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" is carefully defined given that the loss is scale-invariant, a known consequence of normalization. Specifically, for a fairly broad class of neural nets with normalization, our theory explains how GD with a finite learning rate enters the so-called Edge of Stability (EoS) regime, and characterizes the trajectory of GD in this regime via a continuous sharpness-reduction flow.

1 Introduction

Training modern deep neural nets crucially relies on normalization layers to make the training process less sensitive to hyperparameters and initialization. Popular normalization layers include Batch Normalization (BN) [43], Layer Normalization [9], etc. Normalization layers amount to a reparametrization of the neural net so that the loss becomes invariant to the scale of most parameters (and with a minor change, to *all* parameters): $\mathcal{L}(c\boldsymbol{w}) = \mathcal{L}(\boldsymbol{w})$ for all scalings $c > 0$ [43, 7, 61]. The current paper assumes this scale-invariance for all parameters and analyzes the trajectory of gradient descent with *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 WD 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 $\nabla^2\mathcal{L}(\boldsymbol{w}) = \frac{1}{\|\boldsymbol{w}\|_2^2}\nabla^2\mathcal{L}(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2})$, so 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 [7, 61, 64]. More notably, such a change in dynamics also yields a very different solution at the end — in particular a solution that generalizes better [102].

The goal of the current paper is to improve mathematical understanding of how normalization together with accompanying WD can improve generalization. While this may arise from many places, we focus on sharpness-based generalization measures and exhibit settings where gradient descent persistently reduces sharpness in training normalized nets with WD, 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.

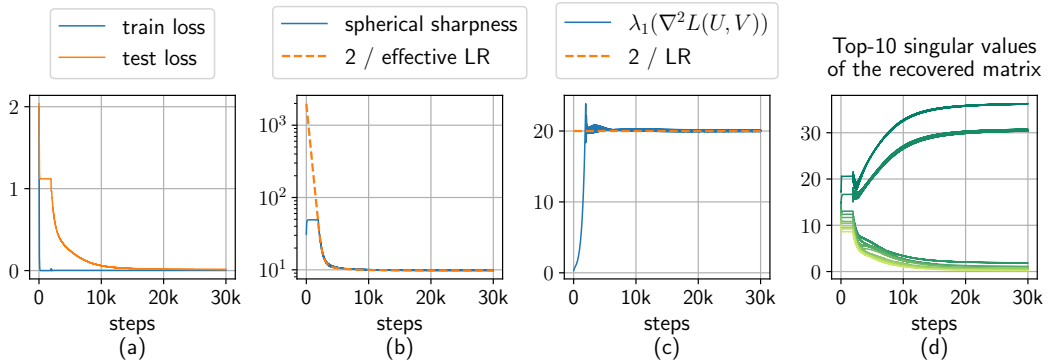


Figure 1: Experiment on overparameterized matrix completion with Batch Normalization. Given 800 (32%) entries Ω of a rank-2 matrix $\boldsymbol{M} \in \mathbb{R}^{50\times 50}$, use GD+WD to optimize the loss $\mathcal{L}(\boldsymbol{U},\boldsymbol{V}) := \frac{1}{|\Omega|}\sum_{(i,j)\in\Omega}(\mathrm{BN}([\boldsymbol{U}\boldsymbol{V}^\top]_{i,j}) - M_{i,j})^2$, where $\boldsymbol{U},\boldsymbol{V} \in \mathbb{R}^{50\times 50}$ (thus no explicit constraint on rank). Starting from step ~ 2000, spherical sharpness drops significantly (**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 also Appendix P.1.

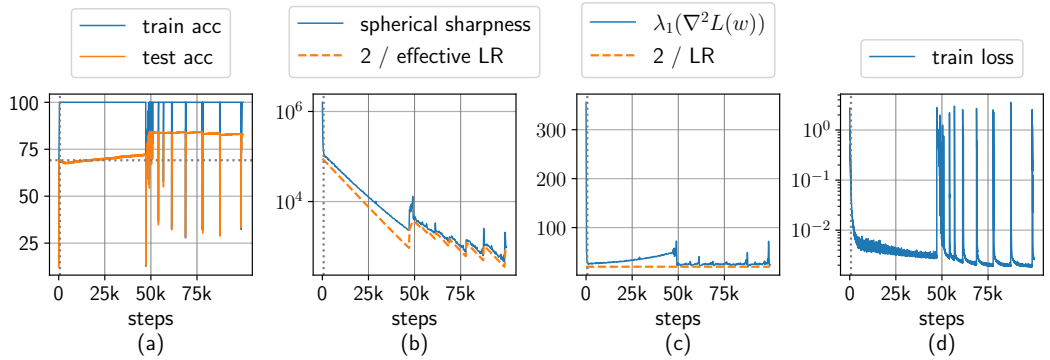


Figure 2: In training a smooth and scale-invariant VGG-11 on CIFAR-10 with (full-batch) GD+WD, the spherical sharpness keeps decreasing and the test accuracy keeps increasing. BN is added after every linear layer to ensure scale-invariance. 100% training accuracy is achieved after ~ 680 steps (dotted 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%. Removing either of normalization or WD eliminates this phenomenon; see Appendices P.4 and P.5.

It is long believed that flatter minima generalize better [39, 49, 78], but the notion of sharpness/flatness makes sense only if it is carefully defined in consideration of various symmetries in neural nets. One of the most straightforward measures of sharpness is 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 [22]. Also, this sharpness measure may not decrease with the number of training steps: an empirical study by Cohen et al. [16] shows that for various neural nets (including normalized nets), GD has an overwhelming tendency to persistently increase $\lambda_1(\nabla^2\mathcal{L}(\boldsymbol{w}_t))$ until it reaches the *Edge of Stability (EoS) regime*, a regime where $\lambda_1(\nabla^2\mathcal{L}(\boldsymbol{w}_t))$ stabilizes around $2/\hat{\eta}$ ($\hat{\eta}$ is the learning rate). See also Figure 2c.

The sharpness measure we use in this paper takes the scale-invariance property into account. We note that techniques from previous works [75, 78, 27] 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 $\lambda_1(\nabla^2\mathcal{L}(\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2}))$, 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 with weight decay (GD+WD), assuming the loss $\mathcal{L}$ is scale-invariant due to the presence of normalization. For

constant learning rate $\hat{\eta}$ and weight decay $\hat{\lambda}$, we can rewrite gradient descent (1) on scale-invariant loss equivalently as Projected Gradient Descent (PGD) on the unit sphere with adaptive learning rates, $\boldsymbol{\theta}_{t+1} \leftarrow \Pi(\boldsymbol{\theta}_t - \tilde{\eta}_t \nabla \mathcal{L}(\boldsymbol{\theta}_t))$, 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). We call $\tilde{\eta}_t$ adaptive because it can be shown that $\tilde{\eta}_t$ increases when gradient is small and decreases when gradient is large, which resembles the behaviors of adaptive gradient methods such as RMSprop [38]. Our main contributions are as follows:

1. We theoretically show that once $\boldsymbol{\theta}_t$ reaches a point near the manifold of minimizers of $\mathcal{L}$, the effective learning rate $\tilde{\eta}_t$ will keep increasing until $2/\tilde{\eta}_t$ roughly equals to the spherical sharpness, or in other words, gradient descent enters the EoS regime (Section 4.1).
2. In the EoS regime, we show that for gradient descent with a small (but finite) learning rate, $\boldsymbol{\theta}_t$ oscillates around the manifold and moves approximately along a sharpness-reduction flow, which is a gradient flow for minimizing spherical 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 surprisingly 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 benefits to generalization on CIFAR-10 with VGG-11 and ResNet-20, as well as matrix completion with BN (Appendix P).
5. We generalize our theoretical results 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 [87, 86, 70, 47, 32, 31, 60, 82, 4, 14, 63, 71, 83, 88, 28] where the same bias exists at infinitesimal LR. Our analysis is inspired by a previous line of works [12, 17, 65] 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 [12, 17, 65], or using non-standard variants of gradient descent update rule, e.g., using normalized GD or non-smooth wrappings on the loss function [8].

2 Related Works

Flatness and Generalization. It has been long believed that minima locate in a flat valley generalize better [39]. Li et al. [58] first visualized the loss landscape of neural networks and found "sharp" minima generalizes worse. Keskar et al. [49], Wu et al. [94], Jastrzębski et al. [45] empirically verified the positive correlation between flatness and generalization. Neyshabur et al. [78] gave a theoretical explanation for generalization benefit of low sharpness using PAC-Bayes theory [75]. In light of this, Foret et al. [27] 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 [22], as networks with different sharpness may represent the same function. Towards resolving this challenge, multiple definitions of scale-invariant sharpness have been proposed [99, 100, 91, 81]. Kwon et al. [52] derived new algorithms with better generalization by defining new sharpness notions that are aware of positive homogeneity and invariance. He et al. [34] argued that the local minima of modern deep networks are more than being flat or sharp and could be asymmetric valleys. The theoretical implication of flatness has also been explored specifically for two-layer nets [12, 77, 33, 65, 21] and deep linear nets [76].

Understanding Normalization Layers. Ioffe and Szegedy [43] proposed BN with the original motivation to reduce Internal Covariate Shift (ICS), but Santurkar et al. [84] challenged this view by arguing that the effectiveness of BN comes from a smoothing effect on the training objective. A common feature of normalization layers (including BN, LN [9], GN [96]) is that they make the loss invariant to the scale of layer weights. Based on this, several existing works have reported that training neural nets with normalization and weight decay can go out of the scope of the classical optimization theory. Specifically, Li and Arora [61] showed that one can train the net to small loss even with learning rates exponentially increasing; Li et al. [62] showed that the dynamic of (full-batch) GD may leave the stable regime; Lobacheva et al. [67] empirically showed that the training loss can exhibit a periodic behavior that sometimes improves the test accuracy. However, despite a lot of effort devoted to understanding the optimization and generalization benefits of normalization in various specific settings, *e.g.*, [92, 11, 102, 69, 20, 66, 18, 19, 68, 53, 73], 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 [92, 41, 7] 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:*

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

*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. [64]. Here we present a variant of their theorem that bounds both the gradient and effective LR.

Theorem 3.2 (Variant of Theorem D.2 in [64])**.** *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 $\tilde{\eta}_0 \le \frac{1}{\pi^2\rho_2(1-\hat{\eta}\hat{\lambda})}$, let $T_0 := \left\lceil \frac{1}{2\hat{\eta}\hat{\lambda}} \ln \frac{\|\boldsymbol{w}_0\|_2^2}{\rho_2\pi^2\hat{\eta}} \right\rceil$ 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 $\tilde{\eta}_t \le \frac{2}{\pi^2\rho_2(1-\hat{\eta}\hat{\lambda})}$.*

4 GD+WD on Scale-Invariant Loss Functions

This section analyzes GD+WD (1) on a scale-invariant loss $\mathcal{L}(\boldsymbol{w})$, in particular what happens after the approximate convergence of Theorem 3.2. 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 $\mathbb{S}^{D-1}$. Section 4.1 analyzes the dynamics in the stable regime, where loss is guaranteed to decrease monotonically, and Theorem 3.2 suggests $\boldsymbol{w}_t$ can get close to a local minimizer at some time t_0. We show that the effective LR keeps increasing after t_0, causing GD+WD to eventually leave this stable regime and enter 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 of 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 $\mathcal{L}(\boldsymbol{\theta}_t)$ keeps 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 difference, then we call it as the *Edge of Stability (EoS) regime*. This EoS regime is conceptually the same as that defined by Cohen et al. [16]; see Appendix G.3 for discussion.

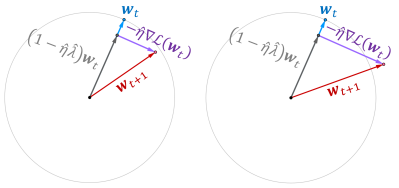
Fix an initial point $\boldsymbol{w}_0 \in \mathbb{R}^D \setminus \{\boldsymbol{0}\}$. Now we aim to characterize the dynamics of GD+WD when LR $\hat{\eta}$ and WD $\hat{\lambda}$ are small enough. Theorem 3.2 shows that for some $t_0 \le T_0$, $\|\nabla\mathcal{L}(\boldsymbol{\theta}_{t_0})\|_2^2 \le O(\hat{\lambda}\hat{\eta})$ 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 [55, 79, 56, 101]. 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 $\|\boldsymbol{\theta}_{t_0} - \boldsymbol{\theta}^*\|_2 \le O((\hat{\lambda}\hat{\eta})^{1/2})$ (assuming smoothness, the latter implies the former).

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 $\tilde{\eta}_t \approx 2/\lambda_{\max}^{(t)}$.

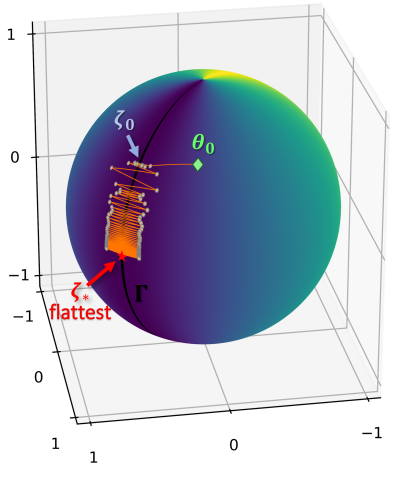
To establish Theorem 4.3, we need to assume that $\mathcal{L}$ satisfies Polyak-Łojasiewicz (PL) condition locally, which is a standard regularity condition in the optimization literature to ease theoretical analysis around a minimizer. Intuitively, PL condition guarantees that the gradient grows faster 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., [48]).

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 $\mathcal{C}^2$-smooth scale-invariant loss that satisfies μ-PL around a local minimizer $\boldsymbol{\theta}^*$ on the unit sphere, and $\rho_2 := \sup\{\|\nabla^2\mathcal{L}(\boldsymbol{w})\|_2 : \boldsymbol{w} \in \mathbb{S}^{D-1}\}$. 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 $\tilde{\eta}_{t_0} \le \frac{2}{\rho_2} < \frac{2}{\lambda_1^{\mathrm{H}}(\boldsymbol{\theta}^*)}$, and if $\hat{\lambda}\hat{\eta}$ is small enough, 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 $\tilde{\eta}_{t_1} = \frac{2}{\lambda_1^{\mathrm{H}}(\boldsymbol{\theta}^*)} + O((\hat{\lambda}\hat{\eta})^{1/2})$.*



(a) (b)



(c)

Figure 3: **(a), (b)**: The norm of $\boldsymbol{w}_t$ decreases when gradient is small and increases when gradient is large. **(c)**: The trajectory of $\boldsymbol{\theta}_t$ on a 3D scale-invariant loss function. Darker color means lower loss on the unit sphere, and points in the black line are minimizers (see Appendix F). In the end, $\boldsymbol{\theta}_t$ approaches 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 $\mathcal{L} : \mathbb{R}^3 \setminus \{\mathbf{0}\} \to \mathbb{R}$, where the black line is a manifold Γ consisting of all the minimizers. In training with GD+WD, $\boldsymbol{\theta}_t$ first goes close to a local minimizer $\boldsymbol{\zeta}_0$, then Theorem 4.3 suggests that WD causes the effective LR to steadily increase until the dynamic enters the EoS regime. 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 actually forms a small angle that pushes $\boldsymbol{\theta}_t$ to move downward persistently until $\boldsymbol{\theta}_t$ approaches the minimizer $\boldsymbol{\zeta}_*$ denoted in the plot.

For a general scale-invariant loss $\mathcal{L} : \mathbb{R}^D \setminus \{\mathbf{0}\} \to \mathbb{R}$, which minimizer does $\boldsymbol{\theta}_t$ move towards? In this work, we consider the setting where there is a manifold Γ consisting only of local minimizers (but not necessarily all of them). We show that $\boldsymbol{\theta}_t$ always oscillates around the manifold once it approaches the manifold and enters the EoS regime, and meanwhile $\boldsymbol{\theta}_t$ keeps moving in a direction of reducing spherical sharpness.

4.2.1 Assumptions

Now we formally introduce our main assumption on the local minimizer manifold Γ.

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 $\mathrm{rank}(\boldsymbol{H}(\boldsymbol{\theta})) = D - D_\Gamma$.

Scale-invariance has become a standard assumption in studying neural nets with normalization layers [61, 62, 67]. For VGG and ResNet, the scale-invariance can be ensured after making minor changes to the architectures (see Appendix Q.1). The training loss $\mathcal{L}$ may not be smooth if the activation is ReLU, but lately it has become clear that differentiable activations such as Swish [80], GeLU [37] can perform equally well. Swish is indeed used in our VGG-11 experiments (Figure 2), but ResNet with ReLU activation also exhibits a sharpness-reduction bias empirically (see Appendix P.2).

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 $\mathrm{rank}(\boldsymbol{H}(\boldsymbol{\theta})) = D - D_\Gamma$ ensures that the Hessian is maximally non-degenerate on Γ, which also appears as a key assumption in previous works [65, 8, 25]. 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. [8], 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. [62]) 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). We connect GD+WD in the EoS regime to the following gradient flow (3) on the manifold Γ minimizing spherical sharpness (with gradient-dependent learning rate), and show that 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] Alternatively, 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.

Initialization Scheme. Given a local minimizer $\boldsymbol{\zeta}_0 \in \Gamma$, we initialize $\boldsymbol{w}_0 \in \mathbb{R}^D \setminus \{\boldsymbol{0}\}$ as follows: draw $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \sigma_0^2 \boldsymbol{I}/D)$ from Gaussian and set the direction of $\boldsymbol{w}_0$ to $\frac{\boldsymbol{\zeta}_0+\boldsymbol{\xi}}{\|\boldsymbol{\zeta}_0+\boldsymbol{\xi}\|_2}$, where σ_0 can take any value in $[\exp(-\eta_{\text{in}}^{-o(1)}), \eta_{\text{in}}^{1/2-o(1)}]$; then set the parameter norm $\|\boldsymbol{w}_0\|_2$ to be any value that satisfies $\left|\tilde{\eta}_0 - \frac{2}{\lambda_1^{\text{H}}(\boldsymbol{\zeta}_0)}\right| \le \eta_{\text{in}}^{1/2-o(1)}$, where $\tilde{\eta}_0 := \frac{\hat{\eta}}{(1-\hat{\eta}\hat{\lambda})\|\boldsymbol{w}_0\|_2^2}$ is the effective LR for the first step.

Theorem 4.6. *Under Assumptions 4.4 and 4.5, for GD+WD (1) with sufficiently small intrinsic learning rate $\eta_{\text{in}} := \hat{\eta}\hat{\lambda}$, if we follow the above initialization scheme for some $\boldsymbol{\zeta}_0 \in \Gamma$, then with probability $1 - O(\eta_{\text{in}}^{1/2-o(1)})$, the trajectory of $\boldsymbol{\theta}_t := \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|_2}$ approximately tracks a sharpness-reduction flow $\boldsymbol{\zeta} : [0, T] \to \Gamma$ that starts from $\boldsymbol{\zeta}_0$ and evolves as the ODE (3) up to time T (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 according to our analysis, the magnitude of oscillation is as small as $O(\eta_{\text{in}}^{1/2-o(1)})$, so it is absorbed into our final bound $O(\eta_{\text{in}}^{1/4-o(1)})$ for the distance between $\boldsymbol{\theta}_t$ and $\boldsymbol{\zeta}(t\eta_{\text{in}})$.

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 $\Pi : \mathbb{R}^D \setminus \{\boldsymbol{0}\} \to \mathbb{S}^{D-1}$. 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 to power method, we show that $\boldsymbol{x}_t$ quickly aligns 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]

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

[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. As mentioned later, this need for random initialization is very similar to the one needed in power method for computing eigenvalues.

[2] Our construction of $\boldsymbol{\phi}_t$ ensures that $\boldsymbol{x}_t$ only 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 $\pm\boldsymbol{v}_1^{\text{H}}(\boldsymbol{\phi}_t)$.

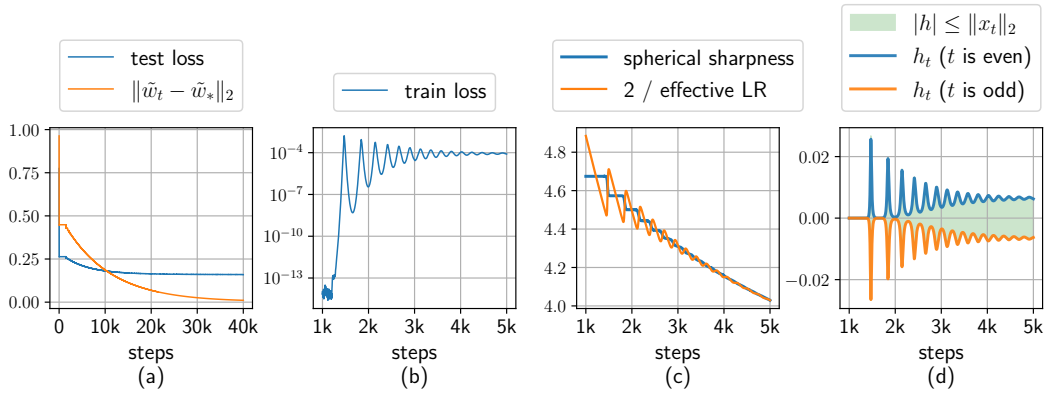


Figure 4: Illustration of the oscillation and periodic behaviors of GD+WD on linear regression with BN (see Sections 4.2.3 and 5). The training loss decreases to $\approx 10^{-14}$ in the first 1k steps and achieves test loss 0.26. Starting from step $\sim$ 1k, the dynamic enters the EoS regime. **(a).** The test loss decreases to 0.16 as a distance measure to the flattest solution (M) decreases towards 0; **(b).** The training loss oscillates around $\sim 10^{-4}$ in the EoS regime; **(c).** $2/\tilde{\eta}_t$ switches back and forth between being smaller and larger than $\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$; **(d).** The parameter oscillates around the minimizer manifold along the top eigenvector direction, and the magnitude of oscillation $|h_t|$ rises and falls periodically.

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 $\pm \boldsymbol{v}_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$ by using 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, $\boldsymbol{\phi}_{t+2} \approx \boldsymbol{\phi}_t - 4h_t^2 \nabla_\Gamma \log \lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t) + O(\eta_{\mathrm{in}}^{1.5-o(1)})$, which resembles two steps of gradient descent on Γ to minimize the logarithm of spherical sharpness with learning rate $2h_t^2$,

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. [67], which reveals a periodic behavior of gradients and effective learning rates in training normalized nets with weight decay. In our theoretical setting, we capture this periodic behavior by showing that h_t and $\tilde{\eta}_t$ *do* 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), and in our case the gradient norm scales as $|h_t|$ since $\nabla \mathcal{L}(\boldsymbol{\theta}_t) \approx h_t \lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t) \boldsymbol{v}_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$. 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 $|h_t|$ cannot decrease forever, since $\tilde{\eta}_t$ increases when $|h_t|$ is sufficiently small. When $\tilde{\eta}_t$ rises to over $2/\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$, $|h_t|$ changes from decreasing to increasing according to our approximation. But h_t cannot increase indefinitely either, since $\tilde{\eta}_t$ decreases when $|h_t|$ is sufficiently large. A period is finally finished when $\tilde{\eta}_t$ drops below $2/\lambda_1^{\mathrm{H}}(\boldsymbol{\phi}_t)$.

In our theoretical analysis, we connect this periodic behavior with a 1-dimensional Hamiltonian system (see Appendix H.2), and show that $2h_t^2$ in each step can be approximated by its average value in the period without incurring a large error. Further calculations show that this average value is approximately $\frac{2\eta_{\mathrm{in}}}{4+\|\nabla_\Gamma \log \lambda_1^{\mathrm{H}}(\boldsymbol{\zeta}(t\eta_{\mathrm{in}}))\|_2}$, the learning rate in the flow (3) multiplied with η_{in}. We can therefore conclude that each step of $\boldsymbol{\phi}_t$ (or $\boldsymbol{\theta}_t$) tracks a time interval of η_{in} in the flow.

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 LR in each step, and Quasi-RMSprop is a specific class of schedulers we define, including the way that the effective LR changes 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 LR changes in each step 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 the LR changes as a Quasi-RMSprop 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 to 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 [61], 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 applies to dynamics around a minimizer manifold and requires a small (but finite) learning rate so that we can show that the parameter oscillates locally and approximately tracks a sharpness-reduction flow. We note that in practice a decrease in spherical sharpness is observed even with moderate LR and even before getting close to a minimizer manifold. Explaining these phenomena is left for future work. Now we list some other future directions. The 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 the 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 (e.g., with certain unnormalized layers).

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

References

[1] Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 247–257. PMLR, 17–23 Jul 2022.

[2] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in over-parameterized neural networks, going beyond two layers. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[3] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. 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 242–252. PMLR, 09–15 Jun 2019.

[4] 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.

[5] Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pages 322–332. PMLR, 2019.

[6] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. 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 8139–8148. Curran Associates, Inc., 2019.

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

[8] Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on the edge of stability in deep learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 948–1024. PMLR, 17–23 Jul 2022.

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

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

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

[12] 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.

[13] Yuejie Chi, Yue M. Lu, and Yuxin Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269, 2019. doi: 10.1109/TSP.2019.2937282.

[14] 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.

[15] Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. 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 2937–2947. Curran Associates, Inc., 2019.

[16] 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.

[17] 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.

[18] Hadi Daneshmand, Jonas Kohler, Francis Bach, Thomas Hofmann, and Aurelien Lucchi. Batch normalization provably avoids ranks collapse for randomly initialised 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 18387–18398. Curran Associates, Inc., 2020.

[19] 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.

[20] 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.

[21] Lijun Ding, Dmitriy Drusvyatskiy, and Maryam Fazel. Flat minima generalize for low-rank matrix recovery. *arXiv preprint arXiv:2203.03756*, 2022.

[22] 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.

[23] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[24] K. J. Falconer. Differentiation of the Limit Mapping in a Dynamical System. *Journal of the London Mathematical Society*, s2-27(2):356–372, 04 1983. ISSN 0024-6107. doi: 10.1112/jlms/s2-27.2.356.

[25] 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.

[26] Robert L. Foote. Shorter notes: Regularity of the distance function. *Proceedings of the American Mathematical Society*, 92(1):153–155, 1984. ISSN 00029939, 10886826.

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

[28] 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.

[29] Justin Gilmer, Behrooz Ghorbani, Ankush Garg, Sneha Kudugunta, Behnam Neyshabur, David Cardoze, George Edward Dahl, Zachary Nado, and Orhan Firat. A loss curvature perspective on training instabilities of deep learning models. In *International Conference on Learning Representations*, 2022.

[30] Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6151–6159. Curran Associates, Inc., 2017.

[31] 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.

[32] 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.

[33] Jeff Z. HaoChen, Colin Wei, Jason Lee, and Tengyu Ma. Shape matters: Understanding the implicit bias of the noise covariance. In Mikhail Belkin and Samory Kpotufe, editors, *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 2315–2357. PMLR, 15–19 Aug 2021.

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

[35] 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.

[36] 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.

[37] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[38] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a: Overview of mini-batch gradient descent. Technical report, 2012. URL https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.

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

[40] Elad Hoffer, Itay Hubara, and Daniel Soudry. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[41] 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.

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

[43] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456, Lille, France, 07–09 Jul 2015. PMLR.

[44] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural 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 8571–8580. Curran Associates, Inc., 2018.

[45] 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.

[46] Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho*, and Krzysztof Geras*. The break-even point on optimization trajectories of deep neural networks. In *International Conference on Learning Representations*, 2020.

[47] 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.

[48] 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.

[49] 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. In *International Conference on Learning Representations*, 2017.

[50] 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.

[51] Lingkai Kong and Molei Tao. Stochasticity of deterministic gradient descent: Large learning rate for multiscale objective function. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 2625–2638. Curran Associates, Inc., 2020.

[52] 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.

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

[54] Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pages 1302–1338, 2000.

[55] 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.

[56] 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.

[57] Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.02218*, 2020.

[58] 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.

[59] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *arXiv preprint arXiv:1808.01204*, 2018.

[60] 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.

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

[62] 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.

[63] 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*, 2021.

[64] Zhiyuan Li, Srinadh Bhojanapalli, Manzil Zaheer, Sashank Reddi, and Sanjiv Kumar. Robust training of neural networks using scale invariant architectures. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 12656–12684. PMLR, 17–23 Jul 2022.

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

[66] 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.

[67] 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.

[68] 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.

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

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

[71] 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.

[72] 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.

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

[74] Chao Ma, Lei Wu, and Lexing Ying. The multiscale structure of neural network loss functions: The effect on optimization and origin. *arXiv preprint arXiv:2204.11326*, 2022.

[75] David McAllester. Simplified PAC-Bayesian margin bounds. In *Learning theory and Kernel machines*, pages 203–215. Springer, 2003.

[76] Rotem Mulayoff and Tomer Michaeli. Unique properties of flat minima in deep networks. 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 7108–7118. PMLR, 13–18 Jul 2020.

[77] Rotem Mulayoff, Tomer Michaeli, and Daniel Soudry. The implicit bias of minima stability: A view from function space. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 17749–17761. Curran Associates, Inc., 2021.

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

[79] 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.

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

[81] 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.

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

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

[84] 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.

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

[86] 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.

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

[88] 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.

[89] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[90] Hidenori Tanaka and Daniel Kunin. Noether's learning dynamics: Role of symmetry breaking in neural networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 25646–25660. Curran Associates, Inc., 2021.

[91] 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.

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

[93] Yuqing Wang, Minshuo Chen, Tuo Zhao, and Molei Tao. Large learning rate tames homogeneity: Convergence and balancing effect. In *International Conference on Learning Representations*, 2022.

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

[95] Lei Wu, Chao Ma, and Weinan E. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. 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.

[96] Yuxin Wu and Kaiming He. Group normalization. *arXiv preprint arXiv:1803.08494*, 2018.

[97] 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.

[98] Greg Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019.

[99] 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.

[100] 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.

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

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

[103] 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.

[104] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep relu networks. *arXiv preprint arXiv:1811.08888*, 2018.

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.3.

(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 Q.

(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.

(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 Q.

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]