The Annals of Statistics
2020, Vol. 48, No. 1, 251–273
https://doi.org/10.1214/18-AOS1801

STATISTICAL INFERENCE FOR MODEL PARAMETERS IN STOCHASTIC GRADIENT DESCENT

BY XI CHEN[1,*], JASON D. LEE[2], XIN T. TONG[3] AND YICHEN ZHANG[1,**]

[1]*Department of Technology, Operations, and Statistics, Stern School of Business, New York University,* [*]*xchen3@stern.nyu.edu;* [**]*yzhang@stern.nyu.edu*

[2]*Data Sciences and Operations, Marshall School of Business, University of Southern California, jasonlee@marshall.usc.edu*

[3]*Department of Mathematics, National University of Singapore, mattxin@nus.edu.sg*

The stochastic gradient descent (SGD) algorithm has been widely used in statistical estimation for large-scale data due to its computational and memory efficiency. While most existing works focus on the convergence of the objective function or the error of the obtained solution, we investigate the problem of statistical inference of true model parameters based on SGD when the population loss function is strongly convex and satisfies certain smoothness conditions.

Our main contributions are twofold. First, in the fixed dimension setup, we propose two consistent estimators of the asymptotic covariance of the average iterate from SGD: (1) a plug-in estimator, and (2) a batch-means estimator, which is computationally more efficient and only uses the iterates from SGD. Both proposed estimators allow us to construct asymptotically exact confidence intervals and hypothesis tests.

Second, for high-dimensional linear regression, using a variant of the SGD algorithm, we construct a debiased estimator of each regression coefficient that is asymptotically normal. This gives a one-pass algorithm for computing both the sparse regression coefficients and confidence intervals, which is computationally attractive and applicable to online data.

1. Introduction. Estimation of model parameters by minimizing an objective function is a fundamental idea in statistics. Let $x^* \in \mathbb{R}^d$ be the true d-dimensional model parameters. In common models, x^* is the minimizer of a convex objective function $F(x) : \mathbb{R}^d \to \mathbb{R}$, that is,

$$x^* = \operatorname{argmin}\left(F(x) := \mathbb{E}_{\zeta\sim\Pi} f(x,\zeta) = \int f(x,\zeta)\,\mathrm{d}\Pi(\zeta)\right), \tag{1}$$

where ζ denotes the random sample from a probability distribution Π and $f(x,\zeta)$ is the loss function.

A widely used optimization method for minimizing $F(x)$ is the *stochastic gradient descent (SGD)*, which has a long history in optimization (see, e.g., Nemirovski et al. (2008), Polyak and Juditsky (1992), Robbins and Monro (1951)). In particular, let x_0 denote any given starting point. SGD is an iterative algorithm, where the ith iterate x_i takes the following form:

$$x_i = x_{i-1} - \eta_i \nabla f(x_{i-1}, \zeta_i). \tag{2}$$

The step size η_i is a decreasing sequence in i, ζ_i is the ith sample randomly drawn from the distribution Π, and $\nabla f(x_{i-1},\zeta_i)$ denotes the gradient of $f(x,\zeta_i)$ with respect to x at $x = x_{i-1}$. The algorithm outputs either the last iterate x_n, or the average iterate $\bar{x}_n = \frac{1}{n}\sum_{i=1}^n x_i$

Received October 2017; revised July 2018.
MSC2010 subject classifications. Primary 62J10, 62M02; secondary 60K35.
Key words and phrases. Stochastic gradient descent, asymptotic variance, batch-means estimator, high-dimensional inference, time-inhomogeneous Markov chain.

as the solution to the optimization problem in (1). When $\bar{x}_n$ is adopted as the solution, the algorithm is referred to as the averaged SGD (ASGD), and such an averaging step is known as the Polyak–Ruppert averaging (Polyak and Juditsky (1992), Ruppert (1988)). SGD has many computational and storage advantages over traditional deterministic optimization methods. First, SGD only uses *one pass* over the data and the per-iteration time complexity of SGD is $O(d)$, which is independent of the sample size. Second, there is no need for SGD to store the dataset, and thus SGD naturally fits in the *online setting*, where each sample arrives sequentially (e.g., search queries or transactional data). Moreover, ASGD is known to achieve the optimal convergence rate in terms of $\mathbb{E}(F(\bar{x}_n) - F(x^*))$ with the rate of $O(1/n)$ (Rakhlin, Shamir and Sridharan (2012)) when $F(x)$ is smooth and strongly convex. It has become the prevailing optimization method for many machine learning tasks (Srebro and Tewari (2010)), for example, training deep neural networks.

Based on the simple SGD template in (2), there are a large number of variants developed in the optimization and statistical learning literature. Most existing works only focus on the convergence in terms of the objective function or the distance between the obtained solution and the true minimizer x^* of (1). However, the statistical inference (e.g., constructing confidence intervals) for each coordinate of x^* based on SGD has largely remained unexplored. Inference is a core topic in statistics and a confidence interval has been widely used to quantify the uncertainty in the estimation of model parameters. In this paper, we propose computationally efficient methods to conduct the inference for each coordinate of x_j^* for $j = 1, 2, \ldots, d$ based on SGD. With the developed techniques, one can test if $x_j^* = c$ for any number c, and tell a range of values that x_j^* lies within it with a certain probability. These objectives cannot be achieved by deriving deviation inequalities or generalization error bounds (see Section 1.1 for details).

The proposed methods are built on a classical result of ASGD, which characterizes the limiting distribution of $\bar{x}_n$. In particular, let $A = \nabla^2 F(x^*)$ be the Hessian matrix of $F(x)$ at $x = x^*$ and S be the covariance matrix of $\nabla f(x^*, \zeta)$, that is,

$$S = \mathbb{E}\big([\nabla f(x^*, \zeta)][\nabla f(x^*, \zeta)]^T\big). \tag{3}$$

Note that $\mathbb{E}\nabla f(x^*, \zeta) = \nabla F(x^*) = 0$, provided the interchangeability of derivative and expectation. Ruppert (1988) and Polyak and Juditsky (1992) showed that when d is fixed and F is strongly convex with a Lipschitz gradient, by choosing appropriately diminishing step sizes, $\sqrt{n}(\bar{x}_n - x^*)$ converges in distribution to a multivariate normal random vector, that is,

$$\sqrt{n}\big(\bar{x}_n - x^*\big) \Rightarrow \mathcal{N}\big(0, A^{-1} S A^{-1}\big). \tag{4}$$

However, this asymptotic normality result itself cannot be used to provide confidence intervals. To construct an asymptotically valid confidence interval (or equivalently, an asymptotically valid test that controls the type I error), we need to further construct a consistent estimator of the asymptotic covariance of $\sqrt{n}\bar{x}_n$, that is, $A^{-1} S A^{-1}$. The standard covariance estimator simply estimates A and S by their sample versions, and replaces the x^* in A and S by $\bar{x}_n$. However, this standard estimator cannot be constructed in an online fashion. In other words, all the data is required to be stored to compute this estimator since $\bar{x}_n$ can only be known when the SGD procedure terminates. This requirement loses the advantage of SGD in terms of data storage.

To address this challenge, we propose two approaches to estimate $A^{-1} S A^{-1}$ without the need of storing the data. The first approach is the *plug-in* estimator. In particular, we propose a thresholding estimator $\widetilde{A}_n$ of A based on the sample estimate $A_n = \frac{1}{n}\sum_{i=1}^{n} \nabla^2 f(x_{i-1}, \zeta_i)$. Note that this is not the standard sample estimate since each term $\nabla^2 f(x_{i-1}, \zeta_i)$ is regarding different SGD iterates x_{i-1} (in contrast to a single $\bar{x}_n$), and thus can be constructed online. This construction facilitates the online computation of A_n, which does not need to store each

x_i and ζ_i. Together with the sample estimate S_n of S, the asymptotic covariance $A^{-1}SA^{-1}$ is estimated by $\widetilde{A}_n^{-1}S_n\widetilde{A}_n^{-1}$, which is proven to be a consistent estimator (see Theorem 4.2).

However, the plug-in estimator requires the computation of the Hessian matrix of the loss function f and its inverse, which is usually not available for legacy codes where only the SGD iterates are available. Now a natural question arises: can we estimate the asymptotic covariance *only using the iterates from SGD without requiring additional information*? We provide an affirmative answer to this question by proposing a computationally efficient *batch-means* estimator. Basically, we split the sequence of SGD iterates $\{x_1, x_2, \ldots, x_n\}$ into $M+1$ batches with batch size $n_0, n_1, \ldots, n_M$. The 0th batch is discarded since the iterates in that are far from the optimum. The batch-means estimator is a "weighted" sample covariance matrix that treats each batch-means as a sample.

The idea of batch-means estimator can be traced to Markov Chain Monte Carlo (MCMC), where the batch-means method with equal batch size (see, e.g., Damerdji (1991), Fishman (1996), Flegal and Jones (2010), Geyer (1992), Glynn and Iglehart (1990), Glynn and Whitt (1991), Jones et al. (2006)) is widely used for variance estimation in a time-homogeneous Markov chain. The SGD iterates in (2) indeed form a Markov chain, as x_i only depends on x_{i-1}. However, since the step size sequence η_i is a diminishing sequence, it is a *time-inhomogenous* Markov chain. Moreover, the asymptotic behavior of SGD and MCMC are fundamentally different: while the former converges to the optimum, the latter travels ergodically inside the state space. As a consequence of these important differences, previous literature on batch-means methods is not applicable to our analysis. To address this challenge, our new batch-means method constructs batches of *increasing sizes*. The sizes of batches are chosen to ensure that the correlation decays appropriately among far-apart batches, so that far-apart batch-means can be roughly treated as independent. In Theorem 4.3, we prove that the proposed batch-means method is a consistent estimator of the asymptotic covariance. Further, we believe this new batch-means algorithm with increasing batch sizes is of independent interest since it can be used to estimate the covariance structure of other time-inhomogeneous Markov chains.

As both the plug-in and the batch-means estimator provide asymptotically exact confidence intervals, each of them has its own advantages:

1. The plug-in estimator has a faster convergence rate than the batch-means estimator (see Theorem 4.2 and Corollary 4.5).
2. The plug-in estimator requires the computation of the Hessian matrix of the loss function and its inverse, which can be expensive to obtain for many applications. The batch-means estimator does not require computing any of them. To establish the consistency result, the plug-in estimator requires an additional Lipschitz condition over the Hessian matrix of the loss function (see Assumption 4.1).
3. The plug-in estimator directly computes the entire estimator $\widetilde{A}_n^{-1}S_n\widetilde{A}_n^{-1}$ for the purpose of estimating diagonal elements of $A^{-1}SA^{-1}$. Furthermore, when d is large, storing $\widetilde{A}_n$ and S_n requires $O(d^2)$ bits, which is wasteful since only estimates of the diagonal elements of $A^{-1}SA^{-1}$ are useful for the inference of each x_j^* for $j = 1, 2, \ldots, d$. Meanwhile, the batch-means estimator is able to merely compute and store diagonals.

Practitioners may decide to choose between the plug-in and batch-means estimators based on their tasks and computing resources. The plug-in estimator has a faster convergence rate, which leads to better performance in practice. However, in some cases when the computation and storage are limited, the batch-means estimator is able to provide an asymptotically exact confidence interval with comparably good performance. Furthermore, the computation of the Hessian matrix in the plug-in estimator is an "intrusive" requirement for SGD (Sullivan (2015)), that is, it is not available for legacy codes where only the SGD iterates are computed.

For example, if one has already obtained SGD iterates and wants to compute confidence intervals afterward, a noninstructive method like batch-means can be directly applied. Such a nonintrusive method that can operate with black-box SGD iterates is more desirable and welcomed by practitioners, as it only uses the existing SGD iterates without the need to change the original SGD code.

For the second part of our contribution, we further study the problem of confidence interval construction for x^* in high-dimensional linear regression based on SGD, where the dimensionality d can be much larger than the sample size n. In a high-dimensional setup, it is natural to solve a ℓ_1-regularized problem, $\min_x F(x) + \lambda\|x\|_1$, where $F(x)$ is defined in (1). A popular approach to solve it is the *proximal stochastic gradient approach* (see, e.g., Ghadimi and Lan (2012) and references therein). However, due to the proximal operator (i.e., the soft-thresholding operator for ℓ_1-regularized problem), the distribution of the average iterate $\bar{x}_n$ no longer converges to a multivariate normal distribution. To address this challenge, we use the recently proposed RADAR algorithm (Agarwal, Negahban and Wainwright (2012)), which is a variant of SGD, together with the debiasing approach (Javanmard and Montanari (2014), van de Geer et al. (2014), Zhang and Zhang (2014)). The standard debiasing method relies on solving d convex optimization problems (e.g., nodewise Lasso in van de Geer et al. (2014)) to construct an approximation of the inverse of the design covariance matrix. Each deterministic optimization problem requires a per-iteration complexity $O(nd)$, which is prohibitive when n is large. In contrast, we adopt the stochastic RADAR algorithm to solve these optimization problems, where each problem only requires one pass of the data with the per-iteration complexity $O(d)$. Moreover, since the resulting approximate inverse covariance matrix from the stochastic RADAR is not an exact solution of the corresponding optimization problem, the analysis of van de Geer et al. (2014), which heavily relies on the KKT condition, is no longer applicable. We provide a new analysis to establish the asymptotic normality of the obtained estimator of x^* from the stochastic optimization algorithm.

1.1. *Some related works on SGD.* There is a large body of literature on stochastic gradient approaches and their applications to statistical learning problems (see, e.g., Agarwal, Negahban and Wainwright (2012), Ghadimi and Lan (2012), Nesterov and Vial (2008), Roux, Schmidt and Bach (2012), Xiao (2010), Xiao and Zhang (2014), Zhang (2004) and references therein). Most works on SGD focus on the convergence rate of the objective function instead of the asymptotic distribution of the obtained solution. Thus, we only review a few closely related works with results on distributions.

Back in 1960s, Fabian (1968) studied the distribution of SGD iterates. However, without averaging, the asymptotic variance is inflated, and thus the resulting statistical inference would have a reduced power even if the asymptotic is known. Bach and Moulines (2011), Polyak and Juditsky (1992), Ruppert (1988) studied the averaged SGD (ASGD) and established the asymptotic normality and efficiency of the estimators. However, these works do not discuss the estimation of the asymptotic covariance.

A few works in the SGD literature (e.g., Nemirovski et al. (2008), Nesterov and Vial (2008)) show large deviation results of $\Pr(\|\bar{x}_n - x^*\|_2 > t) \le C(t)$ by combining the Markov inequality with the expected deviation of $\bar{x}_n$ to x^*. However, we note that large deviation results cannot be used to obtain asymptotically exact confidence intervals, which refer to the exact $1-q$ coverage as $n \to \infty$. That is, $\Pr(x^* \in \mathrm{CI}_q) \to 1-q$, where CI_q denotes the confidence interval. Deviation inequalities, which are unable to quantify the exact probability, fail to provide the exact $1-q$ coverage and will lead to wider confidence intervals. Moreover, note that the ℓ_2 bounds in the SGD literature are generally $O(\sigma\sqrt{\frac{d}{n}})$ (where σ^2 is the variance of the norm of the stochastic gradient) and do not imply a ℓ_∞ bound of size $O(\frac{\sigma}{\sqrt{n}})$, whereas a confidence interval for any single coordinate should be $O(\frac{\sigma}{\sqrt{n}})$ (the $O(\cdot)$ notation here does

not depend on d). Therefore, although d is fixed, ℓ_2-norm error bound results still lead to conservative confidence intervals. Instead, we will use the central limit theorem that shows $\lim \Pr(|\bar{x}_{n,j} - x^*| < z_{q/2}\sigma/\sqrt{n}) \to 1-q$, where $z_{q/2}$ is the $(1-q/2)$-quantile of the standard normal distribution. This allows us to construct an asymptotically exact confidence interval.

We also note that Toulis and Airoldi (2017) established the asymptotic normality for the *averaged implicit SGD* procedure, which is an algorithm different from ASGD. Moreover, this paper does not discuss the estimation of the asymptotic covariance, and thus their results cannot be directly used to obtain the confidence intervals.

1.2. *Notation and organization of the paper.* As a summary of notation, throughout the paper, we use $\|x\|_p$ to denote the vector ℓ_p-norm of x, $\|x\|_0$ the number of nonzero entries in x, $\|X\|$ the matrix operator norm of X and $\|X\|_\infty$ the elementwise ℓ_∞-norm of X (i.e., $\|X\|_\infty = \max_{i,j} |X_{ij}|$). For a square matrix X, we denote its trace by $\operatorname{tr}(X)$. For a positive semidefinite (PSD) matrix A, let $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ be its maximum and minimum eigenvalue. For a vector a of length d and any index subset $J \subseteq \{1, \dots, d\}$, we denote by a_J the subvector of a with the elements indexed by J and a_{-J} the subvector of a with the elements indexed by $\{1, \dots, d\} \backslash J$. Similarly, for a $d_1 \times d_2$ matrix X and two index subsets $R \subseteq \{1, \dots, d_1\}$ and $J \subseteq \{1, \dots, d_2\}$, we denote by $X_{R,J}$ the $|R| \times |J|$ submatrix of X with elements in rows in R and columns in J. When $R = \{1, \dots, d_1\}$ or $J = \{1, \dots, d_2\}$, we denote $X_{R,J}$ by $X_{\cdot,J}$ or $X_{R,\cdot}$, respectively. We use I to denote the identity matrix. The function $\Phi(\cdot)$ denotes the CDF of the standard normal distribution.

For any sequences $\{a_n\}$ and $\{b_n\}$ of positive numbers, we write $a_n \gtrsim b_n$ if $a_n \geq c b_n$ holds for all n large enough and some constant $c > 0$, $a_n \lesssim b_n$ if $b_n \gtrsim a_n$ holds, and $a_n \asymp b_n$ if $a_n \gtrsim b_n$ and $a_n \lesssim b_n$.

The rest of the paper is organized as follows. In Section 2, we provide more background of SGD and detailed results from Polyak and Juditsky (1992). In Section 3, we provide the assumptions and some error bounds on SGD iterates. In Section 4, we propose the plug-in estimator and batch-means estimator for estimating the asymptotic covariance of $\bar{x}_n$ from ASGD. In Section 5, we discuss how to conduct inference for high-dimensional linear regression. In Section 6, we demonstrate the proposed methods by simulated experiments. Further discussions appear in Section 7 and all proofs are given in the Supplementary Material (Chen et al. (2020)).

2. Background. In the classical work of Polyak and Juditsky (1992), the SGD method was introduced in a form equivalent with (2) to facilitate the analysis. In particular, the iteration is given by

$$x_n = x_{n-1} - \eta_n \nabla F(x_{n-1}) + \eta_n \xi_n, \tag{5}$$

where $\xi_n := \nabla F(x_{n-1}) - \nabla f(x_{n-1}, \zeta_n)$. The formulation (5) decomposes the descent into two parts: $\nabla F(x_{n-1})$ represents the direction of population gradient which is the major driving force behind the convergence of SGD, and ξ_n is a martingale difference sequence under Assumption 3.2 (see below). That is, $\mathbb{E}_{n-1}[\xi_n] = \nabla F(x_{n-1}) - \mathbb{E}_{n-1} \nabla f(x_{n-1}, \zeta_n) = 0$. Here and in the sequel, $\mathbb{E}_n(\cdot)$ denotes the conditional expectation $\mathbb{E}(\cdot|\mathcal{F}_n)$, where $\mathcal{F}_n$ is the σ-algebra generated by $\{\zeta_1, \dots, \zeta_n\}$ (ζ_k is the kth sample). Let $\Delta_n := x_n - x^*$ be the error of the nth iterate. It it noteworthy that by subtracting x^* from both sides of (5), the recursion (5) is equivalent to

$$\Delta_n = \Delta_{n-1} - \eta_n \nabla F(x_{n-1}) + \eta_n \xi_n, \tag{6}$$

which will be extensively used throughout the paper.

Given the SGD recursion in the form of (6) and under suitable assumptions (see Section 3 below), Theorem 2 of Polyak and Juditsky (1992) shows that when the step size sequence $\eta_i = \eta i^{-\alpha}$, $i = 1, 2, \ldots, n$ with $\alpha \in (1/2, 1)$, we have

$$\sqrt{n} \cdot \overline{\Delta}_n \Rightarrow \mathcal{N}\big(0, A^{-1} S A^{-1}\big) \quad \text{if } \alpha \in \left(\frac{1}{2}, 1\right), \tag{7}$$

where $\overline{\Delta}_n = \frac{1}{n}\sum_{i=1}^n \Delta_i = \bar{x}_n - x^*$. Based on this limiting distribution result, we only need to estimate the asymptotic covariance matrix $A^{-1}SA^{-1}$. Then we can form the confidence interval $\bar{x}_{n,j} \pm z_{q/2}\hat{\sigma}_{jj}$, where $\hat{\sigma}_{jj}$ is a consistent estimator of $(A^{-1}SA^{-1})_{jj}$ and $z_{q/2}$ is the $(1 - q/2)$-quantile of the standard normal distribution (i.e., $z_{q/2} = \Phi^{-1}(1 - q/2)$ and $\Phi(\cdot)$ is the CDF of the standard normal distribution). Therefore, the main purpose of the paper is to provide consistent estimators of the asymptotic covariance matrix.

REMARK 2.1. In the model well-specified case, $\overline{x}_n$ is an asymptotically efficient estimator of the true model parameter x^* according to (7). In particular, suppose ζ comes from the probability distribution Π with density $p_{x^*}(\zeta)$ parameterized by x^*. If the loss function $f(x, \zeta) = -\log p_x(\zeta)$ is the negative log-likelihood, under certain regularity conditions, one can show that

$$A = \nabla^2 \mathbb{E}\big[-\log p_{x^*}(\zeta)\big] = \mathbb{E}\big(-\nabla \log p_{x^*}(\zeta)\big)\big(-\nabla \log p_{x^*}(\zeta)\big)^T = S = I\big(x^*\big).$$

Here, $I = I(x^*)$ is the Fisher information matrix. Therefore, the limiting covariance matrix $A^{-1}SA^{-1} = I^{-1}$ achieves the Cramér–Rao lower bound, which indicates that $\bar{x}_n$ is asymptotically efficient. It is worth noting that the asymptotic normality result (7) does not require that the model is well specified. In a model misspecified case, the asymptotic distribution of $\bar{x}_n$ is centered at x^*, where x^* is the unique minimizer of $F(x)$ and the asymptotic covariance $A^{-1}SA^{-1}$ is of the so-called "sandwich covariance" form (e.g., see Buja et al. (2013)).

To illustrate this SGD recursion in (6) and the form of A and S, we consider the following two motivating examples.

EXAMPLE 2.1 (Linear regression). Under the classical linear regression setup, let the nth sample be $\zeta_n = (a_n, b_n)$, where the input $a_n \in \mathbb{R}^d$ is a sequence of random vectors independently drawn from the same multivariate distribution and the response $b_n \in \mathbb{R}$ follows a linear model, $b_n = a_n^T x^* + \varepsilon_n$. Here, $x^* \in \mathbb{R}^d$ represents the true parameters of the linear model, and $\{\varepsilon_n\}$ are independently and identically distributed (*i.i.d.*) centered random variables, which are uncorrelated with a_n. For simplicity, we assume a_n and ε_n have all moments being finite. Given $\zeta_n = (a_n, b_n)$, the loss function at x is a quadratic one:

$$f(x, \zeta_n) = \frac{1}{2}\big(a_n^T x - b_n\big)^2$$

and the true parameters $x^* = \operatorname{argmin}_x (F(x) := \mathbb{E} f(x, \zeta))$. Given the loss function, the SGD iterates in (2) become, $x_n = x_{n-1} - \eta_n a_n (a_n^T x_{n-1} - b_n)$. This can also be written in the form of (5) as

$$x_n = x_{n-1} - \eta_n A \Delta_{n-1} + \eta_n \xi_n, \quad \xi_n := \big(A - a_n a_n^T\big)\Delta_{n-1} + a_n \varepsilon_n,$$

where $A = \mathbb{E} a_n a_n^T$ is the population gram matrix of a_n. It is easy to find that

$$F(x) = \frac{1}{2}\big(x - x^*\big)^T A\big(x - x^*\big) + \mathbb{E}\varepsilon^2,$$

which implies that $\nabla F(x) = A(x - x^*)$ and $\nabla^2 F(x) = A$ for all x. As for matrix S, it is given by $S := \mathbb{E}([\nabla f(x^*, \zeta)][\nabla f(x^*, \zeta)]^T) = \mathbb{E}\varepsilon_n^2 a_n a_n^T$.

EXAMPLE 2.2 (Logistic regression). One of the most popular applications for general loss in statistics is the logistic regression for binary classification problems. In particular, the logistic model assumes that the binary response $b_n \in \{-1, 1\}$ is generated by the following probabilistic model:

$$\Pr(b_n|a_n) = \frac{1}{1+\exp(-b_n\langle a_n, x^*\rangle)},$$

where a_n is an *i.i.d.* sequence. The population objective function is given by $F(x) = \mathbb{E}f(x,\zeta_n) = \mathbb{E}\log(1+\exp(-b_n\langle a_n, x\rangle))$. Let $\varphi(x) := \frac{1}{1+\exp(-x)}$ denote the sigmoid function, we have $\nabla f(x,\zeta_n) = -\varphi(-b_n\langle a_n, x\rangle)b_n a_n$. Moreover, we have the formulation of matrix A and S as

$$A = S = \mathbb{E}\frac{a_n a_n^T}{[1+\exp(\langle a_n, x^*\rangle)][1+\exp(-\langle a_n, x^*\rangle)]}. \tag{8}$$

3. Assumptions and error bounds. In this section, we provide the assumptions used in the fixed-dimensional case and then provide some useful error bounds on Δ_n. We first make the following standard assumption on the population loss function $F(x)$.

ASSUMPTION 3.1 (Strong convexity and Lipschitz continuity of the gradient). Assume that the objective function $F(x)$ is continuously differentiable and strongly convex with parameter $\mu > 0$, that is, for any x_1 and x_2,

$$F(x_2) \geq F(x_1) + \langle \nabla F(x_1), x_2 - x_1\rangle + \frac{\mu}{2}\|x_1 - x_2\|_2^2.$$

Further, assume that $\nabla^2 F(x^*)$ exists, and $\nabla F(x)$ is Lipschitz continuous with a constant L_F, that is, for any x_1 and x_2, $\|\nabla F(x_1) - \nabla F(x_2)\|_2 \leq L_F\|x_1 - x_2\|_2$.

Note that the strong convexity of $F(x)$ was adopted by Polyak and Juditsky (1992) (see Assumption 4.1 in Polyak and Juditsky (1992)) to derive the limiting distribution of averaged SGD, which serves as the basis of our work. In fact, the strong convexity of $F(x)$ implies $\lambda_{\min}(A) = \lambda_{\min}(\nabla^2 F(x^*)) \geq \mu$ is an important condition for parameter estimation and inference. There are recent works in optimization on relaxing the strong convexity assumption (e.g., Bach and Moulines (2013)), but they were only able to obtain fast convergence rates in terms of the objective value $F(\bar{x}_n) - F(x^*)$.

We further assume that the martingale difference ξ_n satisfies the following conditions.

ASSUMPTION 3.2. The following hold for the sequence $\xi_n = \nabla F(x_{n-1}) - \nabla f(x_{n-1},\zeta_n)$:

1. Assume that $f(x,\zeta)$ is continuously differentiable in x for any ζ and $\|\nabla f(x,\zeta)\|_2$ is uniformly integrable for any x so that $\mathbb{E}_{n-1}\xi_n = 0$.

2. The conditional covariance of ξ_n has an expansion around $x = x^*$: $\mathbb{E}_{n-1}\xi_n\xi_n^T = S + \Sigma(\Delta_{n-1})$, and there exists constants Σ_1 and $\Sigma_2 > 0$ such that for any $\Delta \in \mathbb{R}^d$.

$$\|\Sigma(\Delta)\| \leq \Sigma_1\|\Delta\|_2 + \Sigma_2\|\Delta\|_2^2, \qquad |\mathrm{tr}(\Sigma(\Delta))| \leq \Sigma_1\|\Delta\|_2 + \Sigma_2\|\Delta\|_2^2.$$

Note that S is the covariance matrix of $\nabla f(x^*,\zeta)$ defined in (3).

3. There exists constants Σ_3, Σ_4 such that the fourth conditional moment of ξ_n is bounded by $\mathbb{E}_{n-1}\|\xi_n\|_2^4 \leq \Sigma_3 + \Sigma_4\|\Delta_{n-1}\|_2^4$.

For part 1, we note that our assumption on $f(x,\zeta)$ guarantees that Leibniz's integration rule holds, that is, $\mathbb{E}_{\zeta\sim\Pi}\nabla f(x,\zeta) = \nabla F(x)$ for all x. Therefore, we have $\mathbb{E}_{n-1}\xi_n = 0$, which implies that ξ_n is a martingale difference sequence. Assumption 3.2 is a mild condition over the regularity and boundedness of the loss function. In fact, one can easily verify Assumption 3.2 using the following lemma.

LEMMA 3.1. *If there is a function $H(\zeta)$ with bounded fourth moment, such that the Hessian of $f(x,\zeta)$ is bounded by*

$$\|\nabla^2 f(x,\zeta)\| \le H(\zeta)$$

for all x, and $\nabla f(x^*,\zeta)$ have a bounded fourth moment, then Assumption 3.2 holds with $\Sigma_1 = 2\sqrt{\mathbb{E}\|\nabla f(x^*,\zeta)\|_2^2 \mathbb{E}H(\zeta)^2}$, $\Sigma_2 = 4\mathbb{E}H(\zeta)^2$, $\Sigma_3 = 8\mathbb{E}\|\nabla f(x^*,\zeta)\|_2^4$ and $\Sigma_4 = 64\mathbb{E}H(\zeta)^4$.

Although we consider the fixed-dimensional case, it is still of practical interest to investigate the dimension dependence in our results. The dimension dependence is rather complicated since our results involve a number of constants in Assumption 3.1 and 3.2 that all depend on the dimension d (e.g., L_F, Σ_1, Σ_2, Σ_3, Σ_4, $\mathrm{tr}(S)$). For example, $\mathrm{tr}(S)$ grows with d. Moreover, the way it grows depends on how S is configured. Therefore, for the ease of presentation, we define the following quantity:

$$C_d := \max\{L_F, \Sigma_1^{\frac{2}{3}}, \sqrt{\Sigma_2}, \sqrt{\Sigma_3}, \Sigma_4^{\frac{1}{4}}, \mathrm{tr}(S)\}. \tag{9}$$

In both linear and logistic regression, C_d increases linearly in d (see Section A in the Supplementary Material). We will state our results in terms of this single quantity C_d. We also assume $\|x_0 - x^*\|_2^2 = O(C_d)$, and there is a universal constant c such that the step size satisfies $\eta_i C_d \le c\mu$ for all i. Note that the choice of step sizes does not sacrifice much of generality since when d is a constant, we could always ignore the first a few iterations, which is usually considered as the "burn in" stage. Also, for the starting point x_0, if all the components of $x_0 - x^*$ are bounded by a constant, it naturally satisfies $\|x_0 - x^*\|_2^2 = O(d)$.

In the sequel, we will impose Assumptions 3.1 and 3.2. In Section A of the Supplementary Material, we show that Assumptions 3.1 and 3.2 hold on our motivating examples of linear and logistic regression (see Examples 2.1 and 2.2). Under these assumptions, the classical works (Polyak and Juditsky (1992), Ruppert (1988)) establish the asymptotic normality and efficiency of the $\bar{x}_n$ (see (4) and Remark 2.1). Moreover, we could obtain the following error bounds on the SGD iterates.

LEMMA 3.2. *Under Assumptions 3.1 and 3.2, if the step size is chosen to be $\eta_n = \eta n^{-\alpha}$ with $\alpha \in (0,1)$, the iterates of error $\Delta_n = x_n - x^*$ satisfy the following*:

$$\mathbb{E}\|\Delta_n\|_2^k \lesssim n^{-k\alpha/2}\big(C_d^{k/2} + \|\Delta_{n_0}\|_2^k\big), \quad k = 1, 2, 4.$$

The proof of Lemma 3.2 is provided in Section 3 of the Supplementary Material. A result similar to Lemma 3.2 providing the convergence of $\|\Delta_n\|_2$ and $\|\Delta_n\|_2^2$ has been shown in Bach and Moulines (2011) (see Theorem 1 therein). Here, we provide simpler bounds on conditional moments of Δ_n and extend the results in Bach and Moulines (2011) to the fourth moment bound, since we need to access the variance of a variance estimator. This result also tells us how the error decorrelates in terms of the number of iterations. Our proof strategy is similar to Bach and Moulines (2011) in that we setup up a recursive formula for the Δ_n term, and then show it decays at a certain rate by leveraging the convexity of $F(x)$.

4. Estimators for asymptotic covariance. Following the inference procedures illustrated above, when d is fixed and $n \to \infty$, it is essential to estimate the asymptotic covariance matrix $A^{-1}SA^{-1}$. In this section, we will propose two consistent estimators, the plug-in estimator and the batch-means estimator.

4.1. *Plug-in estimator.* The idea of the plug-in estimator is to separately estimate A and S by some $\widehat{A}$ and $\widehat{S}$ and use $\widehat{A}^{-1}\widehat{S}\widehat{A}^{-1}$ as an estimator of $A^{-1}SA^{-1}$. Since x_i converges to x^*, according to the definitions of A and S in (3), an intuitive way to construct $\widehat{A}$ and $\widehat{S}$ is to use the sample estimate

$$A_n := \frac{1}{n}\sum_{i=1}^{n} \nabla^2 f(x_{i-1}, \zeta_i), \qquad S_n := \frac{1}{n}\sum_{i=1}^{n} \nabla f(x_{i-1}, \zeta_i)\nabla f(x_{i-1}, \zeta_i)^T,$$

as long as the information of $\nabla^2 f(x_{i-1}, \zeta_i)$ is available. It is worthwhile noting that each summand in A_n and S_n involves different x_{i-1}. Therefore, A_n and S_n can be computed in an online fashion without the need of storing all the data.

Since we are interested in estimating A^{-1}, it is necessary to avoid the possible singularity of A_n from statistical randomness. Therefore, we propose to use thresholding estimator $\widetilde{A}_n$, which is strictly positive definite. In particular, fix $\delta > 0$, and let $\Psi D_n \Psi^T$ be the eigenvalue decomposition of A_n, where D_n is a nonnegative diagonal matrix. We construct the thresholding estimator $\widetilde{A}_n$:

$$\widetilde{A}_n = \Psi \widetilde{D}_n \Psi^T, \quad (\widetilde{D}_n)_{i,i} = \max\{\delta, (D_n)_{i,i}\}.$$

By construction, it is guaranteed that $\widetilde{A}_n$ is invertible. With the construction of S_n and $\widetilde{A}_n$ in place, we propose the *plug-in estimator* as $\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1}$. Our goal is to establish the consistency of the plug-in estimator, that is,

$$\mathbb{E}\|\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1} - A^{-1}SA^{-1}\| \longrightarrow 0 \quad \text{as } n \to \infty.$$

Since this estimator relies on the Hessian matrix of the loss function, we need an additional assumption to establish the consistency.

ASSUMPTION 4.1. There are constants L_2 and L_4 such that for all x,

$$\begin{aligned} \mathbb{E}\|\nabla^2 f(x,\zeta) - \nabla^2 f(x^*,\zeta)\| &\le L_2 \|x - x^*\|_2, \\ \|\mathbb{E}[\nabla^2 f(x^*,\zeta)]^2 - A^2\| &\le L_4. \end{aligned} \tag{10}$$

Moreover, we assume that for the choice of δ, we have $\lambda_{\min}(A) > \delta$.

We note that it is easy to verify that (10) holds for the two motivating examples in Section 2. For quadratic loss, the Hessian matrix at any x is A itself, and (8) gives the Hessian for the logistic loss, which is Lipschitz in x and also bounded. In addition, according to Assumption 3.1, we have $\lambda_{\min}(\nabla^2 F(x)) \ge \mu$ for any x, and thus $\lambda_{\min}(A) \ge \mu$. Therefore, a valid choice of δ satisfying Assumption 4.1 always exists.

To track the dependence of our results on dimension, we assume L_2 and L_4 are also controlled by C_d in (9) as $L_2 \lesssim C_d^{3/2}$, $L_4 \lesssim C_d^2$. Lemmas A.1 and A.2 in the Supplementary Material verify this requirement is satisfied in linear and logistic regression.

With this additional assumption, we first establish the consistency of the sample estimate A_n and S_n in the following lemma.

LEMMA 4.1. *Under Assumptions 3.1, 3.2 and 4.1, the following holds*:

$$\mathbb{E}\|A_n - A\| \lesssim C_d^2 n^{-\frac{\alpha}{2}}, \qquad \mathbb{E}\|S_n - S\| \lesssim C_d^2 n^{-\frac{\alpha}{2}} + C_d^3 n^{-\alpha},$$

where $\alpha \in (0,1)$ *is given in the step size sequence* $\eta_i = \eta i^{-\alpha}$, $i = 1, 2, \ldots, n$.

The proof of Lemma 4.1 is provided in Section C.1 of the Supplementary Material. Using Lemma 4.1 and a matrix perturbation inequality for the inverse of a matrix (see Lemma C.1 in Section C.2 of the Supplementary Material), we obtain the consistency result of the proposed plug-in estimator $\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1}$:

THEOREM 4.2 (Error rate of the plug-in estimator). *Under Assumptions 3.1, 3.2 and 4.1, the thresholded plug-in estimator initialized from any bounded x_0 converges to the asymptotic covariance matrix,*

$$\mathbb{E}\big\|\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1} - A^{-1} S A^{-1}\big\| \lesssim \|S\|\big(C_d^2 n^{-\frac{\alpha}{2}} + C_d^3 n^{-\alpha}\big), \tag{11}$$

where $\alpha \in (0, 1)$ is given in the step size sequence $\eta_i = \eta i^{-\alpha}$, $i = 1, 2, \ldots, n$. When C_d is a constant, the right-hand side of (11) is dominated by $O(n^{-\frac{\alpha}{2}})$.

REMARK 4.2. In practice, we usually do not need to perform the thresholding step, since A_n is positive definite with high probability as A_n is close to A. The thresholding step is mainly for obtaining the expected error bound in Theorem 4.2. In fact, without the thresholding step, we are still able to the obtain the following error bound. In our numerical experiments, we do not apply the thresholding procedure and the obtained A_n's are always invertible.

COROLLARY 4.3. *Under Assumptions 3.1, 3.2 and 4.1, as $n \to \infty$,*

$$\big\|A_n^{-1} S_n A_n^{-1} - A^{-1} S A^{-1}\big\| = O_p\big(\|S\|\big(C_d^2 n^{-\frac{\alpha}{2}} + C_d^3 n^{-\alpha}\big)\big).$$

We also note that since the elementwise ℓ_∞-norm is bounded from above by the matrix operator norm, we have $\mathbb{E}\max_{ij} |(\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1} - A^{-1} S A^{-1})_{ij}|$ converges to zero as $n \to \infty$ according Theorem 4.2. Therefore, $(A^{-1} S A^{-1})_{jj}^{1/2}$ can be estimated by $\hat{\sigma}_{n,j}^P = (\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1})_{jj}^{1/2}$ for the construction of confidence intervals. In particular, we have the following corollary, which shows that $\bar{x}_{n,j} \pm z_{q/2} \hat{\sigma}_{n,j}^P$ is an asymptotic exact confidence interval.

COROLLARY 4.4. *Under the assumptions of Theorem 4.2, if the step size is chosen to be $\eta_i = \eta i^{-\alpha}$ with $\alpha \in (\frac{1}{2}, 1)$, when d is fixed and $n \to \infty$,*

$$\Pr\big(\bar{x}_{n,j} - z_{q/2} \hat{\sigma}_{n,j}^P \le x_j^* \le \bar{x}_{n,j} + z_{q/2} \hat{\sigma}_{n,j}^P\big) \to 1 - q.$$

Proof of Corollary 4.4 is given in Section D.5 of the Supplementary Material. Note that while Theorem 4.2 holds for all $\alpha \in (0, 1)$, the asymptotic normality in (7) holds only when $\alpha \in (\frac{1}{2}, 1)$. Thus, Corollary 4.4 requires that $\alpha \in (\frac{1}{2}, 1)$.

4.2. *Batch-means estimator.* Although the plug-in estimator is intuitive, it requires the computation of the Hessian matrix and its inverse, as well as an additional Assumption 4.1 on the Lipschitz condition of the Hessian matrix. In this section, we develop the batch-means estimator, which only uses the iterates from SGD without requiring computation of any additional quantities. Intuitively, if all iterates are independent and share the same distribution, the asymptotic covariance can be directly estimated by the sample covariance, $\frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T$. Unfortunately, the SGD iterates are far from independent. To understand the correlation between two consecutive iterates, we note that for sufficiently large n such that x_{n-1} is close to x^*, by the Taylor expansion of $\nabla F(x_{n-1})$ at x^*, we have $\nabla F(x_{n-1}) \approx \nabla F(x^*) + \nabla^2 F(x^*)(x_{n-1} - x^*) = A\Delta_{n-1}$, where $\nabla F(x^*) = 0$ by the first-order condition and $A = \nabla^2 F(x^*)$. Combining this with the recursion in (6), we have for

sufficiently large n,

$$\Delta_n \approx (I_d - \eta_n A)\Delta_{n-1} + \eta_n \xi_n. \tag{12}$$

Based on (12), the strength of correlation between Δ_n and Δ_{n-1} can be approximated by $\|I_d - \eta_n A\|$, which is very close to 1 as $\eta_n \asymp n^{-\alpha}$. To address the challenge of strong correlation among neighboring iterates, we split the entire sequence of iterates into batches with carefully chosen batch sizes. In particular, we split n iterates of SGD $\{x_1, \ldots, x_n\}$ into $M+1$ batches with sizes $n_0, n_1, \ldots, n_M$:

$$\underbrace{\{x_{s_0}, \ldots, x_{e_0}\}}_{\text{0th batch}}, \underbrace{\{x_{s_1}, \ldots x_{e_1}\}}_{\text{1st batch}}, \ldots, \underbrace{\{x_{s_M}, \ldots, x_{e_M}\}}_{M\text{th batch}}.$$

Here, s_k and e_k are the starting and ending index of kth batch with $s_0 = 1$, $s_k = e_{k-1} + 1$, $n_k = e_k - s_k + 1$, and $e_M = n$. We treat the 0th batch as the "burn-in stage." More precisely, the iterates $\{x_{s_0}, \ldots, x_{e_0}\}$ will not be used for constructing the batch-means estimator since the step sizes are not small enough and the corresponding iterates in the 0th batch are far away from the optimum. The batch-means estimator is given by the following:

$$\frac{1}{M}\sum_{k=1}^{M} n_k (\overline{X}_{n_k} - \overline{X}_M)(\overline{X}_{n_k} - \overline{X}_M)^T, \tag{13}$$

where $\overline{X}_{n_k} := \frac{1}{n_k}\sum_{i=s_k}^{e_k} x_i$ is the mean of the iterates for the kth batch and $\overline{X}_M := \frac{1}{e_M - e_0}\sum_{i=s_1}^{e_M} x_i$ is the the mean of all the iterates except for the 0th batch.

Note that when batch sizes n_k are predetermined, we may rewrite (13) in the following form:

$$\frac{1}{M}\sum_{k=1}^{M} n_k \overline{X}_{n_k}\overline{X}_{n_k}^T + \frac{n}{M}\overline{X}_M\overline{X}_M^T - 2\left(\frac{1}{M}\sum_{k=1}^{M} n_k \overline{X}_{n_k}\right)\overline{X}_M^T. \tag{14}$$

Here, $\overline{X}_M$, $\frac{1}{M}\sum_{k=1}^{M} n_k \overline{X}_{n_k}\overline{X}_{n_k}$ and $\frac{1}{M}\sum_{k=1}^{M} n_k \overline{X}_{n_k}$ can be updated recursively so that there is no need to store all the batch-means $\{\overline{X}_{n_k}\}$. In other words, the memory requirement for the batch-means estimator is only $O(d^2)$ instead of $O(Md^2)$.

Intuitively, the reason why our batch-means estimator with increasing batch size can overcome the strong dependence between iterates is as follows. Although the correlation between neighboring iterates is strong, it decays exponentially for far-apart iterates. Roughly speaking, by (12), for large j and k, the strength of correlation between Δ_j and Δ_k is approximately

$$\prod_{i=j}^{k-1} \|I_d - \eta_{i+1} A\| \approx \exp\left(-\lambda_{\min}(A)\sum_{i=j}^{k-1}\eta_{i+1}\right). \tag{15}$$

Therefore, the correlations between the batch-means $\overline{X}_{n_k}$ are close to zero if the batch sizes are large enough, in which case different batch-means can be roughly treated as independent. As a consequence, the sample covariance gathered from the batch-means will serve as a good estimator of the true asymptotic covariance.

The remaining difficulty is how to determine the batch sizes. The approximation of correlation given by (15) provides us a clear clue. If we want the correlation between two neighboring batches to be on the order of $\exp(-cN)$, where N (with $N \to \infty$) is a parameter controlling the amount of decorrelation and c is a constant, we need $\sum_{i=s_k}^{e_k}\eta_i \asymp N$ for every batch k. When $\eta_i = \eta i^{-\alpha}$, $\sum_{i=s_k}^{e_k}\eta_i \asymp (e_k^{1-\alpha} - e_{k-1}^{1-\alpha})$, which leads to the following batch size setting:

$$e_k = \big((k+1)N\big)^{\frac{1}{1-\alpha}}, \quad k = 0, \ldots, M, \tag{16}$$

where e_k is the ending point for the kth batch. There are two practical scenarios to apply the proposed batch-means estimator:

- Total number of iterates n is given: Noting that $e_M = n$, the decorrelation strength factor N takes the following form:

$$N = \frac{n^{1-\alpha}}{M+1}, \tag{17}$$

  where M is the number of batches. Based on the result of Theorem 4.3 below, it is preferable to take $N = n^{\frac{1-\alpha}{2}}$.
- When n is unknown (but sufficiently large): Given a target error bound ϵ, we pick an $N \asymp \epsilon^{-2}$. Then, we receive the online data and batch the SGD iterates according to (16). When the number of batches M is sufficiently large (e.g., the upper bound in (18) below is smaller than ϵ), we stop our SGD procedure and output the batch-means estimator.

Under this setting, the batch-means covariance estimator (13) is consistent as shown in the following theorem.

THEOREM 4.3 (Error rate of the batch-means estimator). *Under Assumptions 3.1 and 3.2, when d is fixed and the step size is chosen to be $\eta_i = \eta i^{-\alpha}$ with $\alpha \in (\frac{1}{2}, 1)$, the batch-means estimator initialized by any bounded x_0 is a consistent estimator. In particular, for sufficiently large N and M, we have*

$$\begin{aligned}\mathbb{E}\left\| M^{-1}\sum_{k=1}^{M} n_k(\overline{X}_{n_k} - \overline{X}_M)(\overline{X}_{n_k} - \overline{X}_M)^T - A^{-1}SA^{-1}\right\| \\ \lesssim C_d M^{-\frac{1}{2}} + C_d N^{-\frac{1}{2}} + C_d^{\frac{3}{2}}(MN)^{-\frac{\alpha}{4-4\alpha}} + C_d^2 M^{-1} + C_d^3 M^{-1} N^{\frac{1-2\alpha}{1-\alpha}}.\end{aligned} \tag{18}$$

As $n \to \infty$, by (17), we can choose $M, N \to \infty$, and thus the right- hand side of (18) will converge to zero for any $\alpha \in (1/2, 1)$, which shows the consistency of the proposed covariance estimator. When d is fixed, C_d is a constant, and it is straightforward to see that the right-hand side of (18) is dominated by $C_d(M^{-\frac{1}{2}} + N^{-\frac{1}{2}})$. Therefore, according to (17) (i.e., $N(M+1) = n^{1-\alpha}$), we have the following Corollary 4.5 that suggests the optimal order of M.

COROLLARY 4.5. *Under Assumptions 3.1 and 3.2, when d is fixed and n is sufficiently large, by choosing the step size $\eta_i = \eta i^{-\alpha}$ with $\alpha \in (\frac{1}{2}, 1)$, $M \asymp n^{\frac{1-\alpha}{2}}$ and $N \asymp n^{\frac{1-\alpha}{2}}$, we have*

$$\begin{aligned}\mathbb{E}\left\| M^{-1}\sum_{k=1}^{M} n_k(\overline{X}_{n_k} - \overline{X}_M)(\overline{X}_{n_k} - \overline{X}_M)^T - A^{-1}SA^{-1}\right\| \\ \lesssim C_d n^{-\frac{1-\alpha}{4}} + C_d^{\frac{3}{2}} n^{-\frac{\alpha}{4}} + C_d^2 n^{-\frac{1-\alpha}{2}} + C_d^3 n^{-\alpha}.\end{aligned} \tag{19}$$

When C_d is a constant, the right-hand side of (19) is dominated by $O(n^{-\frac{1-\alpha}{4}})$.

As we will show in simulations in Section 6, wide choices between $M = n^{0.2}$ to $M = n^{0.3}$ lead to reasonably good coverage rates when α is close to 1/2. Moreover, since $\alpha \in (1/2, 1)$, the convergence rate $n^{-\frac{1-\alpha}{4}}$ is slower than the rate of the plug-in estimator $n^{-\frac{\alpha}{2}}$. Although batch-means estimator has a slower convergence rate, the next corollary shows that this method still constructs asymptotic exact confidence intervals.

COROLLARY 4.6. *Under the assumptions of Theorem 4.3, when d is fixed, $n \to \infty$, and the step size $\eta_i = \eta i^{-\alpha}$ with $\alpha \in (\frac{1}{2}, 1)$, we have that*

$$\Pr\big(\bar{x}_{n,j} - z_{q/2}\hat{\sigma}^B_{n,j} \le x^*_j \le \bar{x}_{n,j} + z_{q/2}\hat{\sigma}^B_{n,j}\big) \to 1 - q,$$

where $\hat{\sigma}^B_{n,j} := [M^{-1}\sum_{k=1}^M n_k(\overline{X}_{n_k} - \overline{X}_M)(\overline{X}_{n_k} - \overline{X}_M)^T]^{1/2}_{j,j}$.

The proof is identical to the one of Corollary 4.4 and, therefore, omitted.

4.3. *Intuition behind the proof.* Now let us provide the main idea behind the proof of Theorem 4.3. Recall that the SGD recursion in (6) can be approximated by (12): $\Delta_n \approx (I_d - \eta_n A)\Delta_{n-1} + \eta_n \xi_n$. We replace "$\approx$" by the equal sign and define an auxiliary sequence U_n:

$$U_n = U_{n-1} - \eta_n A U_{n-1} + \eta_n \xi_n, \quad U_0 = \Delta_0. \tag{20}$$

Note that $\Delta_n = U_n$ in the linear model setting, but our proof applies to *nonlinear models (e.g., generalized linear models)*. For a nonlinear model, the high-level idea of the proof consists of two steps:

1. Establishing the consistency (and the rate of convergence) of the batch-means estimator based on the sequence U_n;
2. Quantifying the difference between Δ_n and U_n, where Δ_n in (6) is the original sequence of interest generated from SGD for general loss functions, and U_n in (20) is its auxiliary linear approximation sequence.

In fact, the sequence U_n is the so-called "oracle iterate sequence," which has also been considered in Polyak and Juditsky (1992). It can be written in a more explicit form:

$$U_n = \prod_{k=1}^n (I - \eta_k A)U_0 + \sum_{m=1}^n \prod_{k=m+1}^n (I - \eta_k A)\eta_m \xi_m. \tag{21}$$

Given the sequence U_n, we construct the batch-means estimator based on U_n as $\frac{1}{M}\sum_{k=1}^M n_k(\overline{U}_{n_k} - \overline{U}_M)(\overline{U}_{n_k} - \overline{U}_M)^T$, where $\overline{U}_{n_k}$ and $\overline{U}_M$ are defined as in (13) with x_i being replaced by U_i. The analysis of the batch-means estimator from U_n is simpler than that from SGD iterates x_n since the expression of U_n in (21) only involves the product of matrices and the martingale differences ξ_m. In particular, we establish the consistency of the batch-means estimator based on U_n in the following lemma.

LEMMA 4.7. *Under Assumptions 3.1 and 3.2, when d is fixed and the step size is chosen to be $\eta_i = \eta i^{-\alpha}$ with $\alpha \in (\frac{1}{2}, 1)$, the batch-means estimator based on the sequence U_n with any bounded U_0 satisfies the following inequality for sufficiently large N and M:*

$$\mathbb{E}\left\| M^{-1}\sum_{k=1}^M n_k(\overline{U}_{n_k} - \overline{U}_M)(\overline{U}_{n_k} - \overline{U}_M)^T - A^{-1}SA^{-1} \right\|$$
$$\lesssim C_d M^{-\frac{1}{2}} + C_d N^{-\frac{1}{2}} + C_d^{\frac{3}{2}}(MN)^{-\frac{\alpha}{4-4\alpha}}.$$

The proof of Lemma 4.7 is provided in Section D.3 of the Supplementary Material. With Lemma 4.7 in place, to obtain the desired consistency result in Theorem 4.3, we only need to study the difference between Δ_n and U_n. In particular, let $\delta_n := \Delta_n - U_n$. We have the following recursion:

$$\begin{aligned}\delta_n &= \Delta_{n-1} - U_{n-1} - \eta_n \nabla F(x_{n-1}) - \eta_n A U_{n-1} \\ &= \delta_{n-1} - \eta_n A \delta_{n-1} + \eta_n\big(A\Delta_{n-1} - \nabla F(x_{n-1})\big).\end{aligned}$$

Notably, by replacing ξ_n in (20) with $A\Delta_{n-1} - \nabla F(\Delta_{n-1})$, δ_n follows a similar recursion relationship to that of the sequence U_n. Based on this observation, we show that δ_n is a sequence of small numbers, and hence Δ_n and U_n are close to each other. Combining this with Lemma 4.7, we will reach the conclusion in Theorem 4.3 (see Section D.4 in the Supplementary Material for the rigorous proof).

5. High-dimensional linear regression. In Sections 4.1 and 4.2, we assumed that the dimension d is fixed while $n \to \infty$. However in high-dimensional settings, it is often the case that $d \asymp n$ or $n = o(d)$. Below we consider a high-dimensional linear model $b_i = a_i^T x^* + \varepsilon_i$, where x^* is s-sparse (i.e., $\|x\|_0 \le s$) and let $S = \{j : x_j^* \neq 0\}$ be the support of true regression coefficients. Each covariate $a_i \in \mathbb{R}^d$ is an *i.i.d.* sub-Gaussian random vector from a common population a with the covariance matrix A, and $\varepsilon_i \sim N(0, \sigma^2)$. For simplicity, we assume that σ is known. For high-dimensional linear regression, one of the most popular estimators is the Lasso estimator, denoted by $\widehat{x}_{\text{Lasso}}$. That is,

$$\widehat{x}_{\text{Lasso}} = \frac{1}{2n} \operatorname*{argmin}_{x \in \mathbb{R}^d} \|b - Dx\|_2^2 + \lambda \|x\|_1,$$

where $D = [a_1, \ldots, a_n]^T \in \mathbb{R}^{n \times d}$ is the design matrix, $b = [b_1, \ldots, b_n]^T \in \mathbb{R}^{n \times 1}$ is the response vector.

As suggested by earlier work (see, e.g., Belloni and Chernozhukov (2013), Bühlmann and van de Geer (2011), Meinshausen, Meier and Bühlmann (2009), Wainwright (2009)), the Lasso estimator can be used as a screening method to reduce the set of the variables to $\widehat{S}$, a subset which contains the true support S with probability tending to 1. For example, by choosing the regularization parameter λ as (2.12) in Belloni and Chernozhukov (2013) and under certain assumptions, Belloni and Chernozhukov (2013) proved that $S \subseteq \widehat{S}$ and $|\widehat{S} \backslash S| \lesssim s$ with high probability (see Theorems 2 and 3 therein). When s is treated as a constant, the selected model will be of fixed dimension. Based on the selected model, we are able to directly apply our plug-in or batch-means estimator in Section 4 on $\widehat{S}$ to conduct inference for x_j^* for $j \in \widehat{S}$.

However, this approach has several limitations. First, the screening approach requires a strong "beta-min" assumption. In particular, this assumption requires that $\min_{j \in S} |x_j^*| > \max_{j \in S} |\widehat{x}_{\text{Lasso},j} - x_j^*|$, or $\min_{j \in S} |x_j^*| \gtrsim \sqrt{s(\log d)/n}$, for example, Belloni and Chernozhukov (2013), Bühlmann and Mandozzi (2014), Bühlmann and van de Geer (2011). Other screening methods (e.g., "Sure Independence Screening" (SIS) method (Fan and Lv (2008))) also require a similar beta-min condition. However, since we are interested in inference of the model parameters instead of the model selection, the "beta-min" condition should be avoidable. Second, the sparsity level s has to be treated as a constant to apply our theoretical results of either the plug-in or batch-means estimator. Furthermore, when using Lasso as a screening approach, it inevitably requires more than one pass of the data which does not fit our online setting.

5.1. *Debiasing approach.* To relax the strong conditions when using the Lasso as a screening approach, we propose a new approach for conducting inference for high-dimensional linear regression that only uses one pass of the data. Our approach is based on the following debiased Lasso estimator (Javanmard and Montanari (2014), van de Geer et al. (2014), Zhang and Zhang (2014)),

$$\widehat{x}_{\text{Lasso}}^d = \widehat{x}_{\text{Lasso}} + \frac{1}{n} \widehat{\Omega} D^T (b - D\widehat{x}_{\text{Lasso}}),$$

where $\widehat{\Omega}$ is an estimator of the inverse covariance matrix of the design $\Omega = A^{-1}$. To construct $\widehat{\Omega}$, van de Geer et al. (2014) adopts the nodewise Lasso approach (see also Meinshausen and Bühlmann (2006)), that is,

$$\widehat{\gamma}^j = \underset{\gamma^j \in \mathbb{R}^{d-1}}{\operatorname{argmin}} \frac{1}{2n} \|D_{\cdot,j} - D_{\cdot,-j}\gamma^j\|_2^2 + \lambda_j \|\gamma^j\|_1, \tag{22}$$

where $D_{\cdot,j}$ is the jth column of the design matrix D and $D_{\cdot,-j}$ is the design submatrix without the jth column. Further, one can estimate $\Omega_{j,j}$ by

$$\widehat{\tau}_j = \frac{1}{n}\big(D_{\cdot,j} - D_{\cdot,-j}\widehat{\gamma}^j\big)^T D_{\cdot,j}.$$

Given $\widehat{\gamma}^j$ and $\widehat{\tau}_j$, the matrix Ω is estimated by

$$\widehat{\Omega} = \widehat{T}\widehat{C}, \tag{23}$$

where $\widehat{T} := \operatorname{diag}(1/\widehat{\tau}_1, \ldots, 1/\widehat{\tau}_d)$ and

$$\widehat{C} := \begin{pmatrix} 1 & -\widehat{\gamma}_2^1 & \ldots & -\widehat{\gamma}_d^1 \\ -\widehat{\gamma}_1^2 & 1 & \ldots & -\widehat{\gamma}_d^2 \\ \vdots & \vdots & \ddots & \vdots \\ -\widehat{\gamma}_1^d & -\widehat{\gamma}_2^d & \ldots & 1 \end{pmatrix}.$$

Note that in the existing literature, $\widehat{x}_{\text{Lasso}}$ and $\widehat{\Omega}$ are obtained via deterministic convex optimization. Therefore, debiased Lasso approaches cannot be directly applied to the stochastic setting in this work. To address this issue, we propose to compute the estimators for both x^* and Ω using the Regularization Annealed epoch Dual AveRaging (RADAR) algorithm (Agarwal, Negahban and Wainwright (2012)), which is a variant of SGD. Similar to SGD, RADAR computes the stochastic gradient on one data point at each iteration. Please refer to Agarwal, Negahban and Wainwright (2012) for more details of the RADAR algorithm. The reason why we use RADAR instead of the vanilla SGD is because RADAR provides the optimal convergence rate in terms of the ℓ_1-norm. In particular, we apply RADAR to the following ℓ_1-regularized problem:

$$\min_{x \in \mathbb{R}^d} \mathbb{E}\big(b - a^T x\big)^2 + \lambda \|x\|_1 \tag{24}$$

and let $\widehat{x}_n$ be the solution output from RADAR with n iterations. Similarly, we again use stochastic optimization instead of deterministic optimization in the nodewise Lasso in (22), that is, applying RADAR to the following optimization problem for each dimension $1 \le j \le d$:

$$\widehat{\gamma}^j = \underset{\gamma^j \in \mathbb{R}^{d-1}}{\operatorname{argmin}} \mathbb{E}\|a_j - a_{-j}\gamma^j\|_2^2 + \lambda_j \|\gamma^j\|_1, \tag{25}$$

where a_j is the jth coordinate of the population design vector a and a_{-j} is the subvector of a without the jth coordinate. Given $\widehat{\gamma}^j$ from solving (25) via the iterative stochastic algorithm, the inverse covariance estimator $\widehat{\Omega}$ is constructed according to (23).

It is noteworthy that although the proposed $\widehat{\Omega}$ is of the same form as the estimator for A^{-1} in van de Geer et al. (2014), our $\widehat{\gamma}^j$ is different from the one in van de Geer et al. (2014). More precisely, our $\widehat{\gamma}^j$ is the output of a stochastic gradient-based algorithm, while van de Geer et al. (2014) obtained $\widehat{\gamma}^j$ from deterministic optimization in (22). With all these ingredients in place, we present the stochastic gradient based construction of the confidence interval for x_j^* for $j \in \{1, \ldots, d\}$ in Algorithm 1. The hypothesis test can also be performed once the

Algorithm 1 Stochastic optimization based confidence interval construction for high-dimensional sparse linear regression

Inputs:

Regularization parameter $\lambda \asymp \sqrt{\log d/n}$, and $\lambda_j \asymp \sqrt{\log d/n}$ for each dimension j, the noise level σ, confidence level $1-\alpha$.

for $t = 1$ to n **do**

Randomly sample the data (a_t, b_t) and update the design $D \leftarrow [D^T, a_t]^T$ and response $b \leftarrow [b^T, b_t]^T$.

Update x_t by running one iteration of RADAR on the optimization problem (24) using the stochastic gradient $(a_t^T x_{t-1} - b_t)a_t$,

for $j = 1$ to d **do**

Update γ_t^j by running one iteration of RADAR on the optimization problem (25) using the stochastic gradient $(a_{t,-j}^T \gamma_{t-1}^j - a_{t,j})a_{t,-j}$.

end for

end for

Let $\widehat{x}_n = x_n$ and $\widehat{\gamma}^j = \gamma_n^j$ for $j \in \{1, \ldots, d\}$ be the final outputs.

Construct the debiased estimator $\widehat{x}^d$ with $\widehat{\Omega}$ defined in (23).

$$\widehat{x}^d = \widehat{x}_n + \frac{1}{n}\widehat{\Omega} D^T (b - D\widehat{x}_n). \tag{26}$$

Outputs:

The $(1-\alpha)$ confidence interval for each $x_j^* : \widehat{x}_j^d \pm z_{\alpha/2}\sigma\sqrt{(\widehat{\Omega}\widehat{A}\widehat{\Omega})_{jj}/n}$, where $\widehat{A} = \frac{1}{n}D^T D$.

estimator of the asymptotic variance of $\widehat{x}_j^d$ is available (see Theorem 5.2). We note that the proposed method is computationally more efficient than the methods based on deterministic optimization. It only requires one pass of the data with the total per-iteration complexity $O(d^2)$ (note that the nodewise Lasso needs to solve d optimization problems) and is applicable to online data (in contrast to multiple passes of data with deterministic optimization used in existing methods). The details of the algorithm are provided in Algorithm 1.

To provide the theoretical justification for Algorithm 1 in terms of constructing valid confidence intervals, we make the following assumptions (which are similar to the assumptions made in van de Geer et al. (2014)).

ASSUMPTION 5.1. The covariate a is a sub-Gaussian random vector with variance proxy K^2. The population covariance A has bounded eigenvalues,

$$0 < \mu < \lambda_{\min}(A) < \lambda_{\max}(A) < L_F.$$

Denote the set of parameters by $\mathcal{B}(s) = \{x \in \mathbb{R}^d; \|x\|_0 \le s$ and $\|x\|_1$ is bounded by a constant$\}$. The true regression parameter $x^* \in \mathcal{B}(s)$ where $s = o(\sqrt{n}/\log d)$. Moreover, the inverse covariance Ω has sparse rows. In particular, define

$$s_j = \big|\{1 \le k \le d : k \ne j, \Omega_{j,k} \ne 0\}\big|.$$

We assume that $\max_j s_j \le Cs$ for some constant C.

Under Assumption 5.1, we first present an ℓ_1-bound result as a corollary of Proposition 1 in Agarwal, Negahban and Wainwright (2012),

PROPOSITION 5.1. *Under Assumption 5.1 and using the same algorithm parameters as Proposition 1 in Agarwal, Negahban and Wainwright (2012), there exists a constant c_0, such that $\widehat{x}_n$ in Algorithm 1 satisfies $\|\widehat{x}_n - x^*\|_1 \le c_0 s\sqrt{\frac{\log d}{n}}$ uniformly in $x^* \in \mathcal{B}(s)$ with high probability. Further, for each $j = \{1, \dots, d\}$, we have*

$$\|\widehat{\gamma}^j + \Omega_{j,j}^{-1}(\Omega_{j,-j})^T\|_1 \le c_0 s_j \sqrt{\frac{\log d}{n}}$$

holds with high probability.

The proof of Proposition 5.1 is provided in Section E.1 of the Supplementary Material. Let $\mathbb{P}_{x^*}$ be the distribution under the high-dimensional linear model $b_i = a_i^T x^* + \epsilon_i$. Given Proposition 5.1, we state the inference result in the next theorem. We note that although the statement of the following theorem is similar to Theorem 2.2 and Corollary 2.1 in van de Geer et al. (2014), the proof is more technically involved. The main challenge is that the existing analysis in van de Geer et al. (2014) starts from the KKT condition of the deterministic optimization for estimating Ω. However, we estimate Ω using the stochastic optimization, and thus the corresponding KKT condition no longer holds. Please refer to the proof in Section E.2 of the Supplementary Material for more detail.

THEOREM 5.2. *Under Assumption 5.1, for suitable choices of $\lambda \asymp \sqrt{\log d/n}$ and $\lambda_j \asymp \sqrt{\log d/n}$, we have for all $j \in \{1, \dots, d\}$ and all $z \in \mathbb{R}$,*

$$\sup_{x^* \in \mathcal{B}(s)} \left| \mathbb{P}_{x^*}\left(\frac{\sqrt{n}(\widehat{x}_j^d - x_j^*)}{\sigma\sqrt{(\widehat{\Omega}\widehat{A}\widehat{\Omega}^T)_{jj}}} \le z \right) - \Phi(z) \right| = o_p(1),$$

where $\widehat{x}^d$ is the debiased estimator defined in (25), $\widehat{\Omega}$ is defined in (23) and the sample covariance matrix $\widehat{A} = \frac{1}{n} D^T D$.

Theorem 5.2 shows that $\frac{1}{\sigma\sqrt{(\widehat{\Omega}\widehat{A}\widehat{\Omega}^T)_{jj}}}\sqrt{n}(\widehat{x}_j^d - x_j^*)$ converges in distribution to $N(0, 1)$ uniformly for any $x^* \in \mathcal{B}(s)$ and $j \in \{1, 2, \dots, d\}$, which verifies the correctness of the asymptotic pointwise confidence interval in the output of Algorithm 1 for x_j^*. Given the uniform convergence result in Theorem 5.2, we can construct p-values for each single component, and further conduct multiple testing based on componentwise p-values. We also note that a similar uniform convergence result has been established in Ning and Liu (2017) for a score test approach (see Remark 4.6). It is also interesting to investigate the stochastic optimization based score test as a future work.

6. Numerical simulations. In this section, we investigate the empirical performance of the plug-in estimator and batch-means estimator of the asymptotic covariance matrix. We consider both linear and logistic regression models, where $\{a_i, b_i\}$ are *i.i.d.* samples with $a_i \sim \mathcal{N}(0, \Sigma)$ and x^* is the true parameter vector of the model. For both models, we consider three different structures of the $d \times d$ covariance matric Σ:

- Identity: $\Sigma = I_d$;
- Toeplitz: $\Sigma_{i,j} = r^{|i-j|}$;
- Equi Corr: $\Sigma_{i,j} = r$ for all $i \ne j$, $\Sigma_{i,i} = 1$ for all i.

We report $r = 0.5$ for Toeplitz and $r = 0.2$ for equicorrelation (Equi Corr) covariance matrices in the main paper. The experimental results on other settings of r are relegated to the Section F of the Supplementary Material due to space limitations. The noise ε_n in linear regression is set to *i.i.d.* $N(0, \sigma^2)$ with $\sigma = 1$. The parameter α in the step size is chosen to

be 0.501 (slightly larger than 0.5). All the reported results are obtained by taking the average of 500 independent runs. We consider the finite sample behavior of the plug-in estimator and the batch-means estimator for the inference of each individual regression coefficient x_j, $j \in \{1, 2, \ldots, d\}$.

6.1. *Low-dimensional cases*. In each case, we consider the sample size $n = 10^5$ and the dimension $d = 5$, 20, 100, 200. For each model, the corresponding parameter x^* is a d-dimensional vector linearly spaced between 0 and 1. The thresholding scheme is not used for the plug-in estimator. In fact, we observe that the obtained A_n is always invertible and the results are stable without the thresholding. For the batch-means estimator (BM in short), we consider three different choices of the number of batches: $M = n^{0.2}$, $M = n^{0.25}$ and $M = n^{0.3}$. Note that $\alpha = 0.501$. As we suggested in Corollary 4.5, to achieve a better convergence rate, the number of batches M is chosen around the optimal value $n^{\frac{1-\alpha}{2}} \approx n^{0.25}$.

We set the nominal coverage probability $1 - q$ to 95%. The performance of an estimator is measured by the average coverage rate (Cov Rate) of the confidence intervals and the average length (Avg Len) of the intervals for each individual coefficient. For each setting, we also report the oracle length of the confidence interval with respect to the true covariance matrix $A^{-1}SA^{-1}$ and the corresponding coverage rate when using the same center as the BM.

For linear regression, the asymptotic covariance is $A^{-1}SA^{-1} = \sigma^2\Sigma^{-1} = \Sigma^{-1}$ and the oracle interval length for each coordinate j will be $\frac{2z_{q/2}(\Sigma^{-1})_{jj}}{\sqrt{n}}$. Table 1 shows the empirical performance of the plug-in and BM under linear models with three different design covariance matrices.

From Table 1, both the plug-in and BM achieve good performance. The plug-in gives better average coverage rate than BM: the average coverage rates in all different settings are nearly 95%. However, the average length of plug-in is usually larger than that of BM and the corresponding oracle interval length. On the other hand, BM achieves about 92% coverage rate when $M = n^{0.25}$ or $M = n^{0.3}$. We further consider the logistic regression. To provide an oracle interval length based on the true asymptotic covariance $A^{-1}SA^{-1} = A^{-1}$, we estimate A in (8) by its empirical version $\widehat{A}$ using one million fresh samples and the oracle interval length of each coordinate j is computed as $\frac{2z_{q/2}(\widehat{A}^{-1})_{jj}}{\sqrt{n}}$. We provide the result in Table 2 for different design covariance matrices. From Table 2, the plug-in still achieves nearly 95% average coverage rate. The BM achieves about 90% coverage rate and the average length is usually smaller than the oracle length. Moreover, as d becomes larger, the interval lengths for both estimators increase. Finally, the performance of BM is insensitive to the choice of the number of batches M: different M's lead to comparable coverage rates. There are two reasons for the undercoverage of BM. First, the obtained center could deviate from x^* that introduces the bias. Second, the BM has a slower convergence rate as compared to the plug-in (especially for the case of logistic regression). However, since the BM only uses the iterates from SGD, it is computationally more efficient than the plug-in estimator which requires the computation of the Hessian matrix $\tilde{A}_n$ and its inverse.

6.2. *High-dimensional cases*. In a high-dimensional setting, we consider the sample size $n = 100$, and the dimension $d = 500$. The active set $S_0 = \{1, 2, \ldots, s_0\}$, where the cardinality $s_0 = |S_0| = 3$ or 15. The non-zero regression coefficients $\{x_j^*\}_{j \in S_0}$ are from a fixed realization of s_0 *i.i.d.* uniform distribution $U[0, c]$ with $c = 2$.

First, we consider the average coverage rate and the average length of the intervals for individual coefficients corresponding to the variables in either S_0 or S_0^c where $S_0^c = \{1, \ldots, d\} \backslash S_0$. Again, we set the nominal coverage probability $1 - q$ to 95%. Our experimental setup

TABLE 1

Linear regression: *The average coverage rate and length of confidence intervals, for the nominal coverage probability* 95%. *The columns* (*BM*: n^c *for* $c = 0.2, 0.25$ *and* 0.3) *correspond to the batch-means estimator with* $M = n^c$ *number of batches. Cov Rate under "Oracle" refers to coverage rates when using the same center as BM but with oracle interval lengths. Standard errors are reported in the brackets*

			BM			
	d	Plug-in	$M = n^{0.2}$	$M = n^{0.25}$	$M = n^{0.3}$	Oracle
Identity Σ						
Cov Rate (%)	5	95.68 (0.87)	90.28 (0.46)	93.68 (0.79)	91.64 (0.79)	87.44
Avg Len ($\times 10^{-2}$)		1.49 (0.01)	1.39 (0.01)	1.47 (0.01)	1.43 (0.01)	1.24
Cov Rate (%)	20	94.99 (0.94)	91.30 (1.08)	93.92 (1.25)	92.95 (1.19)	88.24
Avg Len ($\times 10^{-2}$)		1.44 (0.01)	1.35 (0.01)	1.41 (0.01)	1.38 (0.01)	1.24
Cov Rate (%)	100	95.04 (1.01)	90.75 (1.36)	93.15 (1.12)	92.37 (1.10)	87.89
Avg Len ($\times 10^{-2}$)		1.41 (0.01)	1.32 (0.01)	1.35 (0.01)	1.35 (0.01)	1.24
Cov Rate (%)	200	94.75 (1.13)	90.49 (1.21)	92.97 (1.17)	91.97 (1.18)	88.12
Avg Len ($\times 10^{-2}$)		1.39 (0.01)	1.30 (0.01)	1.31 (0.01)	1.32 (0.01)	1.24
Toeplitz Σ						
Cov Rate (%)	5	95.24 (0.92)	91.16 (0.50)	94.28 (0.86)	93.04 (0.90)	88.31
Avg Len ($\times 10^{-2}$)		1.83 (0.10)	1.74 (0.10)	1.82 (0.11)	1.78 (0.12)	1.53
Cov Rate (%)	20	94.84 (0.97)	90.97 (1.08)	93.75 (0.93)	92.77 (0.81)	87.26
Avg Len ($\times 10^{-2}$)		1.81 (0.05)	1.71 (0.06)	1.78 (0.06)	1.76 (0.06)	1.58
Cov Rate (%)	100	95.01 (1.12)	90.36 (1.33)	91.83 (1.09)	91.52 (1.17)	89.11
Avg Len ($\times 10^{-2}$)		1.77 (0.02)	1.67 (0.03)	1.67 (0.03)	1.69 (0.02)	1.60
Cov Rate (%)	200	94.69 (1.33)	90.01 (1.41)	91.65 (1.36)	91.24 (1.41)	89.43
Avg Len ($\times 10^{-2}$)		1.74 (0.02)	1.62 (0.02)	1.62 (0.02)	1.62 (0.02)	1.60
Equi Corr Σ						
Cov Rate (%)	5	94.80 (0.88)	90.92 (1.09)	93.60 (0.92)	92.32 (0.68)	86.79
Avg Len ($\times 10^{-2}$)		1.60 (0.01)	1.46 (0.01)	1.55 (0.01)	1.52 (0.01)	1.31
Cov Rate (%)	20	95.10 (0.99)	91.15 (1.14)	93.66 (0.99)	92.78 (0.92)	88.04
Avg Len ($\times 10^{-2}$)		1.59 (0.01)	1.47 (0.01)	1.54 (0.01)	1.51 (0.01)	1.36
Cov Rate (%)	100	94.93 (1.06)	90.86 (1.26)	93.19 (1.15)	92.29 (1.10)	87.15
Avg Len ($\times 10^{-2}$)		1.56 (0.01)	1.47 (0.01)	1.52 (0.01)	1.50 (0.01)	1.38
Cov Rate (%)	200	94.49 (1.09)	90.57 (1.45)	92.45 (1.27)	91.91 (1.13)	87.22
Avg Len ($\times 10^{-2}$)		1.51 (0.01)	1.45 (0.01)	1.49 (0.01)	1.49 (0.01)	1.38

follows directly from van de Geer et al. (2014), and we provide the oracle length of the confidence intervals for comparison. For linear regression, the asymptotic covariance is $A^{-1}SA^{-1} = \sigma^2\Sigma^{-1} = \Sigma^{-1}$ and the oracle interval length for each coordinate j will be $\frac{2z_{q/2}(\Sigma^{-1})_{jj}}{\sqrt{n}}$. We provide the result in Table 3 for different design covariance matrices.

For high-dimensional linear regression, Algorithm 1 achieves good performance, especially in sparse settings ($s_0 = 3$). From Table 3, the average coverage rate is about 90%. For less sparse problems ($s_0 = 15$), our method still achieves about 88% average coverage rate for different design covariance matrices. The coverage rates of the obtained confidence intervals on active sets S_0 are slightly better than those on S_0^c. The average lengths on both sets are slightly smaller than the oracle lengths. The performance of the cases with identity design matrices are better than those with Toeplitz and equicorrelation design matrices (e.g., having smaller standard deviations). It is reasonable since it is easier to estimate the inverse covariance matrix Ω when it is an identity matrix.

In Table 4, we also provide the results using the deterministic optimization (instead of the stochastic RADAR) for constructing $\widehat{\Omega}$ (van de Geer et al. (2014)). Both methods achieve

TABLE 2

Logistic regression: *The average coverage rate and length of confidence intervals, for the nominal coverage probability* 95%. *The columns* (*BM*: n^c *for* $c = 0.2, 0.25$ *and* 0.3) *correspond to the batch-means estimator with* $M = n^c$ *number of batches. Cov Rate under "Oracle" refers to coverage rates when using the same center as BM but with oracle interval lengths. Standard errors are reported in the brackets*

			BM			
	d	Plug-in	$M = n^{0.2}$	$M = n^{0.25}$	$M = n^{0.3}$	Oracle
Identity Σ						
Cov Rate (%)	5	95.04 (1.13)	89.24 (1.55)	90.12 (1.70)	89.36 (1.97)	91.45
Avg Len ($\times 10^{-2}$)		3.24 (0.41)	3.01 (0.26)	2.94 (0.25)	2.87 (0.23)	3.09
Cov Rate (%)	20	95.00 (1.34)	89.35 (2.00)	90.22 (1.67)	89.74 (2.11)	90.37
Avg Len ($\times 10^{-2}$)		3.79 (0.27)	3.53 (0.25)	3.46 (0.23)	3.42 (0.22)	3.68
Cov Rate (%)	100	94.69 (1.06)	89.42 (1.66)	90.84 (1.68)	90.41 (2.01)	91.24
Avg Len ($\times 10^{-2}$)		5.21 (0.26)	4.97 (0.24)	4.87 (0.23)	4.80 (0.24)	5.06
Cov Rate (%)	200	94.47 (0.91)	89.01 (1.41)	90.47 (1.49)	90.36 (1.74)	92.08
Avg Len ($\times 10^{-2}$)		6.05 (0.29)	5.94 (0.26)	5.82 (0.27)	5.71 (0.25)	5.97
Toeplitz Σ						
Cov Rate (%)	5	94.96 (1.58)	88.96 (2.32)	90.56 (2.06)	90.12 (2.04)	92.41
Avg Len ($\times 10^{-2}$)		4.06 (0.34)	3.75 (0.28)	3.73 (0.27)	3.61 (0.25)	4.04
Cov Rate (%)	20	95.17 (1.23)	89.01 (1.93)	90.39 (1.88)	89.79 (1.81)	91.07
Avg Len ($\times 10^{-2}$)		5.74 (0.29)	5.57 (0.25)	5.22 (0.23)	4.95 (0.22)	5.59
Cov Rate (%)	100	94.91 (0.89)	89.91 (1.74)	90.83 (1.81)	90.54 (1.97)	91.47
Avg Len ($\times 10^{-2}$)		8.47 (0.37)	8.01 (0.28)	7.71 (0.26)	7.37 (0.25)	8.28
Cov Rate (%)	200	94.59 (1.04)	89.72 (1.81)	90.74 (1.93)	90.32 (2.02)	92.29
Avg Len ($\times 10^{-2}$)		9.81 (0.41)	9.24 (0.34)	8.95 (0.31)	8.78 (0.29)	9.84
Equi Corr Σ						
Cov Rate (%)	5	94.80 (1.66)	88.08 (1.46)	88.64 (1.73)	89.48 (1.51)	93.79
Avg Len ($\times 10^{-2}$)		3.43 (0.35)	3.28 (0.28)	3.24 (0.25)	3.20 (0.24)	3.38
Cov Rate (%)	20	94.54 (1.73)	89.27 (1.33)	90.64 (1.60)	90.31 (2.10)	92.50
Avg Len ($\times 10^{-2}$)		5.37 (0.31)	4.84 (0.26)	4.77 (0.24)	4.51 (0.21)	5.19
Cov Rate (%)	100	94.79 (1.08)	89.01 (1.70)	90.27 (1.76)	89.42 (2.01)	94.92
Avg Len ($\times 10^{-2}$)		10.24 (0.51)	10.17 (0.47)	9.75 (0.42)	9.24 (0.40)	10.89
Cov Rate (%)	200	94.24 (1.09)	89.13 (1.44)	90.01 (1.92)	89.23 (1.79)	92.40
Avg Len ($\times 10^{-2}$)		15.70 (0.62)	14.82 (0.57)	14.01 (0.55)	13.88 (0.52)	15.31

comparably reliable coverage rates. From Table 4, the average coverage rates are closer to the nominal levels, better than those in Table 3. The undercovering in Table 3 is due to the estimation error of the diagonals of $\hat{\Omega}\hat{A}\hat{\Omega}$ using stochastic optimization method. Based on the computational and storage requirements, a practitioner may decide to use a one-pass algorithm or a more accurate estimator under deterministic optimization.

7. Conclusions and future works. This paper presents two consistent estimators of the asymptotic variance of the average iterate from SGD, especially a computationally more efficient batch-means estimator that only uses iterates from SGD. With the proposed estimators, we are able to construct asymptotically exact confidence intervals and hypothesis tests.

We further discuss statistical inference based on SGD for high-dimensional linear regression. An extension to generalized linear models is an interesting problem for future work.

The seminal work by Toulis and Airoldi (2017) develops the averaged implicit SGD procedure and provides the characterization of the limiting distribution. It would be interesting to establish the consistency of the batch-means estimator based on iterates from implicit SGD.

TABLE 3

High-dimensional linear regression, the average coverage rate and length of confidence intervals, for the nominal coverage probability 95%. *Standard errors are reported in the brackets*

Measure	Identity Σ	Toeplitz Σ	Equi Corr Σ
$S_0 = \{1, 2, 3\}$			
Cov Rate S_0(%)	91.93 (3.13)	91.40 (2.39)	90.20 (1.38)
Avg Len S_0	0.387 (0.002)	0.401 (0.019)	0.360 (0.014)
Cov Rate S_0^c(%)	90.80 (1.79)	90.21 (1.98)	89.73 (1.96)
Avg Len S_0^c	0.386 (0.002)	0.417 (0.022)	0.384 (0.023)
Oracle Len	0.392	0.506	0.438
$S_0 = \{1, 2, \ldots, 15\}$			
Cov Rate S_0 (%)	90.48 (1.73)	89.84 (2.61)	89.45 (0.87)
Avg Len S_0	0.379 (0.002)	0.430 (0.024)	0.384 (0.020)
Cov Rate S_0^c (%)	88.43 (2.30)	86.79 (2.10)	87.12 (1.36)
Avg Len S_0^c	0.360 (0.003)	0.425 (0.024)	0.383 (0.022)
Oracle Len	0.392	0.506	0.438

It is also interesting to relax the current assumptions and consider SGD for more challenging optimization problems (e.g., nonconvex problems).

Acknowledgments. The authors are very grateful to the anonymous referees and Associate Editor for their detailed and constructive comments that considerably improved the quality of this paper. The authors would also like to thank John Duchi, Jessica Hwang, Lester Mackey, Yuekai Sun and Jonathan Taylor for early discussions on Markov process and the relationship to stochastic gradient.

The first author was supported by an Adobe Data Science Research Award, Alibaba Innovation Research Award and Bloomberg Data Science Research Grant.

The second author was supported by the ARO under MURI Award W911NF-11-1-0303. This is part of the collaboration between US DOD, UK MOD and UK Engineering and Physical Research Council (EPSRC) under the Multidisciplinary University Research Initiative.

The third author was supported by NUS Grant R-146-000-226-133.

TABLE 4

High-dimensional linear regression using nodewise lasso instead of RADAR for inference, the average coverage rate and length of confidence intervals, for the nominal coverage probability 95%. *Standard errors are reported in the brackets*

Measure	Identity Σ	Toeplitz Σ	Equi Corr Σ
$S_0 = \{1, 2, 3\}$			
Cov Rate S_0(%)	94.73 (1.42)	92.60 (1.95)	91.33 (1.99)
Avg Len S_0	0.393 (0.001)	0.472 (0.011)	0.431 (0.007)
Cov Rate S_0^c(%)	96.13 (1.44)	95.17 (1.71)	95.92 (2.04)
Avg Len S_0^c	0.386 (0.001)	0.481 (0.014)	0.429 (0.011)
Oracle Len	0.392	0.506	0.438
$S_0 = \{1, 2, \ldots, 15\}$			
Cov Rate S_0 (%)	91.80 (1.04)	91.07 (2.01)	90.33 (1.92)
Avg Len S_0	0.399 (0.001)	0.512 (0.015)	0.424 (0.008)
Cov Rate S_0^c (%)	95.75 (1.80)	93.17 (1.90)	94.11 (1.73)
Avg Len S_0^c	0.379 (0.001)	0.505 (0.016)	0.453 (0.008)
Oracle Len	0.392	0.506	0.438

SUPPLEMENTARY MATERIAL

Supplement to "Statistical inference for model parameters in stochastic gradient descent" (DOI: 10.1214/18-AOS1801SUPP; .pdf). We provide the proofs of all the theorectial results as well as additional simulation studies.

REFERENCES

AGARWAL, A., NEGAHBAN, S. and WAINWRIGHT, M. J. (2012). Stochastic optimization and sparse statistical recovery: Optimal algorithms for high dimensions. In *Proceedings of the Advances in Neural Information Processing Systems*.

BACH, F. and MOULINES, E. (2011). Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Proceedings of the Advances in Neural Information Processing Systems*.

BACH, F. and MOULINES, E. (2013). Non-strongly-convex smooth stochastic approximation with convergence rate $O(1/n)$. In *Proceedings of the Advances in Neural Information Processing Systems*.

BELLONI, A. and CHERNOZHUKOV, V. (2013). Least squares after model selection in high-dimensional sparse models. *Bernoulli* **19** 521–547. MR3037163 https://doi.org/10.3150/11-BEJ410

BÜHLMANN, P. and MANDOZZI, J. (2014). High-dimensional variable screening and bias in subsequent inference, with an empirical comparison. *Comput. Statist.* **29** 407–430. MR3261821 https://doi.org/10.1007/s00180-013-0436-3

BÜHLMANN, P. and VAN DE GEER, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications. Springer Series in Statistics*. Springer, Heidelberg. MR2807761 https://doi.org/10.1007/978-3-642-20192-9

BUJA, A., BERK, R., BROWN, L., GEORGE, E., TRASKIN, M., ZHANG, K. and ZHAO, L. (2013). A conspiracy of random X and model violation against classical inference in linear regression. Technical Report, Dept. Statistics, The Wharton School, Univ. Pennsylvania, Philadelphia, PA.

CHEN, X., LEE, J. D., TONG, X. T. and ZHANG, Y. (2020). Supplement to "Statistical inference for model parameters in stochastic gradient descent." https://doi.org/10.1214/18-AOS1801SUPP.

DAMERDJI, H. (1991). Strong consistency and other properties of the spectral variance estimator. *Manage. Sci.* **37** 1424–1440.

FABIAN, V. (1968). On asymptotic normality in stochastic approximation. *Ann. Math. Stat.* **39** 1327–1332. MR0231429 https://doi.org/10.1214/aoms/1177698258

FAN, J. and LV, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **70** 849–911. MR2530322 https://doi.org/10.1111/j.1467-9868.2008.00674.x

FISHMAN, G. S. (1996). *Monte Carlo: Concepts, Algorithms, and Applications. Springer Series in Operations Research*. Springer, New York. MR1392474 https://doi.org/10.1007/978-1-4757-2553-7

FLEGAL, J. M. and JONES, G. L. (2010). Batch means and spectral variance estimators in Markov chain Monte Carlo. *Ann. Statist.* **38** 1034–1070. MR2604704 https://doi.org/10.1214/09-AOS735

GEYER, C. (1992). Practical Markov chain Monte Carlo. *Statist. Sci.* **7** 473–483.

GHADIMI, S. and LAN, G. (2012). Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: A generic algorithmic framework. *SIAM J. Optim.* **22** 1469–1492. MR3023780 https://doi.org/10.1137/110848864

GLYNN, P. W. and IGLEHART, D. L. (1990). Simulation output analysis using standardized time series. *Math. Oper. Res.* **15** 1–16. MR1038232 https://doi.org/10.1287/moor.15.1.1

GLYNN, P. W. and WHITT, W. (1991). Estimating the asymptotic variance with batch means. *Oper. Res. Lett.* **10** 431–435. MR1141337 https://doi.org/10.1016/0167-6377(91)90019-L

JAVANMARD, A. and MONTANARI, A. (2014). Confidence intervals and hypothesis testing for high-dimensional regression. *J. Mach. Learn. Res.* **15** 2869–2909. MR3277152

JONES, G. L., HARAN, M., CAFFO, B. S. and NEATH, R. (2006). Fixed-width output analysis for Markov chain Monte Carlo. *J. Amer. Statist. Assoc.* **101** 1537–1547. MR2279478 https://doi.org/10.1198/016214506000000492

MEINSHAUSEN, N. and BÜHLMANN, P. (2006). High-dimensional graphs and variable selection with the lasso. *Ann. Statist.* **34** 1436–1462. MR2278363 https://doi.org/10.1214/009053606000000281

MEINSHAUSEN, N., MEIER, L. and BÜHLMANN, P. (2009). p-values for high-dimensional regression. *J. Amer. Statist. Assoc.* **104** 1671–1681. MR2750584 https://doi.org/10.1198/jasa.2009.tm08647

NEMIROVSKI, A., JUDITSKY, A., LAN, G. and SHAPIRO, A. (2008). Robust stochastic approximation approach to stochastic programming. *SIAM J. Optim.* **19** 1574–1609. MR2486041 https://doi.org/10.1137/070704277

NESTEROV, YU. and VIAL, J.-PH. (2008). Confidence level solutions for stochastic programming. *Automatica J. IFAC* **44** 1559–1568. MR2531843 https://doi.org/10.1016/j.automatica.2008.01.017

NING, Y. and LIU, H. (2017). A general theory of hypothesis tests and confidence regions for sparse high dimensional models. *Ann. Statist.* **45** 158–195. MR3611489 https://doi.org/10.1214/16-AOS1448

POLYAK, B. T. and JUDITSKY, A. B. (1992). Acceleration of stochastic approximation by averaging. *SIAM J. Control Optim.* **30** 838–855. MR1167814 https://doi.org/10.1137/0330046

RAKHLIN, A., SHAMIR, O. and SRIDHARAN, K. (2012). Making gradient descent optimal for strongly convex stochastic optimization. In *Proceedings of the International Conference on Machine Learning*.

ROBBINS, H. and MONRO, S. (1951). A stochastic approximation method. *Ann. Math. Stat.* **22** 400–407. MR0042668 https://doi.org/10.1214/aoms/1177729586

ROUX, N. L., SCHMIDT, M. and BACH, F. (2012). A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Proceedings of the Advances in Neural Information Processing Systems*.

RUPPERT, D. (1988). Efficient estimations from a slowly convergent Robbins–Monro process. Technical Report, Dept. Operations Research and Industrial Engineering, Cornell Univ.

SREBRO, N. and TEWARI, A. (2010). Stochastic optimization for machine learning. Tutorial at *International Conference on Machine Learning*.

SULLIVAN, T. J. (2015). *Introduction to Uncertainty Quantification*. *Texts in Applied Mathematics* **63**. Springer, Cham. MR3364576 https://doi.org/10.1007/978-3-319-23395-6

TOULIS, P. and AIROLDI, E. M. (2017). Asymptotic and finite-sample properties of estimators based on stochastic gradients. *Ann. Statist.* **45** 1694–1727. MR3670193 https://doi.org/10.1214/16-AOS1506

VAN DE GEER, S., BÜHLMANN, P., RITOV, Y. and DEZEURE, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *Ann. Statist.* **42** 1166–1202. MR3224285 https://doi.org/10.1214/14-AOS1221

WAINWRIGHT, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (Lasso). *IEEE Trans. Inform. Theory* **55** 2183–2202. MR2729873 https://doi.org/10.1109/TIT.2009.2016018

XIAO, L. (2010). Dual averaging methods for regularized stochastic learning and online optimization. *J. Mach. Learn. Res.* **11** 2543–2596. MR2738777

XIAO, L. and ZHANG, T. (2014). A proximal stochastic gradient method with progressive variance reduction. *SIAM J. Optim.* **24** 2057–2075. MR3285905 https://doi.org/10.1137/140961791

ZHANG, T. (2004). Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the International Conference on Machine Learning*.

ZHANG, C.-H. and ZHANG, S. S. (2014). Confidence intervals for low dimensional parameters in high dimensional linear models. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **76** 217–242. MR3153940 https://doi.org/10.1111/rssb.12026