348–1360. MR0594650

[46] STONE, C. J. (1985). Additive regression and other nonparametric models. *Ann. Statist.* **13** 689–705. MR0790566 https://doi.org/10.1214/aos/1176349548

[47] SUN, H. (2005). Mercer theorem for RKHS on noncompact sets. *J. Complexity* **21** 337–349. MR2138444 https://doi.org/10.1016/j.jco.2004.09.002

[48] TARRÈS, P. and YAO, Y. (2014). Online learning as stochastic approximation of regularization paths: Optimality and almost-sure convergence. *IEEE Trans. Inf. Theory* **60** 5716–5735. MR3252416 https://doi.org/10.1109/TIT.2014.2332531

[49] TSYBAKOV, A. (2008). *Introduction to Nonparametric Estimation*. Springer, Berlin.

[50] VEMPALA, S. S. (2005). *The Random Projection Method* **65**. American Mathematical Soc., Providence.

[51] WAHBA, G. (1990). *Spline Models for Observational Data*. *CBMS-NSF Regional Conference Series in Applied Mathematics* **59**. SIAM, Philadelphia, PA. MR1045442 https://doi.org/10.1137/1.9781611970128

[52] WAINWRIGHT, M. J. (2019). *High-Dimensional Statistics*: *A Non-asymptotic Viewpoint*. *Cambridge Series in Statistical and Probabilistic Mathematics* **48**. Cambridge Univ. Press, Cambridge. MR3967104 https://doi.org/10.1017/9781108627771

[53] WANG, T., BERTHET, Q. and SAMWORTH, R. J. (2016). Statistical and computational trade-offs in estimation of sparse principal components. *Ann. Statist.* **44** 1896–1930. MR3546438 https://doi.org/10.1214/15-AOS1369

[54] WOOD, S. N. (2017). *Generalized Additive Models*: *An Introduction with R*. *Texts in Statistical Science Series*. CRC Press, Boca Raton, FL. MR3726911

[55] YING, Y. and PONTIL, M. (2008). Online gradient descent learning algorithms. *Found. Comput. Math.* **8** 561–596. MR2443089 https://doi.org/10.1007/s10208-006-0237-y

[56] YUAN, M. and CAI, T. T. (2010). A reproducing kernel Hilbert space approach to functional linear regression. *Ann. Statist.* **38** 3412–3444. MR2766857 https://doi.org/10.1214/09-AOS772

[57] YUAN, M. and ZHOU, D.-X. (2016). Minimax optimal rates of estimation in high dimensional additive models. *Ann. Statist.* **44** 2564–2593. MR3576554 https://doi.org/10.1214/15-AOS1422

[58] ZHANG, T. and SIMON, N. (2021). An online projection estimator for nonparametric regression in reproducing kernel Hilbert spaces. arXiv preprint. Available at arXiv:2104.00780.

[59] ZHANG, T. and SIMON, N. (2022). Regression in tensor product spaces by the method of sieves. arXiv preprint. Available at arXiv:2206.02994.

[60] ZHANG, T. and SIMON, N. (2022). Supplement to "A Sieve Stochastic Gradient Descent estimator for online nonparametric regression in Sobolev ellipsoids." https://doi.org/10.1214/22-AOS2212SUPP

[61] ZHANG, Y., WAINWRIGHT, M. J. and JORDAN, M. I. (2014). Lower bounds on the performance of polynomial-time algorithms for sparse linear regression. In *Conference on Learning Theory* 921–948.