Understanding the Generalization Benefit of Normalization Layers: Sharpness Reduction

Anonymous Author(s)
Affiliation
Address
email

Abstract

Normalization layers (e.g., Batch Normalization, Layer Normalization) were introduced to help with optimization difficulties in very deep nets, but they clearly also help generalization, even in not-so-deep nets. Motivated by the long-held belief that flatter minima lead to better generalization, this paper gives mathematical analysis and supporting experiments suggesting that normalization (together with accompanying weight-decay) encourages GD to reduce the sharpness of loss surface. Here "sharpness" needs careful definition when loss is scale-invariant, a known consequence of normalization. Our mathematical analysis proceeds via analyzing the trajectory of GD and applies to fairly general settings in deep learning.

1 Introduction

Training modern deep neural nets crucially rely on normalization layers to make the training process less sensitive to hyperparameters and initialization. Popular normalization layers include Batch Normalization (BN) [25], Layer Normalization [4], etc. Normalization layers amount to a reparametrization of the neural net so that the loss becomes invariant to scale of most parameters (and with a minor change, to *all* parameters): $\mathcal{L}(c\boldsymbol{w}) = \mathcal{L}(\boldsymbol{w})$ for all scalings $c > 0$ [25, 2, 37]. The current paper assumes scale-invariance, as well as *weight decay* (WD):

$$\boldsymbol{w}_{t+1} \leftarrow (1 - \hat{\eta}\hat{\lambda})\boldsymbol{w}_t - \hat{\eta}\nabla\mathcal{L}(\boldsymbol{w}_t). \tag{1}$$

Use of WD may appear nonsensical at first sight because traditionally it is used to penalize large parameter norm, which of course is inconsequential for scale-invariant loss — one can scale down the parameter norm arbitrarily without changing the loss value. However, the scale of the parameter *does* matter for gradient and Hessian, and thus weight decay can affect the training dynamics. In particular, simple calculus shows $\nabla\mathcal{L}(\boldsymbol{w}) = \frac{1}{\|\boldsymbol{w}\|_2}\nabla\mathcal{L}(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2})$ and $\nabla^2\mathcal{L}(\boldsymbol{w}) = \frac{1}{\|\boldsymbol{w}\|_2^2}\nabla^2\mathcal{L}(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2})$. Thus WD is in effect trying to increase the norm of gradient and Hessian in training. This makes the training dynamics very different from unnormalized nets, and requires revisiting classical convergence analyses, as was done in various papers [2, 37, 40]. Such a change in dynamics also yields a very different solution at the end — in particular a solution that generalizes better [71].

The goal of the current paper is to improve mathematical understanding of how normalization together with accompaning WD can improve generalization. While this may arise from many places, we focus on sharpness-like measures formalized in terms of spectral norm of the Hessian. We exhibit settings where normalization provably induces an implicit bias to persistently reduce sharpness in training, which we call the *sharpness-reduction bias*. See Figures 1 and 2 for experiments on matrix completion (with BN) and CIFAR-10.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

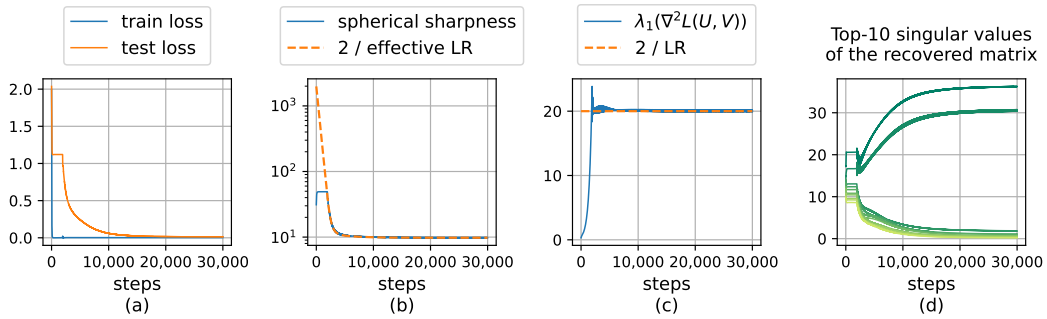

Figure 1: Experiment on overparameterized matrix completion with Batch Normalization. Given 800 (32%) entries Ω of a rank-2 matrix $\boldsymbol{M} \in \mathbb{R}^{50\times 50}$, use GD+WD to optimize $\mathcal{L}(\boldsymbol{U},\boldsymbol{V}) := \frac{1}{|\Omega|}\sum_{(i,j)\in\Omega}(\mathrm{BN}([\boldsymbol{U}\boldsymbol{V}^\top]_{i,j}) - M_{i,j})^2$, where $\boldsymbol{U},\boldsymbol{V} \in \mathbb{R}^{50\times 50}$ (thus no explicit constraint on rank). Spherical sharpness starts to decrease significantly around step 2000 (**b**), which encourages low-rank (**d**) and causes the test loss (MSE of all entries) to decrease from 1.12 to 0.013 (**a**). See Appendix P.1 for details.

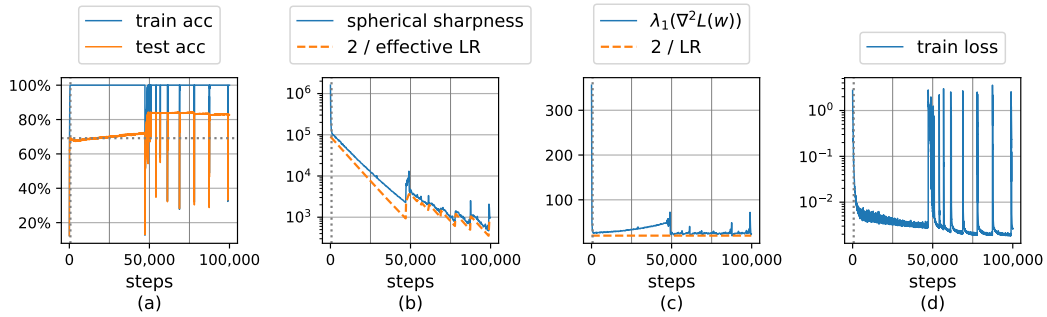

Figure 2: Training a smooth and scale-invariant VGG-11 on CIFAR-10 for a longer time with (full-batch) GD+WD leads to lower sharpness and higher test accuracy. BN is added after every linear layer to ensure scale-invariance. 100% training accuracy is achieved after ~ 680 steps (dashed line), but as the training continues for $47k$ steps, the spherical sharpness keeps decreasing (**b**) and the test accuracy increases from 69.1% to 72.0% (**a**). Then the training exhibits destabilization but the test accuracy is further boosted to 84.3%. See Appendix P.2 for details.

It is long believed that flatter minima generalize better [22] and is made precise by Neyshabur et al. [52] via PAC-Bayes theory [51]. It is usually noted that the notion of sharpness/flatness needs to be carefully defined in consideration of various kind of symmetries in neural networks, The most straightforward attempt to define sharpness is to measure it as the maximum eigenvalue of Hessian, namely $\lambda_1(\nabla^2\mathcal{L}(\boldsymbol{w}_t))$. But for normalized nets, this sharpness measure is vulnerable to weight rescaling, because one can scale the weight norm to make a minimizer arbitrarily flat [13]. Also, this sharpness measure may not be decreasing with the number of training steps; see Figure 2c. In fact, an empirical study by Cohen et al. [9] measured the sharpness by $\lambda_1(\nabla^2\mathcal{L}(\boldsymbol{w}_t))$ for various neural nets (including normalized nets) and argued that GD has an overwhelming tendency to persistently increase this sharpness measure until it reaches the *Edge of Stability (EoS) regime*, in which $\lambda_1(\nabla^2\mathcal{L}(\boldsymbol{w}_t))$ stabilizes around $2/\hat{\eta}$ ($\hat{\eta}$ is the learning rate).

The sharpness measure we use in this paper takes the scale-invariance property into account, and techniques from previous works [52, 15] can be easily adopted here to establish a PAC-Bayes bound on the test error, where our sharpness measure appears as an additive term (see Appendix C).

Definition 1.1 (Spherical Sharpness)**.** For a scale-invariant loss $\mathcal{L}(\boldsymbol{w})$ (i.e., $\mathcal{L}(c\boldsymbol{w}) = \mathcal{L}(\boldsymbol{w})$ for all $c > 0$), the spherical sharpness at $\boldsymbol{w} \in \mathbb{R}^D$ is defined by $\lambda_1(\nabla\mathcal{L}^2(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2}))$, the maximum eigenvalue of the Hessian matrix after projecting $\boldsymbol{w}$ onto the unit sphere.

1.1 Our Contributions

In this work, we study the aforementioned sharpness-reduction bias of gradient descent, assuming the loss is scale-invariant due to the presence of normalization. We rewrite GD+WD (1) with constant learning rate $\hat{\eta}$ on scale-invariant loss exactly as a specific kind of Projected Gradient Descent (PGD) on the unit sphere, $\boldsymbol{\theta}_{t+1} \leftarrow \Pi(\boldsymbol{\theta}_t - \tilde{\eta}_t\nabla\mathcal{L}(\boldsymbol{\theta}_t))$, where $\boldsymbol{\theta}_t := \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|_2}$ is the direction of $\boldsymbol{w}_t$, and $\tilde{\eta}_t$ is

the "effective" learning rate at step t (defined in Section 3). In this view, $\tilde{\eta}_t$ changes in a similar way as adaptive gradient methods — $\tilde{\eta}_t$ increases when gradient is small, and decreases when gradient is large. Our main contributions are as follows:

1. We theoretically show that once gradient descent reaches the manifold of minimizers of $\mathcal{L}(\cdot)$, the effective learning rate will increase until $2/\tilde{\eta}_t$ roughly equals to the spherical sharpness, which indicates that gradient descent enters the EoS regime (Section 4.1).
2. In the EoS regime, we show that with small learning rate, $\boldsymbol{\theta}_t$ oscillates around the manifold while moving forward and approximately tracking a sharpness-reduction flow, which is a gradient flow for minimizing sharpness on the manifold (with gradient-dependent learning rate) (Section 4.2).
3. As an application of our theory, we show that for linear regression with BN, GD+WD finds the minimizer that corresponds to the linear model with minimum weight norm, which looks surprising the same as the conventional effect of WD but is achieved through the completely different sharpness-reduction mechanism (Section 5).
4. We experimentally verified the sharpness-reduction bias phenomena predicted by our theorem and its benefit to generalization on CIFAR-10 with scale-invariant VGG-11 and ResNet-20, as well as matrix completion with BN (Appendix Q).
5. We generalize our theoretical result of sharpness-reduction bias to a broader class of adaptive gradient methods, most notably a variant of RMSprop with scalar learning rate (Appendix B). Our proof technique is novel and may have independent interest to the ML community.

Technical Contribution. The main challenge to establish our theorem is that we need to analyze the implicit bias of GD in the EoS regime. In particular, we crucially rely on step size being finite — this is in sharp contrast to many previous works on implicit bias of GD [61, 60, 47, 27, 18, 17, 36, 56, 1, 8, 38, 46, 48, 57, 62, 16] where the same bias exists at infinitesimal LR. Our analysis is inspired by a previous line of works [7, 10, 41] showing that label noise can drive SGD to move on the minimizer manifold along the direction of minimizing the trace of Hessian, but the key difference here is that we do not have any stochastic gradient noise. Instead, we study the EoS regime and exhibit that GD oscillates around the minimizer manifold by connecting it to power method. We show that this oscillation is a driving power that pushes the parameter to move on the manifold, and analyze the speed of this movement by modeling two key parameters of the dynamics as a 1-dimensional Hamiltonian system. To the best of our knowledge, our work is the first one that theoretically proves a sharpness measure to decrease in standard training processes with gradient descent, without any additional regularization such as label noise [7, 10, 41], or using non-standard variants of gradient descent update rule, e.g., using normalized GD or non-smooth wrappings on the loss function [3].

2 Related Works

Flatness and Generalization. It has been long believed that minima locate in a flat valley generalize better [22]. Li et al. [35] first visualized the loss landscape of neural networks and found "sharp" minima generalizes worse. Keskar et al. [29], Wu et al. [65], Jastrzębski et al. [26] empirically verified the positive correlation between flatness and generalizaiton. Neyshabur et al. [52] gave an theoretic explanation for generalization benefit of low sharpness using PAC-Bayes theory McAllester [51]. In light of this, Foret et al. [15] proposed SAM algorithm to improve the generalization of SGD by minimizing the sharpness of the loss. However, the definition of sharpness remains ambiguous in face of the positive homogeneity and invariance in ReLU networks [13], as networks with different with different parameters and sharpness be the same in function space. Towards this challenge, multiple definitions of scale-invariant sharpness has been proposed [67, 68, 63, 55]. Yi et al. [69], Kwon et al. [31] derived new algorithms with better generalization by defining new sharpness notions aware of positive homogeneity and invariance. He et al. [19] argues that the local minima of modern deep networks are more than being flat or sharp and could be asymmetric valleys.

Understanding Normalization Layers. BN is proposed in [25] with the original motivation being to reduce Internal Covariate Shift (ICS). Santurkar et al. [58] challenged this view by arguing that the

effectiveness of BN comes from a smoothing effect on the training objective. Several existing works have reported that optimization with normalization can be far away from classical optimization theory. Li and Arora [37] showed that one can train normalized networks (which is scale-invariant) even with exponential increasing LR. Li et al. [39] showed that GD+WD on scale-invariant loss can leave the stable regime. Lobacheva et al. [43] empirically showed that GD+WD on scale-invariant loss leads to a periodic behavior that sometimes improves the test accuracy. However, despite a lot of effort has been devoted to understanding the optimization and generalization benefits of normalization in various specific settings, *e.g.*, [64, 6, 71, 45, 12, 42, 11, 44, 32, 50], we still lack a complete and rigorous analysis for the role of normalization in general, especially in terms of generalization.

3 Preliminaries

Let $\mathbb{S}^{D-1} := \{\boldsymbol{\theta} \in \mathbb{R}^D : \|\boldsymbol{\theta}\|_2 = 1\}$ be the unit sphere equipped with subspace topology. We say a loss function $\mathcal{L}(\boldsymbol{w})$ defined on $\mathbb{R}^D \setminus \{\mathbf{0}\}$ is *scale-invariant* if $\mathcal{L}(c\boldsymbol{w}) = \mathcal{L}(\boldsymbol{w})$ for all $c > 0$. In other words, the loss value does not change with the parameter norm. For a differentiable scale-invariant function $\mathcal{L}(\boldsymbol{w})$, the gradient is (-1)-homogeneous and it is always perpendicular to $\boldsymbol{w}$, i.e., $\nabla\mathcal{L}(c\boldsymbol{w}) = c^{-1}\nabla\mathcal{L}(\boldsymbol{w})$ for all $c > 0$ and $\langle \nabla\mathcal{L}(\boldsymbol{w}), \boldsymbol{w}\rangle = 0$. The focus of this paper is the dynamics of GD+WD on scale-invariant loss. (1) gives the update rule for learning rate (LR) $\hat{\eta}$ and weight decay (WD) $\hat{\lambda}$. We use $\boldsymbol{\theta}_t := \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|_2}$ to denote the projection of $\boldsymbol{w}_t$ onto the unit sphere at step t. A number of works [64, 23, 2] propose to use the "effective learning rate" $\frac{\hat{\eta}}{\|\boldsymbol{w}_t\|_2^2}$ to measure the update efficiency of $\boldsymbol{\theta}_t$ on $\mathbb{S}^{D-1}$. Inspired by this notion, we write GD+WD on scale-invariant loss as a specific kind of Projected Gradient Descent (PGD), and define $\tilde{\eta}_t := \frac{\hat{\eta}}{(1-\hat{\eta}\hat{\lambda})\|\boldsymbol{w}_t\|_2^2}$ to be the effective learning rate with slight abuse of terminology.

Lemma 3.1. *When the parameters $\boldsymbol{w}_t$ are updated as (1), $\boldsymbol{\theta}_t$ satisfies the following equation:*

$$\boldsymbol{\theta}_{t+1} = \Pi(\boldsymbol{\theta}_t - \tilde{\eta}_t \nabla\mathcal{L}(\boldsymbol{\theta}_t, \boldsymbol{z}_t)), \tag{2}$$

*where $\tilde{\eta}_t := \frac{\hat{\eta}}{(1-\hat{\eta}\hat{\lambda})\|\boldsymbol{w}_t\|_2^2}$ is called the **effective learning rate** at step t, and $\Pi : \boldsymbol{w} \mapsto \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2}$ is the projection operator that projects any vector onto the unit sphere.*

The convergence rate of GD+WD has been analyzed by Li et al. [40]. Here we present a variant of their theorem that bounds both the gradient and effective LR.

Theorem 3.2 (Variant of Theorem 4.1 in [40])**.** *Let $\mathcal{L}(\boldsymbol{w})$ be a scale-invariant loss function and $\rho_2 := \sup\{\|\nabla^2\mathcal{L}(\boldsymbol{w})\|_2 : \boldsymbol{w} \in \mathbb{S}^{D-1}\}$ be the smoothness constant of $\mathcal{L}$ restricted on the unit sphere. For GD+WD (1) with $\hat{\eta}\hat{\lambda} \le 1/2$ and $\tilde{\eta}_0 \le \frac{1}{\pi^2\rho_2}$, after at most $T_0 := \left\lceil \frac{1}{2\hat{\eta}\hat{\lambda}}\left(\ln\frac{\|\boldsymbol{w}_0\|_2^2}{\rho_2\pi^2\hat{\eta}} + 3\right)\right\rceil$ steps, there must exist $0 \le t \le T_0$ such that $\|\nabla\mathcal{L}(\boldsymbol{\theta}_t)\|_2^2 \le 8\pi^4\rho_2^2\hat{\lambda}\hat{\eta}$ and $\tilde{\eta}_t \le \frac{1}{\pi^2\rho_2}$.*

4 GD+WD on Scale-Invariant Loss Functions

In this section, we analyze GD+WD (1) on a scale-invariant loss $\mathcal{L}(\boldsymbol{w})$. We use $\boldsymbol{w}_t$ to denote the trainable parameter at step t and $\boldsymbol{\theta}_t := \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|_2}$ to denote its projection onto $\mathbb{S}^{D-1}$. In Section 4.1, we analyze the dynamics in the stable regime, which is a regime that the loss is guaranteed to decrease. Theorem 3.2 suggests that $\boldsymbol{w}_t$ can get close to a local minimizer at some time t_0, but we show that the effective LR is increasing after t_0 until GD+WD eventually leaves this stable regime and enters a new regime which we call the Edge of Stability (EoS). In Section 4.2, we establish our main theorem, which connects the dynamics of $\boldsymbol{w}_t$ in the EoS regime to a sharpness-reduction flow.

4.1 GD+WD Eventually Leaves the Stable Regime

A standard step for analyzing optimization methods is to do Taylor expansion locally for the loss function, and show that how the optimization method decreases the loss using a *descent lemma*. In our case of scale-invariant loss functions, we use $\boldsymbol{H}(\boldsymbol{w}) := \nabla^2\mathcal{L}(\boldsymbol{w}) \in \mathbb{R}^{D\times D}$ to denote the Hessian matrix of $\mathcal{L}$ at $\boldsymbol{w} \in \mathbb{R}^D$, and $\lambda_1^{\mathrm{H}}(\boldsymbol{w}) := \lambda_1(\boldsymbol{H}(\boldsymbol{w}))$ to denote the top eigenvalue of $\boldsymbol{H}(\boldsymbol{w})$.

Lemma 4.1 (Descent Lemma)**.** *For scale-invariant loss $\mathcal{L}(\boldsymbol{w})$, at step t of GD+WD we have*

$$\mathcal{L}(\boldsymbol{\theta}_{t+1}) \le \mathcal{L}(\boldsymbol{\theta}_t) - \tilde{\eta}_t(1 - \tilde{\eta}_t \lambda_{\max}^{(t)}/2)\|\nabla \mathcal{L}(\boldsymbol{\theta}_t)\|_2^2.$$

where $\lambda_{\max}^{(t)} := \sup_{\alpha \in [0, \tilde{\eta}_t]} \left\{ \lambda_1^{\mathrm{H}}(\boldsymbol{\theta}_t - \alpha \nabla \mathcal{L}(\boldsymbol{\theta}_t)) \right\}$ is an upper bound of spherical sharpness locally.

This descent lemma shows that the training loss $\mathcal{L}(\boldsymbol{\theta}_t)$ is decreasing as long as the effective LR $\tilde{\eta}_t$ is smaller than $2/\lambda_{\max}^{(t)}$, We call the regime of $\tilde{\eta}_t < 2/\lambda_{\max}^{(t)}$ as the *stable regime* of GD+WD. If $\tilde{\eta}_t \approx 2/\lambda_{\max}^{(t)}$ with a small error, then we call it as the *Edge of Stability (EoS) regime*.

Fix an initial point $\boldsymbol{w}_0 \in \mathbb{R}^D \setminus \{\boldsymbol{0}\}$. Now we aim to characterize the dynamics of GD+WD when LR $\hat{\eta}$ and WD $\hat{\lambda}$ are small enough. Theorem 3.2 shows that for some $t_0 \le T_0$, $\|\nabla \mathcal{L}(\boldsymbol{\theta}_{t_0})\|_2^2 \le O(\hat{\lambda}\hat{\eta})$ and $\tilde{\eta}_{t_0} \le \frac{1}{\pi^2 \rho_2} < \frac{2}{\rho_2}$, which means $\boldsymbol{\theta}_{t_0}$ is an approximate first-order stationary point of $\mathcal{L}$ on the unit sphere. This does not guarantee that $\boldsymbol{\theta}_{t_0}$ is close to any global minimizer, but in practice the training loss rarely gets stuck at a non-optimal value when the model is overparameterized [33, 53, 34, 70]. We are thus motivated to study the case where $\boldsymbol{\theta}_{t_0}$ not only has small gradient $\|\nabla \mathcal{L}(\boldsymbol{\theta}_{t_0})\|_2^2 \le O(\hat{\lambda}\hat{\eta})$ but also is close to a local minimizer $\boldsymbol{\theta}^* \in \mathbb{S}^{D-1}$ of $\mathcal{L}$ in the sense that $\|\boldsymbol{\theta}_{t_0} - \boldsymbol{\theta}^*\|_2 \le O((\hat{\lambda}\hat{\eta})^{1/2})$ (note that the latter implies the former assuming smoothness).

As the gradient is small near the local minimizer $\boldsymbol{\theta}^*$, starting from step t_0, the norm of $\boldsymbol{w}_t$ decreases due to the effect of WD. See Figure 3a. Since the effective LR is inversely proportional to $\|\boldsymbol{w}_t\|_2^2$, this leads to the effective LR to increase. Then Theorem 4.3 will show that the GD+WD dynamic eventually leaves the stable regime at some time $t_1 > t_0$, and enters the EoS regime where $\tilde{\eta}_t \approx 2/\lambda_{\max}^{(t)}$.

To establish Theorem 4.3, we need to assume that $\mathcal{L}$ satisfies Polyak-Łojasiewicz (PL) condition locally, which is a standard regularity condition in the optimization literature to ease theoretical analysis around a minimizer. Intuitively, PL condition guarantees that the gradient grows faster than a quadratic function as we move a parameter $\boldsymbol{\theta}$ away from $\boldsymbol{\theta}^*$. Note that PL condition is strictly weaker than convexity as the function can still be non-convex under PL condition (see, e.g., [28]).

Definition 4.2 (Polyak-Łojasiewicz Condition)**.** For a scale-invariant loss $\mathcal{L}(\boldsymbol{w})$ and $\mu > 0$, we say that $\mathcal{L}$ satisfies μ-Polyak-Łojasiewicz condition (or μ-PL) locally around a local minimizer $\boldsymbol{\theta}^*$ on $\mathbb{S}^{D-1}$ if for some neighborhood $U \subseteq \mathbb{S}^{D-1}$ of $\boldsymbol{\theta}^*$, $\forall \boldsymbol{\theta} \in U : \frac{1}{2}\|\nabla \mathcal{L}(\boldsymbol{\theta})\|_2^2 \ge \mu \cdot (\mathcal{L}(\boldsymbol{\theta}) - \mathcal{L}(\boldsymbol{\theta}^*))$.

Theorem 4.3. *Let $\mathcal{L}(\boldsymbol{w})$ be a scale-invariant loss that satisfies μ-PL around a local minimizer $\boldsymbol{\theta}^*$ on the unit sphere (Definition 4.2). For GD+WD on $\mathcal{L}(\boldsymbol{w})$ with learning rate $\hat{\eta}$ and weight decay $\hat{\lambda}$, if at some step t_0, $\|\boldsymbol{\theta}_{t_0} - \boldsymbol{\theta}^*\|_2 \le O((\hat{\lambda}\hat{\eta})^{1/2})$ and $\tilde{\eta}_{t_0} < \frac{2}{\rho_2}$, where $\boldsymbol{\theta}^* \in \mathbb{S}^{D-1}$ is a local minimizer, $\hat{\lambda}\hat{\eta}$ is small, and $\rho_2 := \sup\{\|\nabla^2 \mathcal{L}(\boldsymbol{w})\|_2 : \boldsymbol{w} \in \mathbb{S}^{D-1}\}$, then there exists a time $t_1 > t_0$ such that $\|\boldsymbol{\theta}_{t_1} - \boldsymbol{\theta}^*\|_2 = O((\hat{\lambda}\hat{\eta})^{1/2})$ and $\tilde{\eta}_{t_1} = \frac{2}{\lambda_1^{\mathrm{H}}(\boldsymbol{\theta}^*)} + O((\hat{\eta}\hat{\lambda})^{1/2})$.*

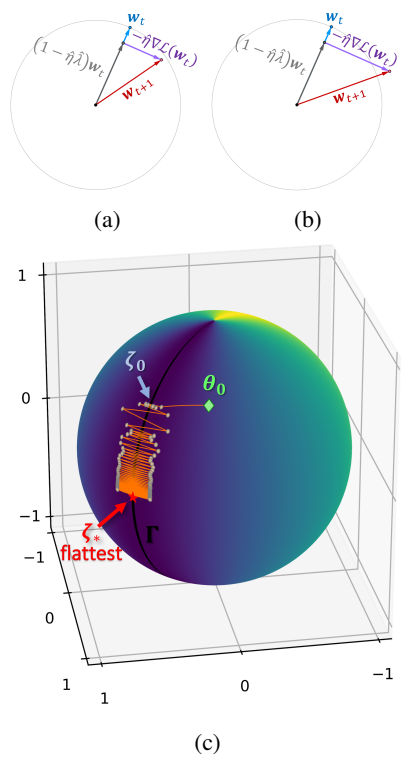

Figure 3: **(a), (b)**: The norm of $\boldsymbol{w}_t$ decreases when gradient is small and increases when gradient is large. **(c)**: Dynamics of $\boldsymbol{\theta}_t$ on a 3D scale-invariant function, where darker color means lower loss (see Appendix F). All points in the black line are minimizers. $\boldsymbol{\theta}_t$ eventually moves close to the flattest one (red star).

4.2 Dynamics at the Edge of Stability

From the analysis in the previous subsection, we know that $\boldsymbol{\theta}_t$ can get close to a local minimizer $\boldsymbol{\theta}^*$ and enter the EoS regime at some step t_1. But what happens after t_1?

Figure 3c gives a warm-up example on a 3D scale-invariant loss $\mathcal{L}:\mathbb{R}^3\to\mathbb{R}$, where the black line is a manifold Γ consisting of all the minimizers of the loss. In training with GD+WD, $\boldsymbol{\theta}_t$ first goes close to a local minimizer $\boldsymbol{\zeta}_0$, then Theorem 4.3 suggests that the effective LR steadily increases due to WD until the EoS regime is reached. Now something interesting happens — $\boldsymbol{\theta}_t$ moves a bit away from $\boldsymbol{\zeta}_0$ and starts to oscillate around the manifold Γ. This oscillation is not completely perpendicular to Γ but forms a small angle that pushes $\boldsymbol{\theta}_t$ to move downward persistently until $\boldsymbol{\theta}_t$ goes near the minimizer $\boldsymbol{\zeta}_*$.

For a general scale-invariant loss $\mathcal{L}:\mathbb{R}^D\to\mathbb{R}$, which minimizer does $\boldsymbol{\theta}_t$ move towards? Assuming the existence of a local minimizer manifold and some regularity conditions, we can prove that $\boldsymbol{\theta}_t$ always oscillates around the manifold and moves towards a direction that reduces spherical sharpness. Now we introduce our assumptions.

4.2.1 Assumptions

We assume that there is a manifold Γ consisting of local minimizers. Later in our theorem, we will analyze the dynamics when $\boldsymbol{\theta}_{t_1}$ is close to the manifold.

Assumption 4.4. The loss function $\mathcal{L}:\mathbb{R}^D\setminus\{\mathbf{0}\}\to\mathbb{R}$ is $\mathcal{C}^4$-smooth and scale-invariant. Γ is a $\mathcal{C}^2$-smooth, $(D_\Gamma-1)$-dimensional submanifold of $\mathbb{S}^{D-1}$ for some $0\le D_\Gamma<D$, where every $\boldsymbol{\theta}\in\Gamma$ is a local minimizer of $\mathcal{L}$ on $\mathbb{S}^{D-1}$ and $\operatorname{rank}(\boldsymbol{H}(\boldsymbol{\theta}))=D-D_\Gamma$.

Scale-invariance has become a standard assumption in studying neural nets with normalization layers [37, 39, 43]. For VGG and ResNet, the scale-invariance can be ensured with minor changes to the architectures (see Appendix P.2). The training loss of a neural net may not be smooth if the activation is ReLU, but lately it has become clear that differentiable activations such as Swish [54] can perform equally well. We use Swish in our VGG-11 experiments (Figure 2), but we note that ResNet with ReLU activation also exhibits the sharpness-reduction bias empirically (see Appendix Q).

For any local minimizer $\boldsymbol{\theta}\in\Gamma$, the eigenvalues $\lambda_k^{\mathrm{H}}(\boldsymbol{\theta})$ must be non-negative. And $\lambda_k^{\mathrm{H}}(\boldsymbol{\theta})=0$ for all $D-D_\Gamma<k\le D$, since Γ is of dimension $D_\Gamma-1$. The condition $\operatorname{rank}(\boldsymbol{H}(\boldsymbol{\theta}))=D-D_\Gamma$ ensures that the Hessian is maximally non-degenerate on Γ, which also appears as a key assumption in previous works [41, 3, 14]. This condition simplifies the calculus on Γ in our analysis as it ensures that the null space of the matrix $\boldsymbol{H}(\boldsymbol{\theta})$ equals to the tangent space of Γ at $\boldsymbol{\theta}\in\Gamma$. It is also closely related to PL condition (Definition 4.2) as Assumption 4.4 implies that $\mathcal{L}(\boldsymbol{\theta})$ satisfies μ-PL (for some $\mu>0$) locally around every $\boldsymbol{\theta}\in\Gamma$ on the unit sphere (Arora et al. [3], Lemma B.3).

To ease our analysis, we also need the following regularity condition to ensure that the largest eigenvalue is unique. In our experiments, sharpness reduction happens even when the multiplicity of the top eigenvalue is more than 1, but we leave the analysis of that case to future work.

Assumption 4.5. For all $\boldsymbol{\theta}\in\Gamma$, $\lambda_1^{\mathrm{H}}(\boldsymbol{\theta})>\lambda_2^{\mathrm{H}}(\boldsymbol{\theta})$. That is, the top eigenvalue of $\boldsymbol{H}(\boldsymbol{\theta})$ is unique.

4.2.2 Main Theorem

First, we define $\eta_{\mathrm{in}}:=\hat{\eta}\hat{\lambda}$ as the intrinsic learning rate (name from Li et al. 39) for convenience. As suggested in Theorems 3.2 and 4.3, $\boldsymbol{\theta}_t$ can get close to a local minimizer and be in the EoS regime at some step t_1: if $\boldsymbol{\zeta}_0$ is the local minimizer, then $\|\boldsymbol{\theta}_{t_1}-\boldsymbol{\zeta}_0\|_2=O(\eta_{\mathrm{in}}^{1/2})$ and $\tilde{\eta}_{t_1}=\frac{2}{\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}_0)}+O(\eta_{\mathrm{in}}^{1/2})$, In our main theorem, we start our analysis from step t_1 while setting $t_1=0$ WLOG (otherwise we can shift the step numbers). It connects GD+WD in the EoS regime to a gradient flow on the manifold Γ minimizing the spherical sharpness (with gradient-dependent learning rate), where one step of GD+WD tracks a time interval of η_{in} in the gradient flow.

$$\boldsymbol{\zeta}(0)=\boldsymbol{\zeta}_0\in\Gamma,\qquad \frac{\mathrm{d}}{\mathrm{d}\tau}\boldsymbol{\zeta}(\tau)=-\frac{2\nabla_\Gamma\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))}{4+\|\nabla_\Gamma\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))\|_2^2},\tag{3}$$

Here we use the notation $\nabla_\Gamma R(\boldsymbol{\theta})$ for any $R:\mathbb{R}^D\to\mathbb{R}$ to denote the projection of $\nabla R(\boldsymbol{\theta})$ onto the tangent space $\mathsf{T}_{\boldsymbol{\theta}}(\Gamma)$ at $\boldsymbol{\theta}\in\Gamma$. $\boldsymbol{\zeta}(\tau)$ reduces sharpness as it moves in direction of the negative gradient of $\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))$ on Γ. A simple chain rule shows how fast the spherical sharpness decreases:

$$\frac{\mathrm{d}}{\mathrm{d}t}\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))=-\frac{2\|\nabla_\Gamma\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))\|_2^2}{4+\|\nabla_\Gamma\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))\|_2^2}\approx\begin{cases}-\frac{1}{2}\|\nabla_\Gamma\log\lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(\tau))\|_2^2 & \text{for small gradient;}\\ -2 & \text{for large gradient.}\end{cases}$$

Note that it is not enough to just assume that $\boldsymbol{\theta}_0$ is close to $\boldsymbol{\zeta}_0$. If $\boldsymbol{\theta}_0 = \boldsymbol{\zeta}_0$ holds exactly, then the subsequent dynamic of $\boldsymbol{w}_t$ is described by $\boldsymbol{w}_t = (1 - \hat{\eta}\hat{\lambda})^t \boldsymbol{w}_0$ with direction unchanged. There are also some other bad initial directions of $\boldsymbol{w}_0$ that may not lead to the sharpness-reduction bias. This motivates us to do a smoothed analysis for the initial direction: the initial direction is $\boldsymbol{\zeta}$ with tiny random perturbation, where the perturbation scale is allowed to vary from $\exp(-\eta_{\text{in}}^{-o(1)})$ to $\eta_{\text{in}}^{1/2-o(1)}$, and we show that a good initial direction is met with high probability as $\eta_{\text{in}} \to 0$.[1] Alternately, one can regard it as a modeling of the tiny random noise in GD+WD due to the precision errors in floating-point operations. See Figure 4b; the training loss can never be exactly zero in practice.

Theorem 4.6. *Under Assumptions 4.4 and 4.5, let $\boldsymbol{\zeta}_0 \in \Gamma$ be a point on the local minimizer manifold. Consider GD+WD with learning rate $\hat{\eta} > 0$ and weight decay $\hat{\lambda} > 0$. Let $\eta_{\text{in}} := \hat{\eta}\hat{\lambda}$ be the intrinsic learning rate. Suppose that the direction and norm of $\boldsymbol{w}_0$ are initialized separately as follows:*

1. *The initial direction $\frac{\boldsymbol{w}_0}{\|\boldsymbol{w}_0\|_2}$ is set to $\boldsymbol{\zeta}_0$ with a tiny random perturbation: $\frac{\boldsymbol{\zeta}_0+\boldsymbol{\xi}}{\|\boldsymbol{\zeta}_0+\boldsymbol{\xi}\|_2}$, where $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \sigma_0^2 \boldsymbol{I}/D)$ is Gaussian noise with $\sigma_0 \in [\exp(-\eta_{\text{in}}^{-o(1)}), \eta_{\text{in}}^{1/2-o(1)}]$;*
2. *The initial parameter norm $\|\boldsymbol{w}_0\|_2$ is chosen so that the initial effective learning rate $\tilde{\eta}_0 := \frac{\hat{\eta}}{(1-\eta_{\text{in}})\|\boldsymbol{w}_0\|_2^2}$ satisfies $\left|\tilde{\eta}_0 - \frac{2}{\lambda_1^{\text{H}}(\boldsymbol{\zeta}_0)}\right| \le \eta_{\text{in}}^{1/2-o(1)}$.*

Then with probability $1 - O(\eta_{\text{in}}^{1/2})$, the trajectory of $\boldsymbol{\theta}_t := \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|_2}$ is close to a sharpness-reduction flow $\boldsymbol{\zeta} : [0, T] \to \Gamma$ that starts from $\boldsymbol{\zeta}_0$ and evolves as the ODE (3) (if solution exists), in the sense that $\|\boldsymbol{\theta}_t - \boldsymbol{\zeta}(t\eta_{\text{in}})\|_2 = O(\eta_{\text{in}}^{1/4-o(1)})$ for all $0 \le t \le T/\eta_{\text{in}}$.

Remark 4.7 (Magnitude of Oscillation). As suggested by Figure 3c, $\boldsymbol{\theta}_t$ actually oscillates around the manifold, but our bound shows the closeness of $\boldsymbol{\theta}_t$ and $\boldsymbol{\zeta}(t\eta_{\text{in}})$ directly, because the magnitude of oscillation is as small as $O(\eta_{\text{in}}^{1/2-o(1)})$, so it is absorbed into our error bound $O(\eta_{\text{in}}^{1/4-o(1)})$.

4.2.3 Proof Idea

Throughout our proof, we view GD+WD for $\boldsymbol{w}_t$ as a PGD for $\boldsymbol{\theta}_t$ with effective LR $\tilde{\eta}_t$ (Lemma 3.1). To track $\boldsymbol{\theta}_t$ with $\boldsymbol{\zeta}(t\eta_{\text{in}})$, for each step t, we construct a local minimizer $\boldsymbol{\phi}_t \in \Gamma$ that serves as the "projection" of $\boldsymbol{\theta}_t$ onto the manifold Γ, in the sense that the displacement $\boldsymbol{x}_t := \boldsymbol{\theta}_t - \boldsymbol{\phi}_t$ is approximately perpendicular to the tangent space of Γ at $\boldsymbol{\phi}_t$. Our entire proof works through induction. According to the initial conditions, the dynamic is initially in the EoS regime: $\|\boldsymbol{x}_t\|_2 \le \eta_{\text{in}}^{1/2-o(1)}$ and $|\tilde{\eta}_t - 2/\lambda_1^{\text{H}}(\boldsymbol{\phi}_t)| \le \eta_{\text{in}}^{1/2-o(1)}$ at $t = 0$. In our induction, we maintain the induction hypothesis that these two EoS conditions continue to hold for all $t \ge 0$.

Period-Two Oscillation. A key insight in our proof is that after a few initial steps, $\boldsymbol{\theta}_t$ is oscillating around $\boldsymbol{\phi}_t$ along the $\pm\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\theta})$ directions, where $\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\theta})$ is a unit top eigenvector of $\boldsymbol{H}(\boldsymbol{\theta})$ and is chosen in a way that $\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\theta})$ is continuous on Γ. More specifically, $\boldsymbol{x}_t = h_t \boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t) + O(\|\boldsymbol{x}_t\|_2^2)$ for $h_t := \langle \boldsymbol{x}_t, \boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t)\rangle$. The oscillation is of period 2: $h_t > 0$ when t is even and $h_t < 0$ when t is odd. See Figure 4d for an example.

This oscillation can be connected to a power method for the matrix $\boldsymbol{I} - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t)$. In the EoS regime, we can approximate $\boldsymbol{\theta}_{t+1}$ (when $\boldsymbol{x}_t$ is small) as $\boldsymbol{\theta}_{t+1} = \Pi(\boldsymbol{\theta}_t - \tilde{\eta}_t \nabla\mathcal{L}(\boldsymbol{\theta}_t)) \approx \Pi(\boldsymbol{\theta}_t - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t)\boldsymbol{x}_t) \approx \boldsymbol{\theta}_t - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t)\boldsymbol{x}_t$ by Taylor expansions of $\nabla\mathcal{L}$ and $\Pi : \mathbb{R}^D \to \mathbb{S}^{D-1}$. We can further show that $\boldsymbol{\phi}_{t+1} \approx \boldsymbol{\phi}_t$ due to our choice of projections. Then the connection to power method is shown below:

$$\boldsymbol{x}_{t+1} \approx \boldsymbol{\theta}_{t+1} - \boldsymbol{\phi}_t \approx (\boldsymbol{I} - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t))\boldsymbol{x}_t,$$

By simple linear algebra, $\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t)$ is an eigenvector of $\boldsymbol{I} - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t)$, associated with eigenvalue $1 - \tilde{\eta}_t \lambda_1^{\text{H}}(\boldsymbol{\phi}_t) \approx -1$. The remaining eigenvalues are $\{1 - \tilde{\eta}_t \lambda_i^{\text{H}}(\boldsymbol{\phi}_t)\}_{i=2}^D$, (where $\lambda_i^{\text{H}}(\boldsymbol{\phi}_t)$ is the i-th largest eigenvalue of $\boldsymbol{H}(\boldsymbol{\theta}_t)$), and they lie in the range $(-1, 1]$ since $\lambda_i^{\text{H}}(\boldsymbol{\phi}_t) \in [0, \lambda_1^{\text{H}}(\boldsymbol{\phi}_t))$. Using a similar analysis as power method, we show that $\boldsymbol{x}_t$ quickly align to the direction of $\pm\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t)$ after a few initial steps, as the corresponding eigenvalue has approximately the largest absolute value.[2]

[1] Here $\eta_{\text{in}}^{-o(1)}$ can be constant, $O(\log(1/\eta_{\text{in}}))$, or $O(\text{polylog}(1/\eta_{\text{in}}))$, but not $\eta_{\text{in}}^{-\epsilon}$ if $\epsilon > 0$ is a constant.

[2] Our construction of $\boldsymbol{\phi}_t$ ensures that $\boldsymbol{x}_t$ always has a small overlap with the 1-eigenspace of $\boldsymbol{I} - \tilde{\eta}_t \boldsymbol{H}(\boldsymbol{\phi}_t)$, so $\boldsymbol{x}_t$ can only align to $\pm\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t)$.

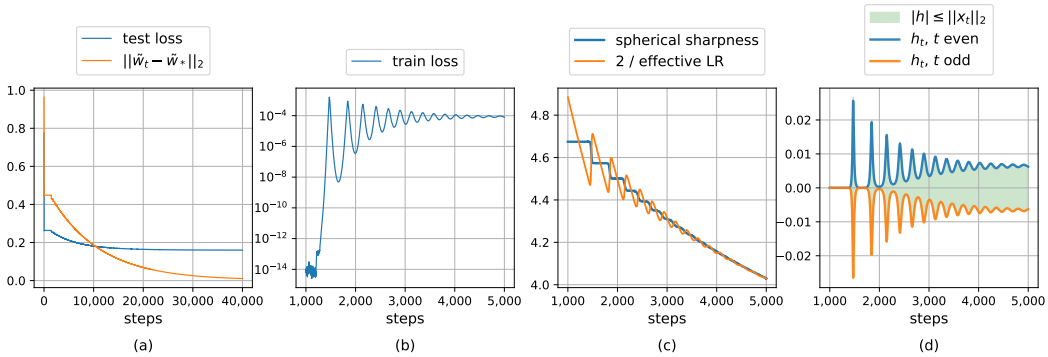

Figure 4: Illustration of the oscillation and periodic behaviors in the dynamics of GD+WD on linear regression with BN (see Sections 4.2.3 and 5). The training loss decreases to $\approx 10^{-14}$ in the first 1k steps and achieves test loss 0.26. Starting from step $\sim$ 1k, the dynamic enters the EoS regime. **(a).** The test loss decreases to 0.16 as a distance measure to the flattest solution ($\mathbb{M}$) decreases towards 0; **(b).** The training loss oscillates around $\sim 10^{-4}$ in the EoS regime; **(c).** $2/\tilde{\eta}_t$ and spherical sharpness decrease and exhibits a periodic behavior; **(d).** The parameter oscillates around the minimizer manifold along the top eigenvector direction.

To formally establish the above result, we need a tiny initial alignment between $\boldsymbol{x}_0$ and $\boldsymbol{v}_1^{\mathrm{H}}(\boldsymbol{\phi}_0)$, just as the initial condition in power method. This is where we need the initial random perturbation.

Oscillation Drives $\boldsymbol{\phi}_t$ to Move. This period-two oscillation is the driving power to push $\boldsymbol{\phi}_t$ to move on the manifold. The main idea here is to realize that the oscillation direction deviates slightly from the direction of $\pm\boldsymbol{v}_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$ if we use a higher-order approximation. We specifically use the Taylor approximation to show that this deviation leads $\boldsymbol{\phi}_t$ to move slightly on Γ: after each cycle of oscillation, $\boldsymbol{\phi}_{t+2} \approx \boldsymbol{\phi}_t - 4\eta_{\mathrm{in}} h_t^2 \nabla_\Gamma \log \lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t) + O(\eta_{\mathrm{in}}^{1.5-o(1)})$, which resembels two steps of gradient descent on Γ to minimize the logarithm of spherical sharpness with learning rate $2\eta_{\mathrm{in}} h_t^2$,

Periodic Behavior of h_t and $\tilde{\eta}_t$. It remains to analyze the dynamics of h_t so that we can know how fast the sharpness reduction is. Our analysis is inspired by an empirical study from Lobacheva et al. [43], which exhibits a periodic behavior of gradients and effective learning rates. In our setting, we show that h_t and $\tilde{\eta}_t$ *does* evolve periodically. See Figures 4c and 4d for an example.

The key is that $\tilde{\eta}_t$ changes as an adaptive gradient method: $\tilde{\eta}_t$ increases when gradient is small and decreases when gradient is large (due to the effect of WD; see Figures 3a and 3b). According to our power method approximation, $h_{t+2} \approx (1-\tilde{\eta}_t \lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t))^2 h_t$, so h_t decreases when $\tilde{\eta}_t < 2/\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$. But $\tilde{\eta}_t$ increases if h_t is small. When $\tilde{\eta}_t$ becomes larger than $2/\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$, h_t is switched from decreasing to increasing according to our approximation. But h_t cannot increase indefinitely because $\tilde{\eta}_t$ decreases when h_t is large enough. Finally, a period is finished when $\tilde{\eta}_t < 2/\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$ again. In Appendix N, we connect this periodic behavior with a 1-dimensional Hamiltonian system, which enables us to prove that each period is actually so short that we can approximate $2\eta_{\mathrm{in}} h_t^2$ in each step with its average value in a period to obtain the final flow approximation (3).

Extensions. We note that this periodic behavior is not limited to GD+WD on scale-invariant loss, as the above intuitive argument holds as long as the effective LR changes adaptively with respect to gradient change. Based on this intuition, an important notion called *Quasi-RMSprop scheduler* is proposed. For a PGD method, a learning rate scheduler is a rule for changing the effective LRs in each step, and Quasi-RMSprop is a specific class of schedulers we defined, including the way that the effective LRs change in GD+WD on scale-invariant loss (if viewed as PGD). Our proof is done in a unified way that works as long as the effective LRs are changed according to a Quasi-RMSprop scheduler. As a by-product, a similar theorem can be proved for GD (without projection) on non-scale-invariant loss, if a Quasi-RMSprop scheduler sets the LR in each step. For example, we can extend our analysis to RMSprop with a scalar learning rate. See Appendix B.

5 Case Study: Linear Regression with Batch Normalization

In this section, we analyze the GD+WD dynamics on linear regression with Batch Normalization (BN), as a simple application of our theory. Let $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ be a dataset, where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ are inputs and regression targets. We study the over-parameterized case where $d \gg n$, and we assume that the regression targets are generated by an unknown linear model.

A classic linear model is parameterized by $(\boldsymbol{w}, b) \in \mathbb{R}^d \times \mathbb{R}$ and outputs $\boldsymbol{w}^\top \boldsymbol{x} + b$ given input $\boldsymbol{x}$, but now we add a BN to the output. More specifically, we consider a batch-normalized linear model $\Phi(\boldsymbol{x}; \boldsymbol{w}, \gamma, \beta) := \gamma \cdot \frac{\boldsymbol{w}^\top \boldsymbol{x} - \mu_1}{\sigma_1} + \beta$, where μ_1, σ_1 are the mean and variance of $\{\boldsymbol{w}^\top \boldsymbol{x}_i\}_{i=1}^n$ over the whole dataset[3], and the bias term b is cancelled out due to BN. Note that $\Phi(\boldsymbol{x}; \boldsymbol{w}, \gamma, \beta)$ is still a linear function with respect to $\boldsymbol{x}$. Let $\boldsymbol{\mu}_{\mathrm{x}} \in \mathbb{R}^d$ and $\boldsymbol{\Sigma}_{\mathrm{x}} \in \mathbb{R}^{d \times d}$ be the mean and covariance of the input data $\{\boldsymbol{x}_i\}_{i=1}^n$. Then $\Phi(\boldsymbol{x}; \boldsymbol{w}, \gamma, \beta)$ can be rewritten as:

$$\Phi(\boldsymbol{x}; \boldsymbol{w}, \gamma, \beta) = \tilde{\boldsymbol{w}}^\top \boldsymbol{x} + \tilde{b}, \qquad \text{where} \quad \tilde{\boldsymbol{w}} := \gamma \boldsymbol{w} / \|\boldsymbol{w}\|_{\boldsymbol{\Sigma}_{\mathrm{x}}}, \quad \tilde{b} := \beta - \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_{\mathrm{x}}. \tag{4}$$

No matter how $\boldsymbol{w}$ is set, the output mean and variance of Φ are always β and γ^2. To simplify our analysis, we fix β, γ to be non-trainable constants so that the mean and variance of Φ's output match with those of $\{y_i\}_{i=1}^n$, that is, we set $\beta = \mu_{\mathrm{y}}$ and $\gamma = \sigma_{\mathrm{y}}$ to be the mean and standard deviation of y_i over the whole dataset. Then the training loss is $\mathcal{L}(\boldsymbol{w}) := \frac{1}{n} \sum_{i \in [n]} (\Phi(\boldsymbol{x}_i; \boldsymbol{w}, \gamma, \beta) - y_i)^2$.

Theorem 5.1. *In our setting of linear regression with BN, the sharpness-reduction flow $\boldsymbol{\zeta}$ defined in (3) converges to the solution $\boldsymbol{w}^* \in \mathbb{S}^{d-1}$ that minimizes sharpness $\lambda_1^{\mathrm{H}}(\boldsymbol{w}^*)$ on Γ, regardless of the initialization. Moreover, the coefficients $(\tilde{\boldsymbol{w}}, \tilde{b})$ associated with $\boldsymbol{w}^*$ (defined in (4)) are the optimal solution of the following constrained optimization problem (M):*

$$\min \quad \|\boldsymbol{w}\|_2^2 \quad \textit{s.t.} \quad \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i, \quad \forall i \in [n]. \tag{M}$$

At first sight the result may appear trivial because the intent of WD is to regularize L^2 norm. But this is deceptive because in scale-invariant nets WD changes from an explicit regularizer as an implicit one. This also challenges conventional view of optimization. GD is usually viewed as a discretization of its continuous counterpart, gradient flow (GF), and theoretical insight for the discrete update including convergence rate and implicit bias is achieved by analyzing the continuous counterpart (See Appendix A for a list). However, GF does not have the same sharpness-reduction bias as GD. As discussed in [37], adding WD only performs a time-rescaling on the GF trajectory on scale-invariant loss, but does not change the point that GF converge to if we project the trajectory onto the unit sphere. One can easily show that GF may converge to any zero-loss solution, but no matter how small LR is, GD exhibits the sharpness-reduction bias towards the optimal solution of (M). To our best knowledge, this result is the first concrete example where even with arbitrarily small LR, GD can still generalize better than GF under natural settings.

6 Conclusions and Future Work

We exhibited settings where gradient descent has an implicit bias to reduce spherical sharpness in training neural nets with normalization layers and weight decay, and we verified experimentally this sharpness-reduction bias predicted by our theorem as well as its generalization benefit on CIFAR-10.

Our theoretical analysis holds for dynamics around a minimizer manifold and requires the learning rate to be small enough so that the parameter oscillates locally and approximately tracks a sharpness-reduction flow on the manifold. We note that in practice a decrease in spherical sharpness is observed even with moderate LR and also before getting close to a minimizer manifold. Explaining these phenomena is left for future work. Now we list some other future directions. First is to generalize our results to SGD, where the sharpness measure may not be the spherical sharpness and could depend on the structure of gradient noise. Second, to understand benefit of reducing spherical sharpness on specific tasks, e.g., why does reducing spherical sharpness encourage low-rank on matrix completion with BN (Figure 1)? Third, to study sharpness-reduction bias for neural net architectures that are not scale-invariant on all parameters (eg., with certain unnormalized layers).

[3] Note that the batch size is n here as we are running full-batch GD

References

[1] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 7411–7422. Curran Associates, Inc., 2019.

[2] Sanjeev Arora, Zhiyuan Li, and Kaifeng Lyu. Theoretical analysis of auto rate-tuning by batch normalization. In *International Conference on Learning Representations*, 2019.

[3] Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on edge of stability in deep learning. *arXiv preprint*, 2022.

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[5] David Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Learning Representations*, 2021.

[6] Johan Bjorck, Carla Gomes, and Bart Selman. Understanding batch normalization. *arXiv preprint arXiv:1806.02375*, 2018.

[7] Guy Blanc, Neha Gupta, Gregory Valiant, and Paul Valiant. Implicit regularization for deep neural networks driven by an ornstein-uhlenbeck like process. 125:483–513, 09–12 Jul 2020.

[8] Lenaic Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on Learning Theory*, pages 1305–1338. PMLR, 2020.

[9] Jeremy Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021.

[10] Alex Damian, Tengyu Ma, and Jason D Lee. Label noise sgd provably prefers flat global minimizers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 27449–27461. Curran Associates, Inc., 2021.

[11] Hadi Daneshmand, Amir Joudaki, and Francis Bach. Batch normalization orthogonalizes representations in deep random networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[12] Soham De and Sam Smith. Batch normalization biases residual blocks towards the identity function in deep networks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19964–19975. Curran Associates, Inc., 2020.

[13] Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1019–1028. JMLR.org, 2017.

[14] Benjamin Fehrman, Benjamin Gess, and Arnulf Jentzen. Convergence rates for the stochastic gradient descent method for non-convex objective functions. *Journal of Machine Learning Research*, 21(136):1–48, 2020.

[15] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

[16] Rong Ge, Yunwei Ren, Xiang Wang, and Mo Zhou. Understanding deflation process in over-parametrized tensor decomposition. *Advances in Neural Information Processing Systems*, 34, 2021.

[17] Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9482–9491. Curran Associates, Inc., 2018.

[18] Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. *Advances in Neural Information Processing Systems*, 31, 2018.

[19] Haowei He, Gao Huang, and Yang Yuan. Asymmetric valleys: Beyond sharp and flat local minima. *Advances in neural information processing systems*, 32, 2019.

[20] 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*, pages 770–778, 2016.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 630–645, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46493-0.

[22] Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

[23] Elad Hoffer, Ron Banner, Itay Golan, and Daniel Soudry. Norm matters: efficient and accurate normalization schemes in deep networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[24] Hikaru Ibayashi and Masaaki Imaizumi. Exponential escape efficiency of sgd from sharp minima in non-stationary regime. *arXiv preprint arXiv:2111.04004*, 2021.

[25] Sergey Ioffe and Christian Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 448–456. JMLR.org, 2015.

[26] Stanisław Jastrzębski, 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.

[27] Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17176–17186. Curran Associates, Inc., 2020.

[28] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *European Conference on Machine Learning and Knowledge Discovery in Databases - Volume 9851*, ECML PKDD 2016, pages 795–811, Berlin, Heidelberg, 2016. Springer-Verlag. ISBN 9783319461274. doi: 10.1007/978-3-319-46128-1_50.

[29] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. 2017.

[30] Bobby Kleinberg, Yuanzhi Li, and Yang Yuan. An alternative view: When does SGD escape local minima? In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2698–2707. PMLR, 10–15 Jul 2018.

[31] Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5905–5914. PMLR, 18–24 Jul 2021.

[32] Susanna Lange, Kyle Helfrich, and Qiang Ye. Batch normalization preconditioning for neural network training. *arXiv preprint arXiv:2108.01110*, 2021.

[33] Jason D. Lee, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1246–1257, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

[34] Jason D Lee, Ioannis Panageas, Georgios Piliouras, Max Simchowitz, Michael I Jordan, and Benjamin Recht. First-order methods almost always avoid saddle points. *arXiv preprint arXiv:1710.07406*, 2017.

[35] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

[36] Yuanzhi Li, Tengyu Ma, and Hongyang Zhang. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 2–47. PMLR, 06–09 Jul 2018.

[37] Zhiyuan Li and Sanjeev Arora. An exponential learning rate schedule for deep learning. In *International Conference on Learning Representations*, 2020.

[38] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2020.

[39] Zhiyuan Li, Kaifeng Lyu, and Sanjeev Arora. Reconciling modern deep learning with traditional optimization analyses: The intrinsic learning rate. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 14544–14555. Curran Associates, Inc., 2020.

[40] Zhiyuan Li, Srinadh Bhojanapalli, Manzil Zaheer, Sashank J. Reddi, and Sanjiv Kumar. Robust training of neural networks using scale invariant architectures. *arXiv preprint arXiv:2202.00980*, 2022.

[41] Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss? –a mathematical framework. In *International Conference on Learning Representations*, 2022.

[42] Zinan Lin, Vyas Sekar, and Giulia Fanti. Why spectral normalization stabilizes gans: Analysis and improvements. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 9625–9638. Curran Associates, Inc., 2021.

[43] Ekaterina Lobacheva, Maxim Kodryan, Nadezhda Chirkova, Andrey Malinin, and Dmitry P. Vetrov. On the periodic behavior of neural network training with batch normalization and weight decay. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

[44] Ekdeep S Lubana, Robert Dick, and Hidenori Tanaka. Beyond batchnorm: Towards a unified understanding of normalization in deep learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 4778–4791. Curran Associates, Inc., 2021.

[45] Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. Towards understanding regularization in batch normalization. In *International Conference on Learning Representations*, 2019.

[46] Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. *arXiv preprint arXiv:1906.05890*, 2019.

[47] Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. In *International Conference on Learning Representations*, 2020.

[48] Kaifeng Lyu, Zhiyuan Li, Runzhe Wang, and Sanjeev Arora. Gradient descent on two-layer nets: Margin maximization and simplicity bias. *Advances in Neural Information Processing Systems*, 34, 2021.

[49] Chao Ma and Lexing Ying. On linear stability of SGD and input-smoothness of neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[50] Chao Ma and Lexing Ying. A riemannian mean field formulation for two-layer neural networks with batch normalization. *arXiv preprint arXiv:2110.08725*, 2021.

[51] David McAllester. Simplified pac-bayesian margin bounds. In *Learning theory and Kernel machines*, pages 203–215. Springer, 2003.

[52] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30, 2017.

[53] Ioannis Panageas and Georgios Piliouras. Gradient Descent Only Converges to Minimizers: Non-Isolated Critical Points and Invariant Regions. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:12, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.2.

[54] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[55] Akshay Rangamani, Nam H. Nguyen, Abhishek Kumar, Dzung Phan, Sang Peter Chin, and Trac D. Tran. A scale invariant measure of flatness for deep network minima. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1680–1684, 2021.

[56] Noam Razin and Nadav Cohen. Implicit regularization in deep learning may not be explainable by norms. *arXiv preprint arXiv:2005.06398*, 2020.

[57] Noam Razin, Asaf Maman, and Nadav Cohen. Implicit regularization in hierarchical tensor factorization and deep convolutional neural networks. *arXiv preprint arXiv:2201.11729*, 2022.

[58] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2483–2493. Curran Associates, Inc., 2018.

[59] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[60] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19 (70):1–57, 2018.

[61] Daniel Soudry, Elad Hoffer, and Nathan Srebro. The implicit bias of gradient descent on separable data. In *International Conference on Learning Representations*, 2018.

[62] Dominik Stöger and Mahdi Soltanolkotabi. Small random initialization is akin to spectral learning: Optimization and generalization guarantees for overparameterized low-rank matrix reconstruction. *Advances in Neural Information Processing Systems*, 34, 2021.

[63] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Normalized flat minima: Exploring scale invariant definition of flat minima for neural networks using PAC-Bayesian analysis. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9636–9647. PMLR, 13–18 Jul 2020.

[64] Twan van Laarhoven. L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*, 2017.

[65] Lei Wu, Zhanxing Zhu, et al. Towards understanding generalization of deep learning: Perspective of loss landscapes. *arXiv preprint arXiv:1706.10239*, 2017.

[66] Zeke Xie, Issei Sato, and Masashi Sugiyama. A diffusion theory for deep learning dynamics: Stochastic gradient descent exponentially favors flat minima. In *International Conference on Learning Representations*, 2021.

[67] Mingyang Yi, Qi Meng, Wei Chen, Zhi-ming Ma, and Tie-Yan Liu. Positively scale-invariant flatness of relu neural networks. *arXiv preprint arXiv:1903.02237*, 2019.

[68] Mingyang Yi, Huishuai Zhang, Wei Chen, Zhi-Ming Ma, and Tie-Yan Liu. Bn-invariant sharpness regularizes the training model to better generalization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 4164–4170. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

[69] Mingyang Yi, Qi Meng, Wei Chen, and Zhi-Ming Ma. Towards accelerating training of batch normalization: A manifold perspective. *arXiv preprint arXiv:2101.02916*, 2021.

[70] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

[71] Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. In *International Conference on Learning Representations*, 2019.

[72] Zhanxing Zhu, Jingfeng Wu, Bing Yu, Lei Wu, and Jinwen Ma. The anisotropic noise in stochastic gradient descent: Its behavior of escaping from sharp minima and regularization effects. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7654–7663. PMLR, 09–15 Jun 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Sections 4.2.1 and 6.

(c) Did you discuss any potential negative societal impacts of your work? [N/A] We are basically a theoretical work studying the generalization myestery in deep learning. We do not see any negative societal impact.

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [Yes] The assumptions of our main theorem for GD+WD on scale-invariant loss are stated in Section 4.2.1. For GD/PGD with Quasi-RMSprop schedulers in general, see Appendix B.4.

(b) Did you include complete proofs of all theoretical results? [Yes] See Appendices G and H for the proofs for our main theorems, Appendix O for the proof for the linear example.

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See our supplementary material.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix P.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Most of our experiments have only run once due to computational constraints, but we verified the sharpness-reduction bias across various settings of hyperparameters.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix P.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [N/A]

(b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]