THE IMPACT OF THE MINI-BATCH SIZE ON THE DYNAMICS OF SGD: VARIANCE AND BEYOND

Anonymous authors
Paper under double-blind review

ABSTRACT

We study mini-batch stochastic gradient descent (SGD) dynamics under linear regression and deep linear networks by focusing on the variance of the gradients only given the initial weights and mini-batch size, which is the first study of this nature. In the linear regression case, we show that in each iteration the norm of the gradient is a decreasing function of the mini-batch size b and thus the variance of the stochastic gradient estimator is a decreasing function of b. For deep neural networks with L_2 loss we show that the variance of the gradient is a polynomial in $1/b$. The results theoretically back the important intuition that smaller batch sizes yield larger variance of the stochastic gradients and lower loss function values which is a common believe among the researchers. The proof techniques exhibit a relationship between stochastic gradient estimators and initial weights, which is useful for further research on the dynamics of SGD. We empirically provide insights to our results on various datasets and commonly used deep network structures. We further discuss possible extensions of the approaches we build in studying the generalization ability of the deep learning models.

1 INTRODUCTION

Deep learning models have achieved great success in a variety of tasks including natural language processing, computer vision, and reinforcement learning (Goodfellow et al., 2016). Despite their practical success, there are only limited studies of the theoretical properties of deep learning; see survey papers (Sun, 2019; Fan et al., 2019) and references therein. The general problem underlying deep learning models is to optimize (minimize) a loss function, defined by the deviation of model predictions on data samples from the corresponding true labels. The prevailing method to train deep learning models is the mini-batch stochastic gradient descent algorithm and its variants (Bottou, 1998; Bottou et al., 2018). SGD updates model parameters by calculating a stochastic approximation of the full gradient of the loss function, based on a random selected subset of the training samples called a mini-batch.

It is well-accepted that selecting a large mini-batch size reduces the training time of deep learning models, as computation on large mini-batches can be better parallelized on processing units. For example, Goyal et al. (2017) scale ResNet-50 (He et al., 2016) from a mini-batch size of 256 images and training time of 29 hours, to a larger mini-batch size of 8,192 images. Their training achieves the same level of accuracy while reducing the training time to one hour. However, noted by many researchers, larger mini-batch sizes suffer from a worse generalization ability (LeCun et al., 2012; Keskar et al., 2017). Therefore, many efforts have been made to develop specialized training procedures that achieve good generalization using large mini-batch sizes (Hoffer et al., 2017; Goyal et al., 2017). Smaller batch sizes have the advantage of allegedly offering better generalization (at the expense of a higher training time).

The focus of this study is on the behavior of SGD subject to the conditions on the initial point. This is different from previous results which analyze SGD via stringing one-step recursions together. The dynamics of SGD are not comparable if we merely consider the one-step behavior, as the model parameters change iteration by iteration. Therefore, fixing the initial weights and the learning rate can give us a fair view of the impact of different mini-batch sizes on the dynamics of SGD. We hypothesize that, given the same initial point, smaller sizes lead to lower training loss and, unfortunately,

decrease stability of the algorithm on average. The latter follows from the fact that the smaller is the batch size, more stochasticity and volatility is introduced. After all, if the batch size equals to the number of samples, there is no stochasticity in the algorithm. To this end, we conjecture that the variance of the gradient in each iteration is a decreasing function of the mini-batch size. The conjecture is the focus of the work herein.

Variance correlates to many other important properties of SGD dynamics. For example, there is substantial work on variance reduction methods (Johnson & Zhang, 2013; Allen-Zhu & Hazan, 2016; Wang et al., 2013) which show great success on improving the convergence rate by controlling the variance of the stochastic gradients. Mini-batch size is also a key factor deciding the performance of SGD. Some research focuses on how to choose an optimal mini-batch size based on different criteria (Smith & Le, 2017; Gower et al., 2019). However, these works make strong assumptions on the loss function properties (strong or point or quasi convexity, or constant variance near stationary points) or about the formulation of the SGD algorithm (continuous time interpretation by means of differential equations). The statements are approximate in nature and thus not mathematical claims. The theoretical results regarding the relationship between the mini-batch size and the variance (and other performances, like loss and generalization ability) of the SGD algorithm applied to general machine learning models are still missing. The work herein partially addresses this gap by showing the impact of the mini-batch size on the variance of gradients in SGD. We further discuss possible extensions of the approaches we build in studying the generalization ability.

We are able to prove the hypothesis about variance in the convex linear regression case and to show significant progress in a deep linear neural network setting with samples based on a normal distribution. In this case we show that the variance is a polynomial in the reciprocal of the mini-batch size and that it is decreasing if the mini-batch size is larger than a threshold (further experiments reveal that this threshold can be as small as 2). The increased variance as the mini-batch size decreases should also intuitively imply convergence to lower training loss values and in turn better prediction and generalization ability (these relationships are yet to be confirmed analytically; but we provide empirical evidence to their validity).

The major contributions of this paper are as follows.

- For linear regression, we show that in each iteration the norm of any linear combination of sample-wise gradients is a decreasing function of the mini-batch size b (Theorem 1). As a special case, the variance of the stochastic gradient estimator and the full gradient at the iterate in step t are also decreasing functions of b at any iteration step t (Theorem 2). In addition, the proof provides a recursive relationship between the norm of gradients and the model parameters at each iteration (Lemma 2). This recursive relationship can be used to calculate any quantity related to the full/stochastic gradient or loss at any iteration with respect to the initial weights.

- For the deep linear neural network with L_2-loss and samples drawn from a normal distribution, we take two-layer linear network as an example and show that in each iteration step t the trace of any product of the stochastic gradient estimators and weight matrices is a polynomial in $1/b$ with coefficients a sum of products of the initial weights (Theorem 3). As a special case, the variance of the stochastic gradient estimator is a polynomial in $1/b$ without the constant term (Theorem 4) and therefore it is a decreasing function of b when b is large enough (Theorem 5). The results and proof techniques can be easily extended to general deep linear networks. As a comparison, other papers that study theoretical properties of two-layer networks either fix one layer of the network, or assume the over-parameterized property of the model and they study convergence, while our paper makes no such assumptions on the model capacity. The proof also reveals the structure of the coefficients of the polynomial, and thus serving as a tool for future work on proving other properties of the stochastic gradient estimators.

- The proofs are involved and require several key ideas. The main one is to show a more general result than it is necessary in order to carry out the induction. The induction is on time step t. The key idea is to show a much more general result that lets us carry out induction. New concepts and definitions are introduced in order to handle the more general case. Along the way we show a result of general interest establishing expectation of several rank one matrices sampled from a normal distribution intertwined with constant matrices.

- We verify the theoretical results on various datasets and provide further understanding. We further empirically show that the results extend to other widely used network structures and hold for all choices of the mini-batch sizes. We also empirically verify that, on average, in each iteration the loss function value and the generalization ability (measured by the gap between accuracy on the training and test sets) are all decreasing functions of the mini-batch size.

In conclusion, we study the dynamics of SGD under linear regression and a two-layer linear network setting by focusing on the decreasing property of the variance of stochastic gradient estimators with respect to the mini-batch size. The proof techniques can also be used to derive other properties of the SGD dynamics in regard to the mini-batch size and initial weights. To the best of authors' knowledge, the work is the first one to theoretically study the impact of the mini-batch size on the variance of the gradient subject to the conditions on the initial weights, under mild assumptions on the network and the loss function. We support our theoretical results by experiments. We further experiment on other state-of-the-art deep learning models and datasets to empirically show the validity of the conjectures about the impact of mini-batch size on average loss, average accuracy and the generalization ability of the model.

The rest of the manuscript is structured as follows. In Section 2 we review the literature while in Section 3 we present the theoretical results on how mini-batch sizes impact the variance of stochastic gradient estimators, under different models including linear regression and deep linear networks. Section 4 introduces (part of) the experiments that verify our theorems and provide further insights into the impact of the mini-batch sizes on SGD performance. We defer the complete experimental details to Appendix A and the proofs of the theorems and other technical details to to Appendix B.

2 Literature Review

Stochastic gradient descent type methods are broadly used in machine learning (Bottou, 1991; LeCun et al., 1998; Bottou et al., 2018). The performance of SGD highly relies on the choice of the mini-batch size. It has been widely observed that choosing a large mini-batch size to train deep neural networks appears to deteriorate generalization (LeCun et al., 2012). This phenomenon exists even if the models are trained without any budget or limits, until the loss function value ceases to improve (Keskar et al., 2017). One explanation for this phenomenon is that large mini-batch SGD produces "sharp" minima that generalize worse (Hochreiter & Schmidhuber, 1997; Keskar et al., 2017). Specialized training procedures to achieve good performance with large mini-batch sizes have also been proposed (Hoffer et al., 2017; Goyal et al., 2017).

It is well-known that SGD has a slow asymptotic rate of convergence due to its inherent variance (Johnson & Zhang, 2013). Variants of SGD that can reduce the variance of the stochastic gradient estimator, which yield faster convergence, have also been suggested. The use of the information of full gradients to provide variance control for stochastic gradients is addressed in Johnson & Zhang (2013); Roux et al. (2012); Shalev-Shwartz & Zhang (2013). The works in Lei et al. (2017); Li et al. (2014); Schmidt et al. (2017) further improve the efficiency and complexity of the algorithm by carefully controling the variance.

There is prior work focusing on studying the dynamics of SGD. Neelakantan et al. (2015) propose to add isotropic white noise to the full gradient to study the "structured" variance. The works in Li et al. (2017); Mandt et al. (2017); Jastrzebski et al. (2017) connect SGD with stochastic differential equations to explain the property of converged minima and generalization ability of the model. Smith & Le (2017) propose an "optimal" mini-batch size which maximizes the test set accuracy by a Bayesian approach. The Stochastic Gradient Langevin Dynamics (SGLD, a variant of SGD) algorithm for non-convex optimization is studied in Zhang et al. (2017); Mou et al. (2018).

In most of the prior work about the convergence of SGD, it is assumed that the variance of stochastic gradient estimators is upper-bounded by a linear function of the norm of the full gradient, e.g. Assumption 4.3 in Bottou et al. (2018). Gower et al. (2019) give more precise bounds of the variance under different sampling methods and Khaled & Richtárik (2020) extend them to smooth non-convex regime. These bounds are still dependent on the model parameters at the corresponding iteration. To the best of the authors' knowledge, there is no existing result which represents the variance of

stochastic gradient estimators only using the initial weights and the mini-batch size. This paper partially solves this problem.

3 ANALYSIS

Mini-batch SGD is a lighter-weight version of gradient descent. Suppose that we are given a loss function $L(w)$ where w is the collection (vector, matrix, or tensor) of all model parameters. At each iteration t, instead of computing the full gradient $\nabla_w L(w_t)$, SGD randomly samples a mini-batch set $\mathcal{B}_t$ that consists of $b = |\mathcal{B}_t|$ training instances and sets $w_{t+1} \leftarrow w_t - \alpha_t \nabla_w L_{\mathcal{B}_t}(w_t)$, where the positive scalar α_t is the learning rate (or step size) and $\nabla_w L_{\mathcal{B}_t}(w_t)$ denotes the stochastic gradient estimator based on mini-batch $\mathcal{B}_t$.

An important property of the stochastic gradient estimator $\nabla_w L_{\mathcal{B}_t}(w_t)$ is that it is an unbiased estimator, i.e. $\mathbb{E}\nabla_w L_{\mathcal{B}_t}(w_t) = \nabla_w L(w_t)$, where the expectation is taken over all possible choices of mini-batch $\mathcal{B}_t$. However, it is unclear what is the value of $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right) \triangleq \mathbb{E}\left\|\nabla_w L_{\mathcal{B}_t}(w_t)\right\|^2 - \left\|\mathbb{E}\nabla_w L_{\mathcal{B}_t}(w_t)\right\|^2$. Intuitively, we should have $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right) \propto \frac{n^2}{b}\mathsf{var}\left(\nabla_w L(w_t)\right)$, where n is the number of training samples and stochasticity on the right-hand side comes from mini-batch samples behind w_t (Smith & Le, 2017; Gower et al., 2019). However, even the quantities $\nabla_w L(w_t)$ and $\mathsf{var}\left(\nabla_w L(w_t)\right)$ are still challenging to compute as we do not have direct formulas of their precise values. Besides, as we choose different b's, their values are not comparable as we end up with different w_t's.

A plausible idea to address these issues is to represent $\mathbb{E}\nabla_w L_{\mathcal{B}_t}(w_t)$ and $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right)$ using the fixed and known quantities w_0, b, t, and α_t. In this way, we can further discover the properties, like decreasing with respect to b, of $\mathbb{E}\nabla_w L_{\mathcal{B}_t}(w_t)$ and $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right)$. The biggest challenge is how to connect the quantities in iteration t with those of iteration 0. This is similar to discovering the properties of a stochastic differential equation at time t given only the dynamics of the stochastic differential equation and the initial point.

In this section, we address these questions under two settings: linear regression and a deep linear network. In Section 3.1 with a linear regression setting, we provide explicit formulas for calculating any norm of the linear combination of sample-wise gradients. We therefore show that the $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right)$ is a decreasing function of the mini-batch size b. In Section 3.2 with a deep linear network setting and samples drawn from a normal distribution, we show that any trace of the product of weight matrices and stochastic gradient estimators is a polynomial in $1/b$ with finite degree. We further prove that $\mathsf{var}\left(\nabla_w L_{\mathcal{B}_t}(w_t)\right)$ is a decreasing function of the mini-batch size $b > b_0$ for some constant b_0.

For a random matrix M, we define $\mathsf{var}\left(M\right) \triangleq \mathbb{E}\left\|\mathrm{vec}(M)\right\|^2 - \left\|\mathbb{E}\mathrm{vec}(M)\right\|^2$ where $\mathrm{vec}(M)$ denotes the vectorization of matrix M. We denote $[m:n] \triangleq \{m, m+1, \ldots, n\}$ if $m \leqslant n$, and $\varnothing$ otherwise. We use $[n] \triangleq [1:n]$ as an abbreviation. For clarity, we use the superscript b to distinguish the variables with different choices of the mini-batch size b. In each iteration t, we use $\mathcal{B}_t^b$ to denote the batch of samples (or sample indices) to calculate the stochastic gradient. We denote by $\mathcal{F}_t^b$ the filtration of information before calculating the stochastic gradient in the t-th iteration, i.e. $\mathcal{F}_t^b \triangleq \left\{w_0, \mathcal{B}_0^b, \ldots, \mathcal{B}_{t-1}^b\right\}$.

3.1 LINEAR REGRESSION

In this subsection, we discuss the dynamics of SGD applied in linear regression. Given data points $(x_1, y_1), \cdots, (x_n, y_n)$, where $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, we define the loss function to be $L(w) = \frac{1}{n}\sum_{i=1}^n L_i(w) = \frac{1}{n}\sum_{i=1}^n \frac{1}{2}\left(w^T x_i - y_i\right)^2$, where $w \in \mathbb{R}^p$ are the model parameters. We consider minimizing $L(w)$ by mini-batch SGD. Note that the bias term in the general linear regression models is omitted, however, adding the bias term does not change the result of this section. Formally, we first choose a mini-batch size b and initial weights w_0. In each iteration t, we sample $\mathcal{B}_t^b$, a subset of $[n]$ with cardinality b, and update the parameters by $w_{t+1}^b = w_t^b - \alpha_t g_t^b$, where $g_t^b = \frac{1}{b}\sum_{i\in\mathcal{B}_t^b} \nabla L_i\left(w_t^b\right)$.

We first show the relationship between the variance of stochastic gradient g_t^b and the full gradient $\nabla L\left(w_t^b\right)$ and sample-wise gradient $\nabla L_i\left(w_t^b\right), i \in [n]$, derived by considering all possible choices

of the mini-batch $\mathcal{B}_t^b$. Readers should note that Lemma 1 actually holds for all models with L_2-loss, not merely linear regression (since in the proof we do not need to know the explicit form of $L_i(w)$).

Lemma 1. *Let* $c_b \triangleq \frac{n-b}{b(n-1)} \geqslant 0$. *For any matrix* $A \in \mathbb{R}^{p\times p}$ *we have* $\mathsf{var}\left(Ag_t^b|\mathcal{F}_t^b\right) = \mathbb{E}\left[\left\|Ag_t^b\right\|^2\middle|\mathcal{F}_t^b\right] - \left\|A\nabla L\left(w_t^b\right)\right\|^2 = c_b\left(\frac{1}{n}\sum_{i=1}^n\left\|A\nabla L_i\left(w_t^b\right)\right\|^2 - \left\|A\nabla L\left(w_t^b\right)\right\|^2\right)$.

Lemma 1 provides a bridge to connect the norm and variance of g_t^b with sample-wise gradients $\nabla L_i\left(w_t^b\right), i \in [n]$. Therefore, if we can further discover the properties of $\nabla L_i\left(w_t^b\right), i \in [n]$, we are able to calculate the variance of g_t^b. Lemma 2 addresses this problem by showing the relationship between any linear combination of $\nabla L_i\left(w_t^b\right)$'s and $\nabla L_i\left(w_{t-1}^b\right)$'s.

Lemma 2. *For any set of square matrices* $\{A_1, \cdots, A_n\} \in \mathbb{R}^{p\times p}$, *if we denote* $A = \sum_{i=1}^n A_i x_i x_i^T$, *then we have* $\mathbb{E}\left[\left\|\sum_{i=1}^n A_i\nabla L_i\left(w_{t+1}^b\right)\right\|^2\middle|\mathcal{F}_0\right] = \mathbb{E}\left[\left\|\sum_{i=1}^n B_i\nabla L_i\left(w_t^b\right)\right\|^2\middle|\mathcal{F}_0\right] + \frac{\alpha_t^2 c_b}{n^2}\sum_{k=1}^n\sum_{l=1}^n\mathbb{E}\left[\left\|\sum_{i=1}^n B_i^{kl}\nabla L_i\left(w_t^b\right)\right\|^2\middle|\mathcal{F}_0\right]$, *where* $B_i = A_i - \frac{\alpha_t}{n}A$; $B_i^{kl} = A$ *if* $i = k, i \neq l$, $B_i^{kl} = A$ *if* $i = l, i \neq k$, *and* B_i^{kl} *equals the zero matrix, otherwise.*

Lemma 2 provides the tool to reduce the iteration t by one. Therefore, we can easily use it to recursively calculate the norm of any linear combinations of the sample-wise gradients, for all iterations t. Combining the fact that c_b is a decreasing function of b, we are able to show Theorem 1.

Theorem 1. *For any* $t \in \mathbb{N}$ *and any matrices* $A_i \in \mathbb{R}^{p\times p}, i \in [n]$, $\mathbb{E}\left[\left\|\sum_{i=1}^n A_i\nabla L_i\left(w_t^b\right)\right\|^2\middle|\mathcal{F}_0\right]$ *is a decreasing function of* b *for* $b \in [n]$.

Theorem 1 states that the norm of any linear combinations of the sample-wise gradients is a decreasing function of b. Combining Lemma 1 which connects the variance of g_t^b with the linear combination of $\nabla L_i\left(w_t^b\right)$'s, and the fact that $\nabla L\left(w_t^b\right) = \frac{1}{n}\sum_{i=1}^n\nabla L_i\left(w_t^b\right)$, we have Theorem 2.

Theorem 2. *Fixing initial weights* w_0, *both* $\mathsf{var}\left(Bg_t^b\left|\mathcal{F}_0\right.\right)$ *and* $\mathsf{var}\left(B\nabla L\left(w_t^b\right)\left|\mathcal{F}_0\right.\right)$ *are decreasing functions of mini-batch size* b *for all* $b \in [n]$, $t \in \mathbb{N}$, *and all square matrices* $B \in \mathbb{R}^{p\times p}$.

As a special case, Corollary 1 guarantees that the variance of the stochastic gradient estimator is a decreasing function of b.

Corollary 1. *Fixing initial weights* w_0, *both* $\mathsf{var}\left(g_t^b\left|\mathcal{F}_0\right.\right)$ *and* $\mathsf{var}\left(\nabla L\left(w_t^b\right)\left|\mathcal{F}_0\right.\right)$ *are decreasing functions of mini-batch size* b *for all* $b \in [n]$ *and* $t \in \mathbb{N}$.

In conclusion, we provide a framework for calculating the explicit value of variance of the stochastic gradient estimators and the norm of any linear combination of sample-wise gradients. We further show that the variance of both the full gradient and the stochastic gradient estimator are a decreasing function of the mini-batch size b. Readers should note that the framework here is not limited to showing the decreasing property of the variance, but can also be used in many other circumstance. For example, we can use Lemma 2 to induct on t and easily show that $\mathbb{E}\left[\left\|\sum_{i=1}^n A_i\nabla L_i\left(w_t^b\right)\right\|^2\middle|\mathcal{F}_0\right]$ is a polynomial of $\frac{1}{b}$ with degree at most t and estimate the coefficients therein.

3.2 Deep Linear Networks with Online Setting

In this section, we study the dynamics of SGD on deep linear networks. We take the two-layer linear network as an example while the results and proofs can be easily extended to deep linear network with any depth (see Appendix B.3 for more details). We consider the population loss $\mathcal{L}(w) = \mathbb{E}_{x\sim\mathcal{N}(0,I_p)}\left[\frac{1}{2}\left\|W_2W_1x - W_2^*W_1^*x\right\|^2\right]$ under the teacher-student learning framework (Hinton et al., 2015) with $w = (W_1, W_2)$ a tuple of two matrices. Here $W_1 \in \mathbb{R}^{p_1\times p}$ and $W_2 \in \mathbb{R}^{p_2\times p_1}$ are parameter matrices of the student network and W_1^* and W_2^* are the fixed ground-truth parameters of the teacher network. We use online SGD to minimize the population loss $\mathcal{L}(w)$. Formally, we first choose a mini-batch size b and initial weight matrices $\{W_{0,1}, W_{0,2}\}$. In each iteration t, we draw b independent and identically distributed samples $x_{t,i}, i \in [b]$ from $\mathcal{N}(0, I_p)$ to form the mini-batch $\mathcal{B}_t^b$ and update the weight matrices by $W_{t+1,1}^b = W_{t,1}^b - \alpha_t g_{t,1}^b$ and

$W^b_{t+1,2} = W^b_{t,2} - \alpha_t g^b_{t,2}$, where

$$g^b_{t,1} = \frac{1}{b}\sum_{i=1}^{b} \nabla_{W^b_{t,1}} \left(\frac{1}{2}\left\|W^b_{t,2}W^b_{t,1}x_{t,i} - W_2^*W_1^*x_{t,i}\right\|^2\right) = \frac{1}{b}\sum_{i=1}^{b} {W^b_{t,2}}^T \left(W^b_{t,2}W^b_{t,1} - W_2^*W_1^*\right)x_{t,i}x_{t,i}^T, \quad (1)$$

$$g^b_{t,2} = \frac{1}{b}\sum_{i=1}^{b} \nabla_{W^b_{t,2}} \left(\frac{1}{2}\left\|W^b_{t,2}W^b_{t,1}x_{t,i} - W_2^*W_1^*x_{t,i}\right\|^2\right) = \frac{1}{b}\sum_{i=1}^{b} \left(W^b_{t,2}W^b_{t,1} - W_2^*W_1^*\right)x_{t,i}x_{t,i}^T{W^b_{t,1}}^T. \quad (2)$$

The derivation follows from the formulas in Petersen & Pedersen (2012). In the following, we use $\mathcal{W}^b_t = W^b_{t,2}W^b_{t,1} - W_2^*W_1^*$ to denote the gap between the product of model weights and ground-truth weights.

For ease of developing our proofs, we first introduce the definition of a *multiplicative term* in Definition 1. Intuitively, a multiplicative term is a matrix which equals to the product of its parameter matrices and constant matrices (and their transpose). The degree of a matrix A in a multiplicative term M is the number of appearance of A and A^T in M. The degree of M is exactly the number of appearances of all weight matrices in M.

Definition 1. For any set of matrices $\mathcal{S}$, we denote $\bar{\mathcal{S}} = \mathcal{S} \cup \{M^T : M \in \mathcal{S}\}$. Given a set of parameter matrices $\mathcal{X} = \{X_1, X_2, \cdots, X_{n_v}\}$ and constant matrices $\mathcal{C} = \{C_1, C_2, \cdots, C_{n_c}\}$, we say that a matrix M is *a multiplicative term of parameter matrices $\mathcal{X}$ and constant matrices $\mathcal{C}$* if it can be written in the form of $M = M(\mathcal{X}, \mathcal{C}) = \prod_{i=1}^{k} A_i$, where $A_i \in \bar{\mathcal{X}} \cup \bar{\mathcal{C}}$. We write $\deg(X_j; M) = \sum_{i=1}^{k} \left(\mathbb{1}\left\{X_j = A_i\right\} + \mathbb{1}\left\{X_j = A_i^T\right\}\right), j \in [n_v]$ as the degree of parameter matrix X_j in M, $\deg(C_j; M) = \sum_{i=1}^{k} \left(\mathbb{1}\left\{C_j = A_i\right\} + \mathbb{1}\left\{C_j = A_i^T\right\}\right), j \in [n_c]$ as the degree of constant matrix C_j in M, and $\deg(M) = \sum_{i=1}^{k} \mathbb{1}\left\{A_i \in \bar{\mathcal{X}}\right\} = \sum_{j=1}^{n_v} \deg(X_j; M)$ as the total degree of the parameter matrices of M.

As pointed out in the Section 1, the difficulty of studying the dynamics of SGD is how to connect the quantities in iteration t with fixed variables, like initial weights $W_{0,1}, W_{0,2}$ and mini-batch size b. We overcome this challenge by carefully calculating the relationship between $g^b_{t+1,i}$ and $g^b_{t,i}, i = 1, 2$ so that we can reduce the iteration t step by step. With the help of Lemmas 8 and 9 in Appendix B.2, we can represent $g^b_{t+1,i}, i = 1, 2$ using multiplicative terms of $g^b_{t,i}, i = 1, 2$ and some other constant matrices. Theorem 3 precisely gives the representation in the form of a polynomial of $\frac{1}{b}$ and the coefficients as the sum of multiplicative terms of parameter matrices $\left\{W^b_{0,1}, W^b_{0,2}\right\}$ and constant matrices $\{W_1^*, W_2^*\}$.

Theorem 3. *Given $t \geqslant 0$, for any multiplicative terms $M_i, i \in [0:m]$ of parameter matrices $\left\{g^b_{t,1}, g^b_{t,2}\right\}$ and constant matrices $\left\{W^b_{t,1}, W^b_{t,2}, W_1^*, W_2^*\right\}$ with degree d_i, respectively, we denote $M = \prod_{i=1}^{m} \mathrm{tr}\left(M_i\right) M_0$, $d = \sum_{i=0}^{m} d_i$ and $d' = \sum_{i=0}^{m} \left(\deg\left(W^b_{t,1}; M_i\right) + \deg(W^b_{t,2}; M_i)\right)$. There exists a set of multiplicative terms $\left\{M^k_{ij}, i \in [m_k], j \in [0:m_{ki}], k \in [0:q]\right\}$ of parameter matrices $\left\{W^b_{0,1}, W^b_{0,2}\right\}$ and constant matrices $\{W_1^*, W_2^*\}$ such that $\mathbb{E}\left[M|\mathcal{F}_0\right] = N_0 + N_1\frac{1}{b} + \cdots + N_q\frac{1}{b^q}$, where $N_k = \sum_{i=1}^{m_k}\prod_{j=1}^{m_{ki}} \mathrm{tr}\left(M^k_{ij}\right) M^k_{i0}, k \in [0:q]$. Here m_k, m_{ki} and $q \leqslant \frac{1}{2}(3^{t+1}-1)d + \frac{1}{2}(3^t - 1)d'$ are constants independent of b, and $\sum_{j=0}^{m_{ki}} \deg\left(M^k_{ij}\right) \leqslant 3^t(3d + d')$.*

As a special case of Theorem 3, Theorem 4 shows that the variance of the stochastic gradient estimators is also a polynomial of $\frac{1}{b}$ but with no constant term. This backs the important intuition that the variance is approximately inversely proportional to the mini-batch size b. Besides, note that if we consider $b \to \infty$, intuitively we should have $\mathsf{var}\left(g^b_{t,i}|\mathcal{F}_0\right) \to 0, i = 1, 2$. This observation aligns with the statement of Theorem 4.

Theorem 4. *Given $t \geqslant 0$, value $\mathsf{var}\left(g^b_{t,i}|\mathcal{F}_0\right), i = 1, 2$ can be written as a polynomial of $\frac{1}{b}$ with degree at most $2 \cdot 3^t$ with no constant term. Formally, we have $\mathsf{var}\left(g^b_{t,i}|\mathcal{F}_0\right) = \beta_1\frac{1}{b} + \cdots + \beta_r\frac{1}{b^r}$, where $r \leqslant 2 \cdot 3^{t+1}$ and each β_i is a constant independent of b.*

One should note that the polynomial representation of $\mathsf{var}\left(g^b_{t,i}|\mathcal{F}_0\right), i = 1, 2$ does not have the constant term. Therefore, to show the that the variance is a decreasing function of b, we only need to show that the leading coefficient β_1 is non-negative. This is guaranteed by the fact that variance is always non-negative. We therefore have Theorem 5.

Theorem 5. *Given $t \in \mathbb{N}$, there exists a constant b_0 such that for all $b \geqslant b_0$ function* $\mathsf{var}\left(g_{t,i}^b \middle| \mathcal{F}_0\right), i = 1, 2$ *is a decreasing function of b.*

The constant b_0 is the largest root of the equation $\beta_1 b^{r-1} + \beta_2 b^{r-2} + \cdots + \beta_r = 0$. See the proof of Theorem 5 in Appendix B.2 for more details. Although we cannot calculate the precise value of b_0, we verify that b_0 is smaller than 1 in many experiments. From the proofs we conclude that the scale of each β_i is of the order $\mathcal{O}(\|M\|)$, where M is a multiplicative term of parameter matrices $\{W_{0,1}, W_{0,2}, W_1^*, W_2^*\}$ and constant matrix $\varnothing$ with degree $2 \cdot 3^{t+1}$.

Unlike the linear regression setting where we can iteratively calculate the variance by Lemma 2, the closed form expressions for the variance of the stochastic gradients in the deep linear network setting are much harder to calculate. However, we are able to iteratively deducing t one by one and provide a polynomial representation for any multiplicative terms of parameter matrices $\left\{g_{t,i}^b, W_{t,i}^b, i = 1, 2\right\}$ and constant matrices $\{W_1^*, W_2^*\}$ using only the initial weights $W_{0,1}, W_{0,2}$ and the mini-batch size b. As we further study the polynomial representation of $\mathsf{var}\left(g_{t,i}^b \middle| \mathcal{F}_0\right), i = 1, 2$, we are also able to show the decreasing property of the variance of stochastic gradient estimators with respect to b.

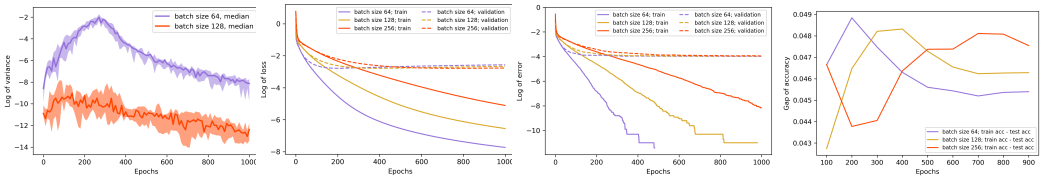
4 Experiments

In this section, we present numerical results to support the theorems in Section 3 and provide further insights into the impact of the mini-batch size on the dynamics of SGD. The experiments are conducted on four datasets and models that are relatively small due to the computational cost of using large models and datasets. We only report the results on the MNIST dataset here due to the limited space. A complete empirical study is deferred in Appendix A.

For all experiments, we perform mini-batch SGD multiple times starting from the same initial weights and following the same choice of the learning rates and other hyper-parameters, if applicable. This enables us to calculate the variance of the gradient estimators and other statistics in each iteration, where the randomness comes only from different samples of SGD.

4.1 Results on MNIST Dataset

The MNIST dataset is to recognize digits in handwritten images of digits. We use all 60,000 training samples and 10,000 validation samples of MNIST. We build a three-layer fully connected neural network with 1024, 512 and 10 neurons in each layer. For the two hidden layers, we use the ReLU activation function. The last layer is the softmax layer which gives the prediction probabilities for the 10 digits. We use mini-batch SGD to optimize the cross-entropy loss of the model. The model deviates from our analytical setting since it has non-linear activations, it has the cross-entropy loss function (instead of L_2), and empirical loss (as opposed to population). MNIST is selected due to its fast training and popularity in deep learning experiments. The goal is to verify the results in this different setting and to back up our hypotheses.



(a) Different initial weights (b) Log of loss (c) Log of error (d) Gap of accuracy (zoomed-in)

Figure 1: Experimental results for the MNIST dataset. (a) The median, min, and max of the log of variance of the stochastic gradient estimators for two different mini-batch sizes (distinguished by colors) and five different initial weights. The solid lines show the median of all five initial weights while the highlighted regions show the min and max of the log of variance. (b) The log of the training and validation loss vs epochs. (c) The log of training and validation error vs epochs. Here error is defined as one minus predicting accuracy. The plot does not show the epochs if error equals to zero. (d) The gap of accuracy on training and test sets vs epochs starting from epoch 100.

As shown in Figure 1(a), we run SGD with two batch sizes 64 and 128 on five different initial weights with 50 runs for each initial point. This plot shows that, even the smallest value of the variance among the five different initial weights with a mini-batch size of 64, is still larger than the largest variance of mini-batch size 128. We observe that the sensitivity to the initial weights is not large. This plot also empirically verifies our conjecture in the introduction that the variance of the stochastic gradient estimators is a decreasing function of the mini-batch size, for all iterations of SGD in a general deep learning model.

In addition, we also conjecture that there exists the decreasing property for the expected loss, error and the generalization ability with respect to the mini-batch size. Figure 1(b) shows that the expected loss (again, randomness comes from different runs of SGD through the different mini-batches with the same initial weights and learning rates) on the training set is a decreasing function of b. However, this decreasing property does not hold on the validation set when the loss tends to be stable or increasing, in other words, the model starts to be over-fitting. We hypothesize that this is because the learned weights start to bounce around a local minimum when the model is over-fitting. As the larger mini-batch size brings smaller variance, the weights are closer to the local minimum found by SGD, and therefore yield a smaller loss function value. Figure 1(c) shows that both the expected error on training and validation sets are decreasing functions of b.

Figure 1(d) exhibits a relationship between the model's generalization ability and the mini-batch size. As suggested by Simard et al. (2013), we build a test set by distorting the 10,000 images of the validation set. The prediction accuracy is obtained on both training and test sets and we calculate the gap between these two accuracies every 100 epochs. We use this gap to measure the model generalization ability (the smaller the better). Figure 1(d) shows that the gap is an increasing function of b starting at epoch 500, which partially aligns with our conjecture regarding the relationship between the generalization ability and the mini-batch size. We test this on multiple choices of the hyper-parameters which control the degree of distortion in the test set and this pattern remains clear.

5 Summary and Future Work

We examine the impact of the mini-batch size on the dynamics of SGD. Our focus is on the variance of stochastic gradient estimators. For linear regression and a two-layer linear network, we are able to theoretically prove that the variance conjecture holds. We further experiment on multiple models and datasets to verify our claims and their applicability to practical settings. Besides, we also empirically address the conjectures about the expected loss and the generalization ability.

A challenging research direction is to theoretically investigate the impact of the mini-batch size on the generalization ability. There are existing works studying the relationship between the variance of the stochastic gradients and the generalization ability (Gorbunov et al., 2020; Meng et al., 2016). Together with the tools developed herein, it would be possible to bridge the mini-batch size with the generalization ability of a neural network. We can further choose an optimal mini-batch size which minimizes the generalization ability by solving the polynomial equation if we have more precise estimations of the coefficients.

Another appealing direction is using our variance estimations to develop better variance reduction methods. As a results, the upper-bound of the variance decides the convergent rate of these algorithms. Researchers usually assume a much larger upper-bound at each iteration, like a linear function of the norm of the full gradient. With the help of our techniques, we should calculate the variance more precisely and further improve the algorithms.

Further interesting work is to extend our techniques to more complicated and sophisticated networks. Although the underlying model of this paper corresponds to deep linear network networks, we are able to show a deeper relationship between the variance and the mini-batch size, the polynomial in $1/b$, while the common knowledge is simply that the variance is proportional to $1/b$. The extension to other optimization algorithms, like Adam and Gradient Boosting Machines, are also very attractive. We hope our theoretical framework can serve as a tool for future research of this kind.

REFERENCES

Mohan S Acharya, Asfia Armaan, and Aneeta S Antony. A comparison of regression models for prediction of graduate admissions. In *2019 International Conference on Computational Intelligence in Data Science*, pp. 1–5, 2019.

Zeyuan Allen-Zhu and Elad Hazan. Variance reduction for faster non-convex optimization. In *International conference on machine learning*, pp. 699–707, 2016.

Léon Bottou. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, 91(8): 12, 1991.

Léon Bottou. Online learning and stochastic approximations. *On-line Learning in Neural Networks*, 17(9):142, 1998.

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

Jianqing Fan, Cong Ma, and Yiqiao Zhong. A selective overview of deep learning. *arXiv preprint arXiv:1904.05526*, 2019.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. A unified theory of sgd: Variance reduction, sampling, quantization and coordinate descent. In *International Conference on Artificial Intelligence and Statistics*, pp. 680–690, 2020.

Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. In *International Conference on Machine Learning*, pp. 5200–5209, 2019.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training Imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

Elad Hoffer, Itay Hubara, and Daniel Soudry. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In *Advances in Neural Information Processing Systems*, pp. 1731–1741, 2017.

Stanislaw Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in SGD. *arXiv preprint arXiv:1711.04623*, 2017.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pp. 315–323, 2013.

Nitish Shirish Keskar, Jorge Nocedal, Ping Tak Peter Tang, Dheevatsa Mudigere, and Mikhail Smelyanskiy. On large-batch training for deep learning: Generalization gap and sharp minima. In *5th International Conference on Learning Representations, 2017*, 2017.

Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the Institute of Electrical and Electronics Engineers*, 86 (11):2278–2324, 1998.

Yann LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pp. 9–48. Springer, 2012.

Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex finite-sum optimization via SCSG methods. In *Advances in Neural Information Processing Systems*, pp. 2348–2358, 2017.

Mu Li, Tong Zhang, Yuqiang Chen, and Alexander J Smola. Efficient mini-batch training for stochastic optimization. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 661–670, 2014.

Qianxiao Li, Cheng Tai, and Weinan E. Stochastic modified equations and adaptive stochastic gradient algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 2101–2110. PMLR, 2017.

Jan R Magnus. *The moments of products of quadratic forms in normal variables*. Instituut voor Actuariaat en Econometrie, 1978.

Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian inference. *The Journal of Machine Learning Research*, 18(1):4873–4907, 2017.

Qi Meng, Yue Wang, Wei Chen, Taifeng Wang, Zhi-Ming Ma, and Tie-Yan Liu. Generalization error bounds for optimization algorithms via stability. *arXiv preprint arXiv:1609.08397*, 2016.

Wenlong Mou, Liwei Wang, Xiyu Zhai, and Kai Zheng. Generalization bounds of SGLD for non-convex learning: Two theoretical viewpoints. In *Conference On Learning Theory*, pp. 605–638, 2018.

Arvind Neelakantan, Luke Vilnis, Quoc V Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach, and James Martens. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*, 2015.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

K. B. Petersen and M. S. Pedersen. The matrix cookbook, 2012. Version 20121115.

Nicolas L Roux, Mark Schmidt, and Francis R Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems*, pp. 2663–2671, 2012.

Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.

P. Y. Simard, D. Steinkraus, and J. C. Platt. Best practices for convolutional neural networks applied to visual document analysis. In *Seventh International Conference on Document Analysis and Recognition*, pp. 958–963, 2013.

Samuel L Smith and Quoc V Le. A bayesian perspective on generalization and stochastic gradient descent. *arXiv preprint arXiv:1710.06451*, 2017.

Ruoyu Sun. Optimization for deep learning: theory and algorithms. *arXiv preprint arXiv:1912.08957*, 2019.

Chong Wang, Xi Chen, Alexander J Smola, and Eric P Xing. Variance reduction for stochastic gradient optimization. In *Advances in Neural Information Processing Systems*, pp. 181–189, 2013.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, pp. 5754–5764, 2019.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pp. 649–657, 2015.

Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient langevin dynamics. In *Conference on Learning Theory*, pp. 1980–2022, 2017.