No Wrong Turns: The Simple Geometry Of Neural Networks Optimization Paths

Charles Guille-Escuret [*1 2] **Hiroki Naganuma** [*1 2] **Kilian Fatras** [1 3 4] **Ioannis Mitliagkas** [1 2 5]

Abstract

Understanding the optimization dynamics of neural networks is necessary for closing the gap between theory and practice. Stochastic first-order optimization algorithms are known to efficiently locate favorable minima in deep neural networks. This efficiency, however, contrasts with the non-convex and seemingly complex structure of neural loss landscapes. In this study, we delve into the fundamental geometric properties of sampled gradients along optimization paths. We focus on two key quantities, the restricted secant inequality and error bound, as well as their ratio γ, which hold high significance for first-order optimization. Our analysis reveals that these quantities exhibit predictable, consistent behavior throughout training, despite the stochasticity induced by sampling minibatches. Our findings suggest that not only do optimization trajectories never encounter significant obstacles, but they also maintain stable dynamics during the majority of training. These observed properties are sufficiently expressive to theoretically guarantee linear convergence and prescribe learning rate schedules mirroring empirical practices. We conduct our experiments on image classification, semantic segmentation and language modeling across different batch sizes, network architectures, datasets, optimizers, and initialization seeds. We discuss the impact of each factor. Our work provides novel insights into the properties of neural network loss functions, and opens the door to theoretical frameworks more relevant to prevalent practice.

1. Introduction

Despite the theoretical complexity of their loss landscapes, deep neural networks have demonstrated remarkable empirical reliability across a broad range of applications. Blum & Rivest (1992) proved decades ago that neural network training is NP-hard. The intricacy of their loss functions, especially the non-convexity implying potential bad local minima and saddle points, has led to an enduring conundrum concerning the empirical efficiency of stochastic first-order optimization methods for training neural networks.

Numerous studies have strived to reconcile this apparent contradiction, focusing on the behaviors of stochastic gradient descent (SGD) and its variants at local minima and saddle points (Panageas et al., 2019; Jin et al., 2019). The central hypothesis in these works posits that the efficiency of training arises from the ability of these algorithms to navigate complex loss landscapes adeptly and manage non-convexity.

Conversely, other investigations have empirically found loss landscapes to be simpler than their theoretical complexity might suggest (Lucas et al., 2021). Notably, Goodfellow et al. (2015) observed that "in fact, on a straight path from initialization to solution, a variety of state of the art neural networks never encounter any significant obstacles."

Notwithstanding, our current understanding of how neural loss landscapes are empirically simpler than expected remains quite limited. There is yet to emerge a robust mathematical characterization of this empirical simplicity. Consequently, we contend that the theoretical assumptions currently in use fail to accurately capture the objective functions typical in deep learning. This discrepancy is a significant barrier to applying theoretical insights effectively in the optimization of neural networks.

One such common assumption, smoothness, is illustrative

[*]Equal contribution [1]Mila, Montreal, Canada [2]Université de Montréal Montreal, Canada [3]University of McGill, Montreal, Canada [4]Dreamfold [5]Archimedes Unit, Athena Research Center, Athens. Correspondence to: Charles Guille-Escuret <guillech@mila.quebec>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

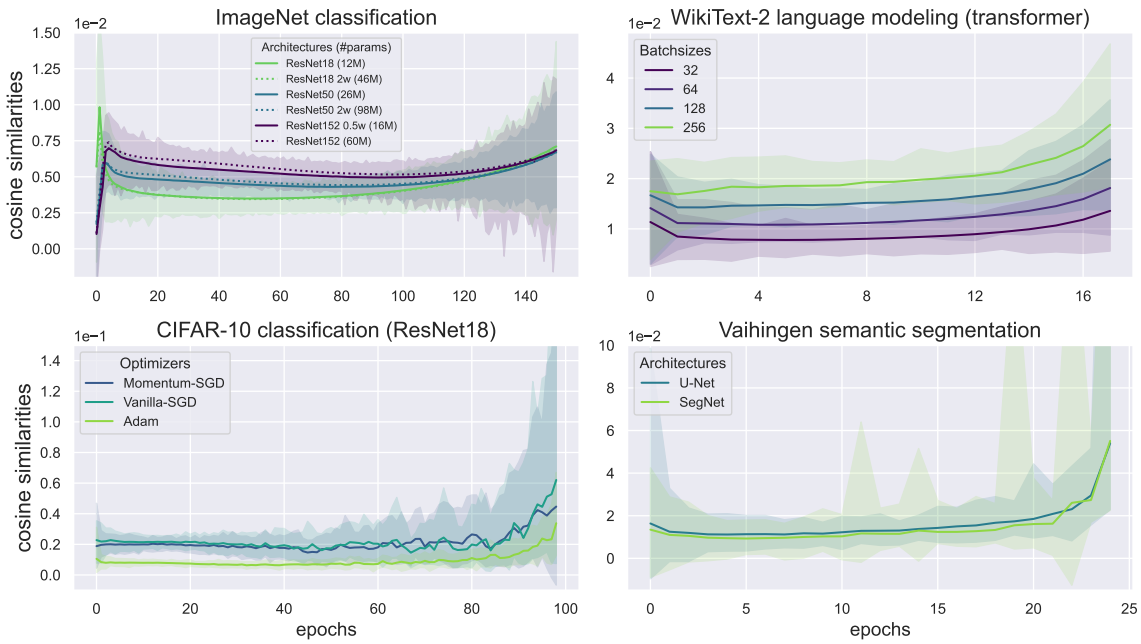

Figure 1. Cosine similarities between the gradients G_t sampled at step t and the difference $(w_t - w_T)$ between current weights and final weights, averaged over each epoch. The shaded regions denote the range from minimum to maximum values observed at each epoch. The results are presented for a selection of scenarios: (top left) varying depths and widths of ResNet on ImageNet, (top right) different batch sizes on WikiText-2 using a Transformer, (bottom left) a range of optimizers on CIFAR-10 using ResNet-18, and (bottom right) distinct architectures on Vaihingen semantic segmentation. This figure highlights the stability of the cosine similarity throughout most of training, suggesting it as a fundamental characteristic of neural network training.

of this gap. Despite its popularity, smoothness is encumbered by several limitations: it is computationally intensive to approximate for large neural networks, and necessitates additional assumptions such as bounded gradients for theoretical guarantees in stochastic settings (Qian et al., 2019; Shamir & Zhang, 2013) although recent works have tried to discard them (Nguyen et al., 2018; Loizou et al., 2021). Finally, recent findings suggesting certain directional sharpness in neural networks (Dinh et al., 2017) call into question the suitability of smoothness as a measure of their simplicity.

To address these issues, our study undertakes an empirical analysis of the geometric properties of the loss function in regions traversed by first-order optimization algorithms. Our focus is on a variant of the quantities involved in the Restricted Secant Inequality (RSI) (Zhang & Yin, 2013) and Error Bound (EB) (Luo & Tseng, 1993), which pertain to the relationship between sampled gradients, current iterate, and final iterate of the optimization sequence. Our findings indicate that these quantities and their ratio exhibit stable, predictable patterns throughout training across diverse settings, thereby quantitatively characterizing the simplicity of neural loss landscape geometry. Furthermore, these quantities offer several advantages over smoothness, including efficient estimations post-training, inherent compatibility with stochasticity due to direct measurement on sampled gradients, and a well-behaved empirical nature that still allows the derivation of theoretical results such as linear convergence or the prescription of learning rate schedules.

Our key contributions are as follows:

- We devise an experimental procedure for examining the geometry of optimization paths on common architectures. We assume almost-everywhere differentiability, but not smoothness.
- We execute experiments across a range of realistic deep learning settings, identifying consistently verified properties. For instance, the cosine similarity between the negative stochastic gradient and the direction to the final iterate is almost always positive and exhibits remarkable stability across iterations and epochs.
- We demonstrate how our empirical investigations can inform the prescription of learning rate schedules, aligning with established empirical knowledge.
- We provide an extensive discussion on the implications and limitations of our findings.

Collectively, our work quantifies crucial geometric properties of stochastic gradients along deep learning optimization paths, underlining their importance in understanding neural network optimization and enhancing current methodologies.

2. Related Work

Our investigation centers on the application of RSI and EB to enhance our comprehension of the geometric principles governing neural network optimization. Consequently, our work intersects with previous research on neural loss landscapes and the utilization of RSI and EB in optimization.

RSI and EB: The study of RSI and EB for first-order optimization is not new. RSI (Zhang & Yin, 2013) has been applied in numerous theoretical works (Yi et al., 2019; Schöpfer, 2016; Yuan et al., 2016; Karimi et al., 2016). EB (Luo & Tseng, 1993) has seen less extensive study (Dmitriy Drusvyatskiy, 2018), possibly due to the dominance of smoothness —a condition stronger than EB— in the field. It should not be confused with error bounds on the distance to a set, a term also prevalent in optimization literature (Qian et al., 2023; Zhou & So, 2015). Both RSI and EB, along with other conditions, were analyzed in Guille-Escuret et al. (2021). Furthermore, it was demonstrated in Guille-Escuret et al. (2022) that gradient descent is optimal for the class of functions defined by this pair of conditions.

Neural Loss Landscape Geometry: The intricacies of neural loss landscapes have been a focal point of research since the emergence of deep learning. Efforts have ranged from loss landscape visualizations (Li et al., 2018) to investigations of low loss basin connectivity (Garipov et al., 2018) and linear mode connectivity (Nagarajan & Kolter, 2019; Frankle et al., 2020). While prior research has noted the seeming simplicity of loss landscape geometry along optimization paths (Lucas et al., 2021; Goodfellow et al., 2015), these observations often involve straightforward phenomena such as monotonic decrease along linear interpolations. Our work takes this approach a step further by studying quantifiable properties with theoretical implications. Additionally, others have examined the geometric properties of neural loss landscapes in the near-infinite width, or Neural Tangent Kernel (NTK), regime (Jacot et al., 2018; Lee et al., 2019). These studies suggest that neural network training can be approximated by linear dynamics or that the loss surface adheres to the Polyak-Łojasiewicz condition (Liu et al., 2022). Concurrently to our work, Liu et al. (2023) showed that the Aiming condition, which is related to RSI, is theoretically verified for sufficiently large width, which also corresponds to the NTK regime. Unfortunately, this scenario was found to be irrealistic in empirical settings (Chizat et al., 2019), although recent studies have delved into the evolution of the NTK under more realistic conditions (Fort et al., 2020). In contrast, our work aims to directly verify that the properties we study are relevant to real-world settings. We also note the active research direction regarding the influence of BatchNorm on the optimization trajectory (Santurkar et al., 2018b; Ioffe & Szegedy, 2015a).

3. Background

The training of a neural network on a dataset comprised of n examples can typically be formulated as the finite-sum optimization problem

$$\min_{w\in\mathbb{R}^d} \mathcal{L}(w) := \frac{1}{n}\sum_{i=1}^{n} l_i(w), \tag{1}$$

where w are the parameters of the neural network, $\mathcal{L}$ is the empirical risk, and l_i corresponds to the loss function for the i-th data sample, for $i = 1, \dots, n$. We denote the empirical risk with respect to any minibatch $\mathcal{B} \subseteq [n]$ of size m as $\mathcal{L}_\mathcal{B} := \frac{1}{m}\sum_{i\in\mathcal{B}} l_i$. Throughout this work, we assume the loss to be differentiable, but we do **not** require it to be smooth.

Given an objective function $\mathcal{L}$ with a convex set of global minima $\mathcal{X}^\star$, and letting $w_p^\star$ be the orthogonal projection of w into $\mathcal{X}^\star$, we now recall the definitions of RSI^- and EB^+ as provided by Guille-Escuret et al. (2021) and Guille-Escuret et al. (2022),

Definition 3.1 (Lower Restricted Secant Inequality)**.** *Let* $\mu > 0$. $\mathcal{L} \in \mathrm{RSI}^-(\mu)$ *iff:*

$$\forall w \in \mathbb{R}^d, \nabla\mathcal{L}(w)^T(w - w_p^\star) \geq \mu \left\|w - w_p^\star\right\|_2^2. \tag{2}$$

Definition 3.2 (Upper Error Bounds)**.** *Let* $L > 0$. $\mathcal{L} \in \mathrm{EB}^+(L)$ *iff:*

$$\forall w \in \mathbb{R}^d, \left\|\nabla\mathcal{L}(w)\right\|_2 \leq L \left\|w - w_p^\star\right\|_2. \tag{3}$$

The classes of functions RSI^- and EB^+ are thus defined in the literature as those respecting the above bounds over the entire parameter space. However, in this work, our focus lies not merely on their extremal values but on the local quantities bounded by RSI^- and EB^+.

For simplicity, we refrain from introducing new terminology, and henceforth denote these quantities as $\mathrm{RSI}(G, w, w^\star)$ and $\mathrm{EB}(G, w, w^\star)$, where G is an estimator for the gradient at w. We do not mandate G to be the full gradient of $\mathcal{L}$; it could, for instance, correspond to the gradient $\nabla\mathcal{L}_\mathcal{B}$ with respect to a minibatch $\mathcal{B}$. Similarly, $w^\star$ is not assumed to be a minimum of the objective function. Formally, for any vector field G, and any $w^\star \in \mathbb{R}^d$, $w \neq w^\star$:

$$\mathrm{RSI}(G, w, w^\star) := \frac{G(w)^T(w-w^\star)}{\|w-w^\star\|_2^2} \quad \text{and}$$

$$\mathrm{EB}(G, w, w^\star) := \frac{\|G(w)\|_2}{\|w - w^\star\|_2}.$$

The ratio between RSI and EB imparts a direct geometrical interpretation:

$$\gamma(G, w, w^\star) := \frac{\mathrm{RSI}(G,w,w^\star)}{\mathrm{EB}(G,w,w^\star)} = \frac{G(w)^T(w-w^\star)}{\|G\|_2\|w-w^\star\|_2}$$

$$= cosine(G(w), w - w^\star),$$

where $cosine(w_1, w_2)$ is the cosine of the angle between vectors w_1 and w_2.

This ratio, γ, signifies the alignment between the negative sampled gradient and the direction from w to $w^\star$. When γ approaches 1, it indicates a negative gradient strongly directed toward $w^\star$. Conversely, a γ close to 0 suggests a gradient almost orthogonal to $w - w^\star$. A negative γ indicates a negative gradient directed away from $w^\star$. Additionally, γ can be interpreted as the inverse of a local variant of the condition number, $\kappa := \frac{\sup \mathrm{EB}}{\inf \mathrm{RSI}}$, which is a measure of the complexity of optimizing $\mathcal{L}$ in prior works (Guille-Escuret et al., 2021).

RSI and EB are intrinsically connected to the dynamics of stochastic gradient descent (SGD). Indeed, the distance to $w^\star$ following an SGD step with step size η can be precisely articulated using RSI and EB. For all $w \neq w^\star, \mathcal{B}$,

$$\begin{aligned} &\left\|w - \eta \nabla \mathcal{L}_{\mathcal{B}}(w) - w^\star\right\|_2^2 \\ &= \left\|w - w^\star\right\|_2^2 - 2\eta \nabla \mathcal{L}_{\mathcal{B}}(w)^T (w - w^\star) + \eta^2 \left\|\nabla \mathcal{L}_{\mathcal{B}}(w)\right\|_2^2 \\ &= \left(1 - 2\eta \,\mathrm{RSI}\left(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star\right) + \eta^2 \,\mathrm{EB}^2\left(\nabla \mathcal{L}_{\mathcal{B}}, w, , w^\star\right)\right) \left\|w - w^\star\right\|_2^2. \end{aligned} \tag{4}$$

Consequently, with a step size of

$$\eta^\star := \operatorname*{argmin}_{\eta} \left\|w - \eta \nabla \mathcal{L}_{\mathcal{B}}(w) - w^\star\right\|_2 = \frac{\mathrm{RSI}(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star)}{\mathrm{EB}^2(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star)}, \tag{5}$$

SGD guarantees

$$\left\|w_{t+1} - w^\star\right\|_2 = \sqrt{1 - \gamma(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star)^2} \left\|w_t - w^\star\right\|_2. \tag{6}$$

Furthermore, if $\inf_{w,\mathcal{B}} \mathrm{RSI}(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star) \geq \mu$ and $\sup_{w,\mathcal{B}} \mathrm{EB}(\nabla \mathcal{L}_{\mathcal{B}}, w, w^\star) \leq L$ hold for some $\mu > 0$, $L > 0$, then equation 4 demonstrates that running SGD with a fixed step size of $\eta = \frac{\mu}{L^2}$ will converge to $w^\star$ at a guaranteed rate:

$$\left\|w_t - w^\star\right\|_2^2 \leq (1 - \tfrac{\mu^2}{L^2})^t \left\|w_0 - w^\star\right\|_2^2, \tag{7}$$

This holds irrespective of how the minibatches are sampled. Under these assumptions, this rate is, in fact, worst-case optimal among all continuous first-order algorithms (Guille-Escuret et al., 2022).

Experimental Measurement of RSI and EB: One of the most significant challenges in experimentally measuring RSI and EB lies in the selection of $w^\star$. Even in cases where the objective function admits an unique global minimum, finding it in the context of deep neural networks is computationally infeasible (Blum & Rivest, 1992). To navigate this complication, we initially train a neural network and subsequently choose the final iterate w_T of the optimization sequence. Given successful training, the sequence will converge to the vicinity of a (local) minimum, and measuring RSI and EB with respect to this minimum will provide insightful understanding of the training dynamics.

Notably, under this procedure, w_T is dependent on the optimization sequence rather than being predetermined. Therefore, interpreting the ensuing results warrants care, see Section 6.

Considering that saving all gradients and iterates observed during training would be prohibitively resource-intensive, we perform two identical training runs. The first run computes $w^\star = w_T$, and the second run computes RSI and EB along the optimization path. A detailed description of our experimental protocol is provided in Algorithm 1 in Appendix A.1, and we share our code at `https://github.com/Hiroki11x/LossLandscapeGeometry`.

4. Empirical Geometry of Landscapes Along Optimization Paths

Figure 1 offers an initial glance at our results, outlining the behavior of γ across four datasets, with variations across architecture, batch size, and optimization technique. Figure 2 presents a more streamlined view on three of these datasets, exhibiting not only γ but also RSI and EB on a single run to preserve clarity. To avoid precision issues when w_t approaches w_T, the results from the final epoch have been excluded. Our hyperparameters were initially adjusted to optimize validation accuracy, echoing practical conditions. All experiments were coded in PyTorch (Paszke et al., 2019) and detailed descriptions of the specific training configurations, along with final test performances, are available in Appendix A to ensure full reproducibility.

CIFAR-10 (ResNet-18): Across the entire training run, not a single iteration exhibits a negative γ, though depending of the seed, it may happen that one or two iterations toward the end of training exhibit slightly negative γ (see Figure 4). Even though there are slight fluctuations across epochs, γ predominantly remains within the $[0.0075, 0.02]$ range and does not exhibit substantial shifts. While the variance of RSI and EB across iterations tends to increase as training progresses, their mean values largely remain stable.

ImageNet-1K (ResNet-50): Except for a few iterations at the very early stage, γ remains positive throughout all of training. Moreover, the variance across iterations is notably low until the last epochs. Epoch-wise, RSI, EB, and γ increase monotonically, with a sharp rise observed towards the end.

WikiText-2 (Transformer): Throughout training, γ remains strictly positive and always exceeds 0.05 after the

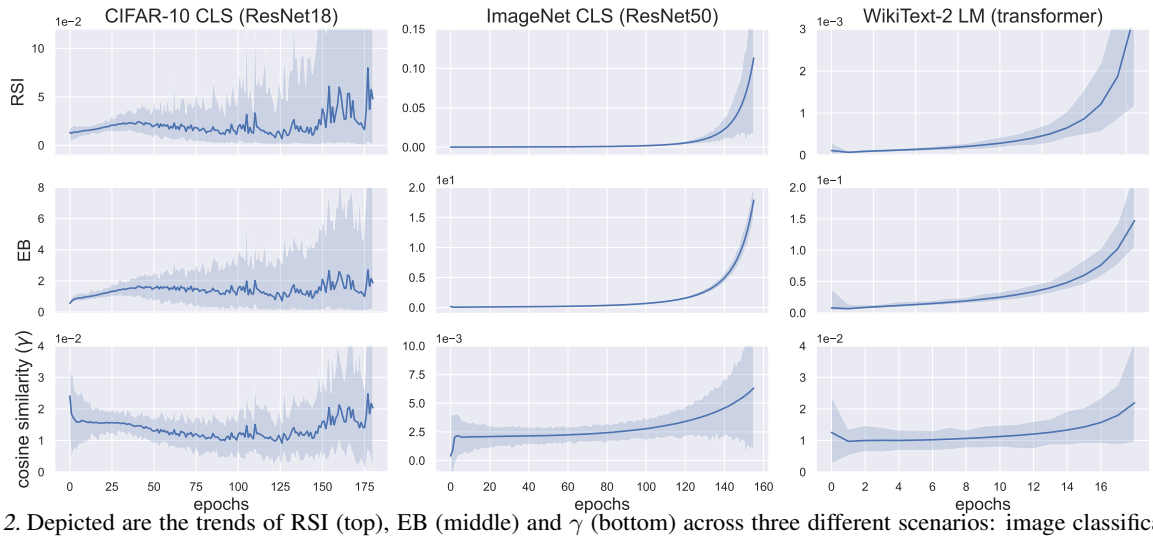

Figure 2. Depicted are the trends of RSI (top), EB (middle) and γ (bottom) across three different scenarios: image classification on CIFAR-10 with a ResNet-18 (left), image classification on ImageNet with a ResNet-50 (middle) and language modeling on WikiText-2 with a transformer model (right).

second epoch. The cosine similarity maintains a remarkable stability, exhibiting only minor variations across iterations and epochs. While RSI and EB show very low variance within epochs, they do increase towards the end of the training period.

4.1. Fundamental properties

Upon careful analysis, we find that the optimization trajectories of deep neural networks exhibit the following major characteristic features:

- The cosine similarity, γ, is almost always positive.
- γ demonstrates notable stability across both epochs and iterations, rarely departing from its (low) average value.
- RSI and EB follow predictable trends, contingent upon whether the model adheres to an interpolation or a non-interpolation regime.

Interpolation vs Non-Interpolation Regime: The behavior of RSI and EB are directly tied to how well the final iterate $w^\star$ interpolates the training data. For CIFAR-10, where the model reaches close to 0 training loss, RSI and EB retain relatively stable mean values up to the last epochs, which is made possible by stochastic gradients decreasing to 0 as w_t approaches $w^\star$. Conversely, in scenarios where the model fails to interpolate the training data, such as for ImageNet and WikiText-2, stochastic gradients remain significant. In such a scenario, RSI and EB inevitably rise to infinity as $w_t - w^\star$ approaches zero. This phenomenon is particularly obvious with ImageNet due to the learning rate decay, which induces minuscule distances between w_t and $w^\star$ in the later stages of training. Additional experimental results supporting this interpretation are provided in Appendix C.

Late Training Behavior: The results obtained towards the end of training should be interpreted with caution. Besides the previously described phenomenon in the non-interpolation regime, the correlation between sampled gradients and $w_t - w^\star$ increases as the sequence nears its termination. Intuitively, $w^\star$ approximates a minima, and the approximation error becomes significant as iterates get sufficiently close. Further discussion on related implications can be found in Section 6.

Low Value of Cosine Similarity: The low values of γ empirically encountered are to be expected: if γ was stable at reasonably high values, then we would find a near-minima in a small number of steps using SGD, which is notoriously not the case for modern problems. Instead, optimization sequences approach their final iterate at a slow but regular pace. While the stability and positivity of γ imply a linear convergence rate, its low value indicate a linear rate close to 1, similarly to a strongly convex and smooth objective being badly conditioned. A plausible cause for γ being small is that the useful signal from generalizable features in sampled gradients is dominated by that of spurious and coincidental correlations.

Significance: These observations imply that, despite the well-documented non-convexity of the loss landscapes associated with neural networks and the inherent stochasticity introduced by minibatch sampling, the learning process of neural networks remains remarkably consistent. The networks progress steadily towards their destination throughout the training, with each stochastic gradient contributing valuable information to reach the final model state. With very few exceptions, gradients always point toward the right direction, and training trajectories never take a wrong turn

when optimizing the loss function. We find these observations to be particularly remarkable on ImageNet. Given the presence of 1000 semantic classes (exceeding the batch size) and in excess of 5000 minibatches per epoch, the consistence of the cosine similarity γ throughout entire epochs seems surprising. In addition, Section 6 establishes links between empirically adopted learning rate schedules and RSI and EB. Overall, RSI and EB are powerful tools to capture the elusive simplicity of neural loss landscapes, with empirical properties theoretically guaranteeing linear convergence rates. We thus encourage future works to consider RSI and EB to characterize the classes of objectives encountered in deep learning applications.

We further explore the impact of various factors and provide a more comprehensive substantiation of our findings in Section 5. Following this, we discuss the implications and potential limitations of our observations in Section 6. We also discuss plausible causes in Appendix D.

5. Influence of Training Settings

Batch Size: The top right of Figure 1 delineates the cosine similarities corresponding to batch sizes ranging from 32 to 256 on the WikiText-2 dataset. As a complementary experiment, Figure 6 in Appendix B portrays the cosine similarities associated with batch sizes from 64 to 512 on the CIFAR-10 dataset. The outcomes of both these experiments consistently reveal a positive correlation between batch size and cosine similarity. This outcome is foreseeable: for two minibatches $\mathcal{B}_i$ and $\mathcal{B}_j$, we have

$$\begin{aligned}&\mathrm{RSI}(\nabla\mathcal{L}_{\mathcal{B}_i}+\nabla\mathcal{L}_{\mathcal{B}_j})=\mathrm{RSI}(\nabla\mathcal{L}_{\mathcal{B}_i})+\mathrm{RSI}(\nabla\mathcal{L}_{\mathcal{B}_j}),\\ &EB(\nabla\mathcal{L}_{\mathcal{B}_i}+\nabla\mathcal{L}_{\mathcal{B}_j})\leq\mathrm{EB}(\nabla\mathcal{L}_{\mathcal{B}_i})+\mathrm{EB}(\nabla\mathcal{L}_{\mathcal{B}_j}).\end{aligned}$$

It should be noted that the selection of batch size not only affects the measurement of RSI and EB, but it also influences the optimization trajectory and the speed of convergence. Therefore, direct numerical comparisons across different batch sizes ought to be interpreted with caution. Nonetheless, our observations suggest that cosine similarities may scale with the square root of the batch size.

Optimizer: Figure 1 (bottom left) illustrates the cosine similarity for three distinct optimizers utilized on the CIFAR-10 dataset. Intriguingly, Adam appears to result in lower cosine similarity values, albeit with reduced variance. We hypothesize that Adam, by amplifying the effective step size along directions with lower curvature, traverses further in flat dimensions, thereby leading to a reduced alignment compared to SGD. This conjecture is substantiated by Figure 24 in Appendix C, demonstrating that the journey undertaken by Adam indeed surpasses that of SGD in terms of distance. Notably, the employment of a momentum value of 0.9 with SGD does not significantly impact the value of γ, compared to not using momentum. Prior works also suggest that the optimization methods may affect the geometry of visited regions (Cohen et al., 2021).

Model Depth and Width: Our attention now turns to the impact of depth and width on the geometric characteristics of the optimization trajectory, as depicted in Figure 1 (top left). In this experiment, we trained ResNets of varying depth — 18, 50, and 152 layers — with both standard and doubled width. A salient observation is that an increase in depth slightly enhances the cosine similarities, while an increase in width appears to have a comparatively trivial impact. These findings could potentially shed light on the prevalent trend in contemporary neural network designs favouring increased depth over width (He et al., 2016).

6. Key Takeaways and Discussion

Geometrically Justified Learning Rate Schedules: As established in equation 5, we define the locally optimal learning rate (loLR) as the minimizer of $\|w_t-\eta\nabla\mathcal{L}_{\mathcal{B}_t}-w^\star\|_2$, $\eta^\star(w)=\frac{\mathrm{RSI}(w)}{\mathrm{EB}^2(w)}$.

It is important to note, however, that $\eta^\star$ may not necessarily be globally optimal. Indeed, certain methodologies may initiate slower but accumulate more information, ultimately leading to faster convergence over a large number of steps. Furthermore, as the measurement of RSI and EB requires the knowledge of $w^\star$, which in turn depends on the learning rate (LR), the expression cannot be utilized to dynamically tune it.

Despite these limitations, we find intriguing parallels between the evolution of the loLR derived from our experiments and the shape of empirically validated LR schedules, as demonstrated in Figure 3. For instance, a widely adopted strategy for training on ImageNet involves a linear warm-up phase of the LR for the initial few epochs, followed by a cosine annealing phase. This pattern is mirrored in our empirical observations on ImageNet, except for a sharper decrease immediately after warmup.

Moreover, the results on WikiText-2 echo two popular practices: linearly decreasing the LR and increasing the batch size over time. These intriguing observations suggest that the geometry of the loss landscape could potentially inform the design of more effective learning rate schedules.

Lastly, the apparent correspondence between loLR and empirical learning rate strategies implies that the efficiency of fixed learning rates may be contingent upon the stationarity of RSI and EB. Similarly, the existence of straightforward and efficient learning rate schedules can be associated with the predictable evolution of these geometrical properties. This strongly reinforces the view that such geometrical attributes play a substantial role in the widespread practical successes of deep learning.

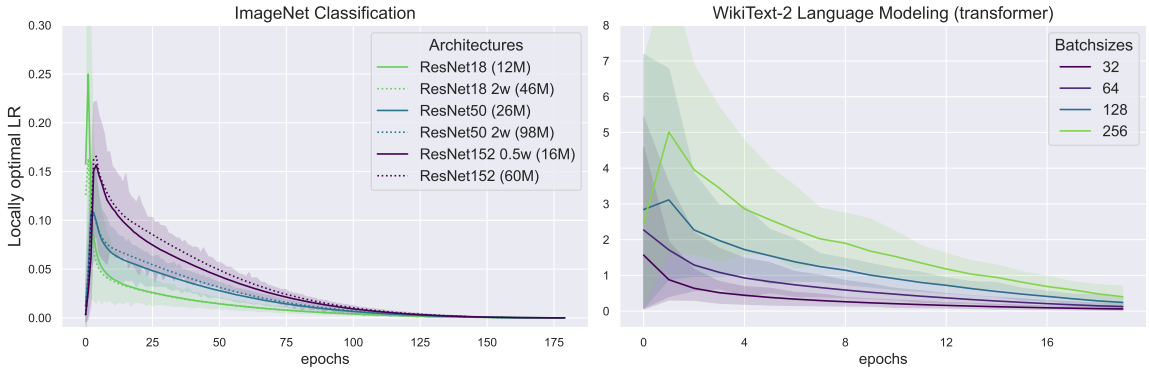

Figure 3. Left panel: The locally optimal learning rate, derived as per equation 5, for various architectures implemented on the CIFAR-10 dataset. Right panel: The locally optimal learning rate, similarly determined, across a spectrum of batch sizes employed on the WikiText-2 dataset.

Biases Induced by Using Final Iterates as Reference Points: A critical limitation of our experimental approach is the inescapable correlation between $w^\star$ and the optimization sequence. This association must be thoroughly addressed to appropriately interpret our findings.

• **Initialization:** Firstly, RSI and EB may represent local properties of the loss landscape, and could be dependent on the initialization region. However, this possibility is refuted by the left panel of Figure 4, which demonstrates minimal variation in γ measurements across different random seeds.

• **Epoch Budget:** Secondly, our results might be influenced by the particular moment when we terminate the optimization sequence to extract $w^\star$. The right panel of Figure 4 presents different measurements for epoch budgets ranging from 100 to 280, with all other parameters kept consistent. Our findings indicate a relative similarity in results before the sequence nears $w^\star$, suggesting that our experiments do not display excessive sensitivity to the epoch budget.

• **Induced Bias:** However, this experiment also underscores the phenomenon detailed in Section 4.1: as the sequence approaches completion, the correlation between sampled gradients and $w_t - w^\star$ - induced by gradient updates - becomes increasingly significant. This correlation is a by-product of the optimization method, rather than a feature of local geometry, and augments the value of RSI and γ by diminishing the impact of stochasticity. Consequently, this correlation should be taken into account when interpreting RSI and EB in the concluding epochs.

A compelling illustration of this correlation can be seen in a discrete isotropic random walk with a fixed step size s in a dimension d. When dimension d significantly exceeds the number of steps, each pair of steps can be assumed to be nearly orthogonal with high probability. In such a setting, if we denote $(x_t)_{t=0\dots T}$ as the sequence generated by the random walk, we can calculate that, with high probability, $\forall t$,

$$\frac{(x_t - x_{t+1})^T (x_t - x_T)}{\|x_t - x_T\|_2^2} \approx \frac{\|x_t - x_{t+1}\|_2^2}{\|x_t - x_T\|_2^2} \approx \frac{1}{T-t} > 0$$
$$\text{and} \quad \frac{\|x_t - x_{t+1}\|_2}{\|x_t - x_T\|_2} \approx \frac{1}{\sqrt{T-t}} \tag{8}$$

Consequently, the cosine similarity $\gamma(x_t) \approx (T-t)^{-0.5}$ remains strictly positive, and experiences a sharp increase toward the end, exemplifying the effect of the correlation induced by the selection of $w^\star$. It's worth noting that in the case of neural networks, γ remains approximately constant for the majority of training (as is clearly visible in Figure 1), which marks a distinction in their dynamics. Nonetheless, akin to the random walk scenario, it can be anticipated that the correlation induced by the choice of $w^\star$ would become increasingly evident as the number of remaining iterations diminishes.

Contrasting Examples: Functions Without Beneficial Geometric Properties: We now turn our attention to delineating the behaviors that could potentially manifest in stochastic and non-convex optimization scenarios. To this end, we have engineered two illustrative counter-examples which effectively demonstrate that the consistency observed in Sections 4 and 5 is not a mere byproduct of our experimental paradigm.

Our first example, termed Asymmetric Linear Model (ALM), entails the training of a linear model with the objective of consistently yielding outputs that are lower than their corresponding targets. The error between these values is calculated on stochastic minibatches using Root Mean Square Error (RMSE), thereby introducing a substantial degree of stochasticity. Despite this, the objective is a finite sum of convex functions and thus remains convex.

The second function, designated Sinusoidal Mixture (SM), is deterministic but exhibits a pronounced degree of non-convexity. The mathematical expressions for both ALM and

Figure 4. Depiction of cosine similarities during the training of a ResNet-18 on the CIFAR-10 dataset, with variations in (left) initialization seed and (right) epoch budget.

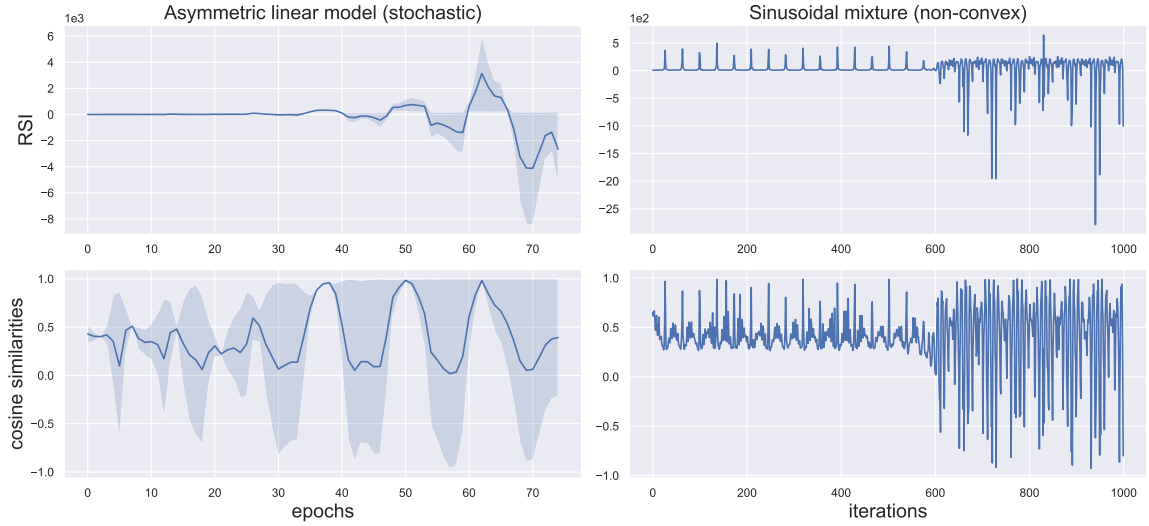

Figure 5. Left column: RSI and γ values for a convex yet significantly stochastic objective. Right column: RSI and γ values for a deterministic objective characterized by substantial non-convexity. Despite their simplicity, these functions exemplify the irregular behaviors one might anticipate encountering within the complex terrain of neural loss landscapes.

SM are presented below, with coefficients a_i, x_i, y_i drawn randomly from normal distributions,

$$\begin{aligned} ALM(w) &= \sum_i \left[max(0, w^T x_i - y_i)\right]^2; \\ SM(w) &= \|w\|_2^2 + 100 \sum_i \sin(a_i w_i)^2. \end{aligned} \tag{9}$$

We show in Appendix the level lines of both functions (see Figure 14 and Figure 15).

Figure 5 presents the measurements of RSI and γ for both ALM and SM. Although these functions are characterized by relatively simple functional forms and do not simultaneously exhibit stochasticity and non-convexity, they demonstrate unpredictable trajectories and negative values for RSI and γ. This evidence compellingly suggests that the observed simplicity associated with neural networks is not a trivial characteristic.

7. Conclusion

We have conducted an extensive series of experiments, assessing RSI and EB across a broad spectrum of training settings. These experiments reveal that these geometric properties display a collection of desirable characteristics, effectively demonstrating that neural network training proceeds smoothly, maintaining a consistently steady advancement towards its destination throughout the training process.

These results contrast starkly with the theoretical complexity of neural landscapes and potentially open new pathways for developing theoretical results tailored to deep learning, or for designing optimization algorithms that exploit the geometry of empirical objective functions.

A noteworthy point is that while RSI and EB appear to encapsulate significant beneficial aspects of neural networks, they likely do not encompass the entire scope of these advantages. There may be additional, complementary properties yet to be discovered. An intriguing indication of this is

the fact that vanilla gradient descent has been proven to be exactly optimal for functions verifying the lower restricted secant inequality and upper error bound (Guille-Escuret et al., 2022). Given the well-documented efficacy of momentum in training neural networks, we conjecture that momentum exploits additional properties not captured by RSI and EB, which we encourage future works to explore.

Impact Statement

The goal of this paper is to improve our understanding of neural loss landscapes. While we hope our work will eventually contribute to designing better algorithms, its impact is tied to that of machine learning in general. We believe there is no specific impact that needs to be highlighted here.

Acknowledgement

The authors would like to extend special thanks to Leonard Boussioux, Damien Scieur and Baptiste Goujaud, for thoughtful discussions and useful feedback. Ioannis Mitliagkas acknowledges support by an NSERC Discovery grant (RGPIN-2019-06512), a Samsung grant and a Canada CIFAR AI chair.

The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF)(Cramer & Haala, 2010)[1]

The computation resource of this project is supported by TSUBAME3.0[2] provided by Tokyo Institute of Technology through the Exploratory Joint Research Project Support Program from JHPCN (EX23401) and TSUBAME Encouragement Program for Young / Female Users.

References

Audebert, N., Saux, B. L., and Lefèvre, S. Beyond rgb: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017. ISSN 0924-2716. doi: https://doi.org/10.1016/j.isprsjprs.2017.11.011.

Badrinarayanan, V., Kendall, A., and Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. doi: 10.1109/TPAMI.2016.2644615.

Blum, A. L. and Rivest, R. L. Training a 3-node neural network is np-complete. *Neural Networks*, 5(1):117–127, 1992. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(05)80010-3. URL https://www.sciencedirect.com/science/article/pii/S0893608005800103.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/ae614c557843b1df326cb29c57225459-Paper.pdf.

Cohen, J. M., Kaur, S., Li, Y., Kolter, J. Z., and Talwalkar, A. Gradient descent on neural networks typically occurs at the edge of stability. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=jh-rTtvkGeM.

Cramer, M. and Haala, N. Dgpf project: Evaluation of digital photogrammetric aerial-based imaging systems-overview and results from the pilot center. *Photogrammetric engineering and remote sensing*, 76(9):1019–1029, 2010.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Dinh, L., Pascanu, R., Bengio, S., and Bengio, Y. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, pp. 1019–1028. PMLR, 2017.

Dmitriy Drusvyatskiy, A. S. L. Error bounds, quadratic growth, and linear convergence of proximal methods. In *Mathematics of Operations Research 43(3):919-948*, 2018. URL https://doi.org/10.1287/moor.2017.0889.

Fatras, K., Damodaran, B. B., Lobry, S., Flamary, R., Tuia, D., and Courty, N. Wasserstein Adversarial Regularization for learning with label noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Fort, S., Dziugaite, G. K., Paul, M., Kharaghani, S., Roy, D. M., and Ganguli, S. Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5850–5861. Curran Associates, Inc.,

[1] http://www.ifp.uni-stuttgart.de/dgpf/DKEP-

[2] https://www.t3.gsic.titech.ac.jp/en

2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/405075699f065e43581f27d67bb68478-Paper.pdf.

Frankle, J., Dziugaite, G. K., Roy, D. M., and Carbin, M. Linear mode connectivity and the lottery ticket hypothesis. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3259–3269. PMLR, 2020. URL http://proceedings.mlr.press/v119/frankle20a.html.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. Loss surfaces, mode connectivity, and fast ensembling of dnns. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/be3087e74e9100d4bc4c6268cdbe8456-Paper.pdf.

Goodfellow, I., Vinyals, O., and Saxe, A. Qualitatively characterizing neural network optimization problems. In *International Conference on Learning Representations*, 2015. URL http://arxiv.org/abs/1412.6544.

Guille-Escuret, C., Girotti, M., Goujaud, B., and Mitliagkas, I. A study of condition numbers for first-order optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 1261–1269. PMLR, 2021.

Guille-Escuret, C., Ibrahim, A., Goujaud, B., and Mitliagkas, I. Gradient descent is optimal under lower restricted secant inequality and upper error bound. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=s1yaWFDLxVG.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 448–456, Lille, France, 07–09 Jul 2015a. PMLR. URL https://proceedings.mlr.press/v37/ioffe15.html.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. pmlr, 2015b.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.

Jin, C., Netrapalli, P., Ge, R., Kakade, S. M., and Jordan, M. I. On nonconvex optimization for machine learning: Gradients, stochasticity, and saddle points, 2019.

Karimi, H., Nutini, J., and Schmidt, M. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In Frasconi, P., Landwehr, N., Manco, G., and Vreeken, J. (eds.), *Machine Learning and Knowledge Discovery in Databases*, volume abs/1608.04636, pp. 795–811, Cham, 2016. Springer International Publishing.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2017.

Krizhevsky, A. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014.

Krizhevsky, A., Nair, V., and Hinton, G. Learning multiple layers of features from tiny images. *Advances in Neural Information Processing Systems*, 2012. URL http://www.cs.toronto.edu/~kriz/cifar.html.

Lee, J., Xiao, L., Schoenholz, S. S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. *arXiv preprint arXiv:1902.06720*, 2019.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/a41b3bb3e6b050b6c9067c67f663b915-Paper.pdf.

Liu, C., Zhu, L., and Belkin, M. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *Applied and Computational Harmonic Analysis*, 59:85–116, 2022. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2021.12.009. URL https://www.sciencedirect.com/science/article/pii/S106352032100110X. Special Issue on Harmonic Analysis and Machine Learning.

Liu, C., Drusvyatskiy, D., Belkin, M., Davis, D., and Ma, Y.-A. Aiming towards the minimizers: fast convergence of sgd for overparametrized problems, 2023.

Logan, IV, R. L., Liu, N. F., Peters, M. E., Gardner, M., and Singh, S. Barack's wife Hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Florence, Italy, July 2019. Association for Computational Linguistics.

Loizou, N., Vaswani, S., Laradji, I. H., and Lacoste-Julien, S. Stochastic polyak step-size for SGD: an adaptive learning rate for fast convergence. In Banerjee, A. and Fukumizu, K. (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 1306–1314. PMLR, 2021. URL http://proceedings.mlr.press/v130/loizou21a.html.

Lucas, J., Bae, J., Zhang, M. R., Fort, S., Zemel, R. S., and Grosse, R. B. Analyzing monotonic linear interpolation in neural network loss landscapes. *CoRR*, abs/2104.11044, 2021. URL https://arxiv.org/abs/2104.11044.

Luo, Z.-Q. and Tseng, P. Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(1):157–178, 1993.

Marcus, M., Santorini, B., and Marcinkiewicz, M. A. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330, 1993.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models, 2016.

Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J., and Khudanpur, S. Recurrent neural network based language model. In *Interspeech*, volume 2, pp. 1045–1048. Makuhari, 2010.

Nagarajan, V. and Kolter, J. Z. Uniform convergence may be unable to explain generalization in deep learning. *CoRR*, abs/1902.04742, 2019. URL http://arxiv.org/abs/1902.04742.

Nguyen, L. M., Nguyen, P. H., van Dijk, M., Richtárik, P., Scheinberg, K., and Takác, M. SGD and hogwild! convergence without the bounded gradients assumption. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3747–3755. PMLR, 2018. URL http://proceedings.mlr.press/v80/nguyen18c.html.

Panageas, I., Piliouras, G., and Wang, X. First-order methods almost always avoid saddle points: The case of vanishing step-sizes. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3fb04953d95a94367bb133f862402bce-Paper.pdf.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library, 2019.

Qian, X., Richtárik, P., Gower, R. M., Sailanbayev, A., Loizou, N., and Shulgin, E. SGD with arbitrary sampling: General analysis and improved rates. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5200–5209. PMLR, 2019. URL http://proceedings.mlr.press/v97/qian19b.html.

Qian, Y., Pan, S., and Xiao, L. Error bound and exact penalty method for optimization problems with nonnegative orthogonal constraint. *IMA Journal of Numerical Analysis*, 02 2023. ISSN 0272-4979. doi: 10.1093/imanum/drac084. URL https://doi.org/10.1093/imanum/drac084. drac084.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *The International Conference on Medical image computing and computer-assisted intervention*, 2015.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitez, S., and Breitkopf, U. The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2012.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Santurkar, S., Tsipras, D., Ilyas, A., and Madry, A. How does batch normalization help optimization? In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018a. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/905056c1ac1dad141560467e0a99e1cf-Paper.pdf.

Santurkar, S., Tsipras, D., Ilyas, A., and Madry, A. How does batch normalization help optimization? In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018b. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/905056c1ac1dad141560467e0a99e1cf-Paper.pdf.

Schöpfer, F. Linear convergence of descent methods for the unconstrained minimization of restricted strongly convex functions. *SIAM J. Optim.*, 26:1883–1911, 2016.

Shamir, O. and Zhang, T. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 71–79. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/shamir13.html.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Yi, X., Zhang, S., Yang, T., Johansson, K. H., and Chai, T. Exponential convergence for distributed smooth optimization under the restricted secant inequality condition, 2019.

Yuan, K., Ling, Q., and Yin, W. On the convergence of decentralized gradient descent. *SIAM J. Optim.*, 26:1835–1854, 2016.

Zhang, H. and Yin, W. Gradient methods for convex minimization: better rates under weaker conditions. Cam report, UCLA, 2013.

Zhou, Z. and So, A. A unified approach to error bounds for structured convex optimization problems. *Mathematical Programming*, 12 2015. doi: 10.1007/s10107-016-1100-9.

A. Experimental Setting

A.1. Algorithm

Algorithm A.1 provides a detailed account of our experimental protocol. This algorithm describes our methodology to measure RSI and EB values throughout the training process.

Intuitively, the algorithm follows two training runs with identical initialization and minibatch sampling. The first runs aims at computing the last iterate $w^\star$ and saving it, and the second runs uses $w^\star$ to compute RSI and EB.

This approach removes the necessity to save all gradients throughout the run (in order to compute RSI and EB at the end), which would be unreasonably expensive in memory.

Algorithm 1 Measurement of RSI and EB

Input: initial weights w_0, sequence of minibatches $\mathcal{B}_{0\dots T-1}$

for $t = 0, \dots, T-1$ **do**
  compute gradient $G_t = \nabla \mathcal{L}_{\mathcal{B}_t}(w_t)$
  update weights $w_{t+1} = Opt(w_{0\dots t}, G_{0\dots t}))$
end for
$w^\star \leftarrow w_T$
reset weights to w_0
for $t = 0, \dots, T-1$ **do**
  compute gradient $G_t = \nabla \mathcal{L}_{\mathcal{B}_t}(w_t)$
  compute $RSI_t = \frac{G_t^T(w_t - w^\star)}{\|w_t - w^\star\|_2^2}$
  compute $EB_t = \frac{\|G_t\|_2}{\|w_t - w^\star\|_2}$
  update weights $w_{t+1} = Opt(w_{0\dots t}, G_{0\dots t}))$
end for

Output: $RSI_{0\dots T-1}, EB_{0\dots T-1}$

A.2. Implementation and Environment for Experiments

Computational Environment: We perform our experiments mainly with cluster A (redacted until publication). For cluster A, each node is composed of NVIDIA A100×4GPU and AMD Milan 7413 @ 2.65 GHz 128M cache L3×2CPU. As a software environment, we use Rocky Linux 8.7, gcc 9.3.0, Python 3.10.2, pytorch 1.13.1, torchvision 0.14.1, cuDNN 8.2.0, and CUDA 11.4.

Licence of Datasets: It should be noted that the CIFAR-10 dataset (Krizhevsky et al., 2012) does not explicitly stipulate any licensing terms [3]. The authors of the CIFAR-10 merely ask users of their dataset to provide appropriate citation. ImageNet-1K (Deng et al., 2009) does not explicitly state its license [4]. Licenses of the WikiText-2 (Logan et al., 2019) is CC-BY-SA-3.0 [5]. No license is specified for the dataset in Vaihingen (Cramer & Haala, 2010), but it is allowed to be used in scientific papers. However, acknowledgment and citation are required[6].

The WIkiText-103 dataset is available under the Creative Commons Attribution-ShareAlike License.

Implementation: All codes for experiments are modifications of the codes provided by PyTorch's official implementation for image classification and language modeling tasks[7] and Audebert et al. (2017) for segmentation task [8]. The license for the official Pytorch implementation is the BSD-3-Clause, and the license for the segmentation task implementation is GPLv3. Our code can be found at the link below.
https://github.com/Hiroki11x/LossLandscapeGeometry

A.3. Datasets description

CIFAR10: CIFAR-10 dataset (Krizhevsky et al., 2012), one of the most widely used datasets for machine learning research, is a unique resource that offers a robust benchmark for algorithms, primarily image recognition. The dataset is a curated collection of 60,000 color images, each of a size of 32x32 pixels, uniformly divided across ten distinctive classes. These classes encompass various common objects: airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks. Each class in the CIFAR-10 dataset is represented equally, with 6,000 images per category. The dataset is split into two segments: a training set comprising 50,000 images and a test set of 10,000 images.

ImageNet-1K: The ImageNet-1K dataset, a subset of the more extensive ImageNet database (Deng et al., 2009), has become an essential resource for research in machine learning, particularly for image recognition and classification tasks. ImageNet-1K is an extensively curated dataset of approximately 1.28 million high-resolution color images spread across 1,000 distinct categories or classes. These classes span various objects, organisms, and phenomena, capturing a rich diversity of the visual world.

WikiText-2: The WikiText-2 dataset (Logan et al., 2019) is a significant benchmark for various natural language processing tasks, specifically those related to language modeling. It comprises over 2 million tokens extracted from

[3] https://www.cs.toronto.edu/~kriz/cifar.html

[4] https://www.image-net.org/challenges/LSVRC/2012/index.php

[5] https://www.salesforce.com/products/einstein/ai-research/the-wikitext-dependency-language-modeling-dataset/

[6] For more details, see page 7 of https://www2.isprs.org/media/komfssn5/complexscenes_revision_v4.pdf

[7] https://github.com/pytorch/examples

[8] https://github.com/nshaud/DeepNetsForEO

verified Wikipedia articles. WikiText-2 retains the original structure and complexity of the language found in the source articles. This characteristic has enabled training models to handle various language structures and styles. The dataset is divided into three segments: a training set with roughly 2.08 million tokens, a validation set with approximately 217,000 tokens, and a test set with about 245,000 tokens.

Vaihingen: The Vaihingen dataset (Rottensteiner et al., 2012) is a land covering remote sensing dataset. Its purpose is to segment correctly aerial images of the Vaihingen city in Germany. It is composed of 33 tiles and we use 11 tiles for training, 5 tiles for validation, and the remaining 17 tiles for testing our model, which is the split used in (Fatras et al., 2021). Furthermore, we only consider the RGB components of the Vaihingen dataset. We follow the training procedure and PyTorch implementation from (Audebert et al., 2017). We build our training (resp. validation) dataset by taking randomly 256×256 patches from the training (resp. validation) tiles. The number of images seen during a training epoch is set to 10.000 patches while it is set to 1000 for the validation set.

A.4. Hyperparameters and Detailed Configurations

In the experimental procedure of our study, we employed a systematic grid search method to explore hyperparameters. This approach facilitates the identification of the most effective combinations that provide superior performance.

The specifics concerning the batch size and the total number of epochs allocated for each dataset and corresponding model have been exhaustively tabulated in Table 1. These parameters were meticulously selected to ensure optimal learning while mitigating overfitting concerns.

Further, we present detailed settings of specific ablation experiments in Table 2, 3,4, and 5. These ranges were defined based on prior search of hyperparameters maximizing validation performance.

'SGD' denotes the standard SGD algorithm without momentum, 'Momentum' denotes SGD with momentum with $\beta = 0.9$, and 'Adam' denotes the Adam algorithm with $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

CIFAR10: We train a ResNet-18 (He et al., 2016) for 190 epochs on CIFAR-10 (Krizhevsky et al., 2012) with SGD + momentum using a batch size of 256, a weight decay of 10^{-6}, and a fixed step size of 10^{-2} as a default configuration.

For batch-size experiments, the learning rate was designated as 5.0×10^{-3} for a batch size of 64, and subsequently scaled proportionally to the square root of the batch size, adhering to the guidelines from prior research (Krizhevsky, 2014).

ImageNet-1K: We train a ResNet-50 on ImageNet (Deng et al., 2009) for 180 epochs with SGD + momentum using a batch size of 256, weight decay of 10^{-4}, and a learning rate of 10^{-3}. The learning rate is subjected to a linear warmup for the first 3 epochs, followed by cosine annealing as a default configuration. We indicate by 'max LR' the maximum value of the learning rate, reached after the warmup epochs.

WikiText-2: We train a transformer (Vaswani et al., 2017) [9] on WikiText-2 (Merity et al., 2016) for 20 epochs with Adam (Kingma & Ba, 2017) using a batch size of 32, weight decay of 10^{-5} and learning rate of 10^{-4}.

Vaihingen: We train a SegNet (Badrinarayanan et al., 2017) and a UNet (Ronneberger et al., 2015). We augment our data with flip and mirror transformations. We use a batch size of 10 patches taken randomly within images as done in (Audebert et al., 2017). We train for 25 epochs with SGD + momentum, learning rate 0.01 and weight decay $1e^{-5}$. We then do an extra epoch with the learning rate and the weight decay divided by 10. Note that we train both the UNet and the SegNet from *scratch*.

[9] We use pytorch official implementation of transformer for language model: https://github.com/pytorch/examples/blob/main/word_language_model/model.py

Table 1. Default setting of experiments

Task	Dataset	Model	Batch size	Epochs
Image Classification	CIFAR-10	ResNet18-1	[64, 128, 256, 512]	[100, 190, 280]
		Medium-MLP	[64, 128, 256, 512]	[100, 190, 280]
	ImageNet-1K	ResNet-18-1	256	[90, 180]
		ResNet-18-2	256	[90, 180]
		ResNet-50-1	256	[90, 180]
		ResNet-50-2	256	[90, 180]
		ResNet-152-0.5	256	[90, 180]
		ResNet-152-1	256	[90, 180]
Word Language Model	WikiText-2	Transformer	[32, 64, 128, 256]	20
Segmentation	Vaihingen	UNet	10	26
		SegNet	10	26

Table 2. Hyperparameter: Image Classification Task (CIFAR-10)

Task	Model	Dataset	Optimizer	Batch size	LR	Epochs Budget
Optimizer	ResNet18-1	CIFAR-10	[SGD, Momentum, Adam]	256	[0.0001, 0.0005, 0.001]	[100, 190, 280]
Seed	ResNet18-1	CIFAR-10	Momentum	256	0.01	190
Batch Size	ResNet18-1	CIFAR-10	Momentum	[64, 128, 256, 512]	0.005 [10]	190
Model	[Medium-MLP, ResNet18-2]	CIFAR-10	Momentum	256	0.01	150

Table 3. Hyperparameter: Image Classification Task (ImageNet-1K)

Task	Model	Dataset	Optimizer	Batch size	max LR	Epochs Budget
Model	[ResNet18-1, ResNet18-2, ResNet50-1, ResNet50-2, ResNet152-05, ResNet152-1]	ImageNet-1K	Momentum	256	0.1	[90, 180]

Table 4. Hyperparameter: Language Model Task (WikiText-2)

Task	Model	Dataset	Optimizer	Batch size	LR	Epochs Budget
Batch Size	Transformer	WikiText-2	Adam	[32, 64, 128, 256]	0.0001 [11]	20

Table 5. Hyperparameter: Segmentation Task (Vaihingen)

Task	Model	Dataset	Optimizer	Batch size	LR	Epochs Budget
Model	[UNet, SegNet]	Vaihingen	Momentum	10	0.01	26

[10]The base learning rate is configured with an assumption of a batch size of 64. If the batch size is doubled, the learning rate should be multiplied by the square root of 2.

[10]Same as above.

[11]Same as above.

A.5. Validation performance

To support the relevance of our experimental setting, we report the validation performance in the standard settings of each dataset and model.

Table 6. Validation accuracy on CIFAR-10 with batch size 256.

Model	Validation accuracy
ResNet18-1	90.25
Vanilla MLP	59.42

Table 7. Validation accuracy on ImageNet with batch size 256.

Model	Validation accuracy
ResNet18-1	67.63
ResNet18-2	69.75
ResNet50-1	72.31
ResNet50-2	73.67
ResNet152-0.5	72.23
ResNet152-1	73.07

Table 8. Validation perplexity on WikiText-2 with batch size 64.

Model	Validation perplexity
Transformer	60.72

Table 9. Validation accuracy on Vaihingen with batch size 10.

Model	Validation accuracy
SegNet	84.56
UNet	85.40

B. Ablation Study

In this section, we provide additional results that did not fit in the main paper for ablation studies. For instance, in addition to the cosine similarities presented in figure 1, we provide the individual values of RSI and EB.

We provide in Figure 6 the cosine similarities for different batch sizes on CIFAR-10, as a complement to Figure 1 to study the impact of batch size RSI and EB.

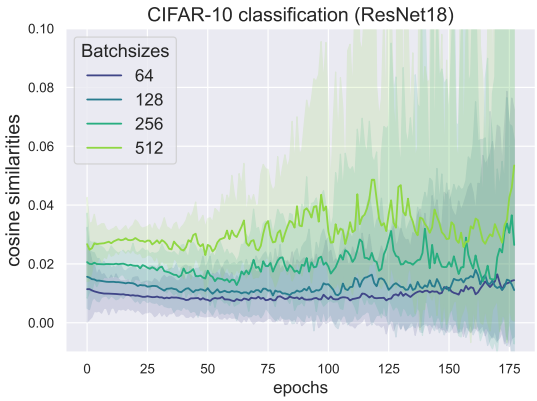

Figure 6. cosine similarities measured during the training of a ResNet-18 on CIFAR-10, for batchsizes ranging from 64 to 512.

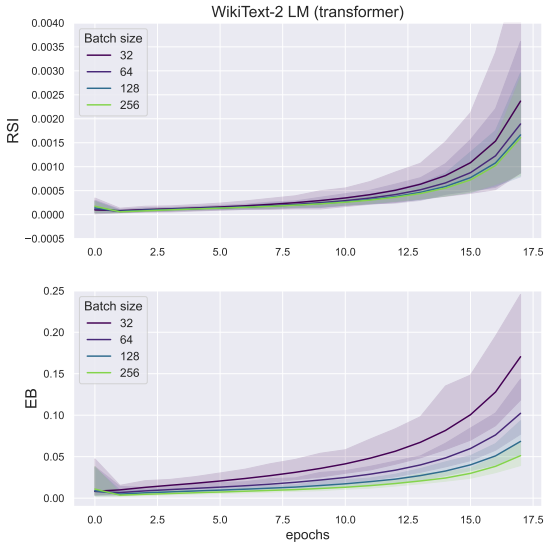

Figure 7. RSI and EB throughout training of a transformer on WikiText-2 with different batch sizes. This figure is complementary to figure 1.

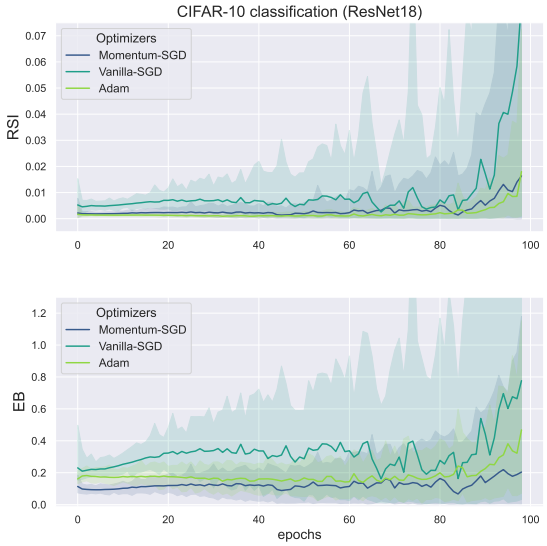

Figure 8. RSI and EB throughout training for the training of a ResNet18 on CIFAR-10 with different optimizers. This figure is complementary to figure 1.

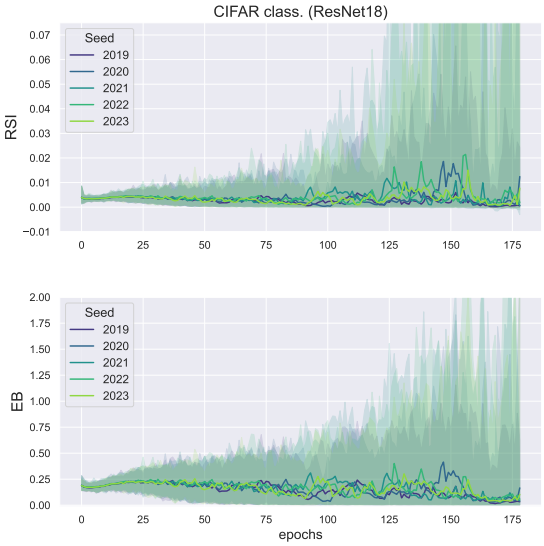

Figure 10. RSI and EB throughout training for the training of a ResNet18 on ImageNet with different random seed. This figure is complementary to figure 4.

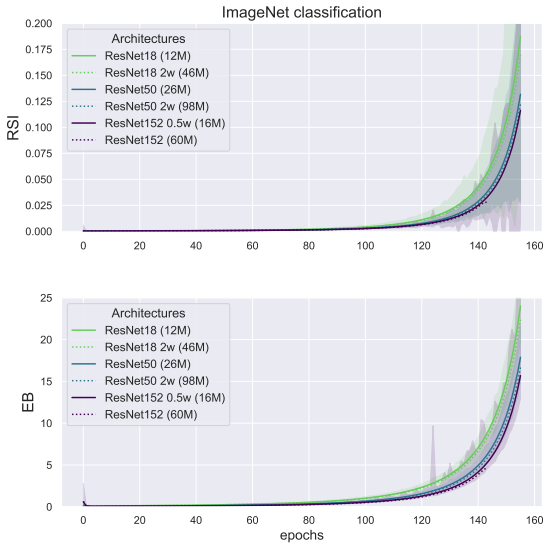

Figure 9. RSI and EB throughout training for the training of different ResNet architectures on ImageNet. This figure is complementary to figure 1.

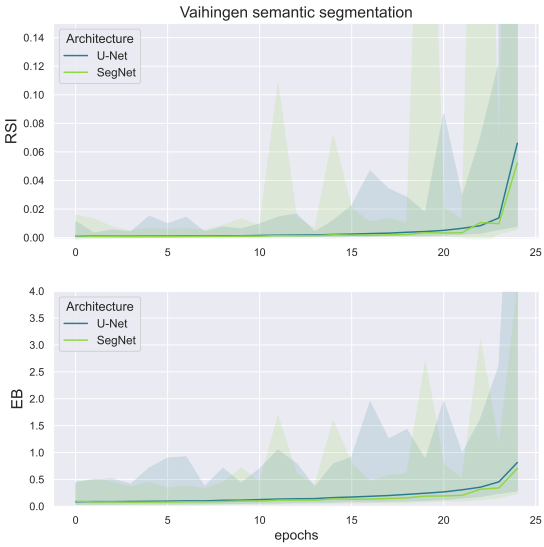

Figure 11. RSI and EB throughout training for two different architectures on Vaihingen semantic segmentation. This figure is complementary to figure 1.

C. Additional figures

In this section, we introduce additional figures supporting claims or conjectures made in the main paper.

Figure 24 shows the evolution of $\|w_t - w^\star\|_2$ throughout training. We can see Adam traverses a larger distance than Vanilla SGD and Momentum SGD, and evolves as a more regular pace. We believe this could be a factor in the lower cosine similarities exhibited by Adam in Figure 1.

Figure 12 indicates the value of $\|w_t - w^\star\|_2$ over training in the three settings of Figure 2. An important remark is that due to the cosine decreasing learning rate schedule, in the case of ImageNet, this distance becomes negligible in the last 25 epochs. This raise precision issues as discussed in section 4.1. Since w_t is subject to negligible variations in the last 25 epochs of the ImageNet experiment, Figures 1 and Figure 2 omit the epochs after 155 in the case of ImageNet, in order to improve readability and focus on meaningful settings.

Figure 14 and Figure 15 show the level lines of the synthetic functions introduced in Section 6 as example of irregular behavior for RSI, EB and γ.

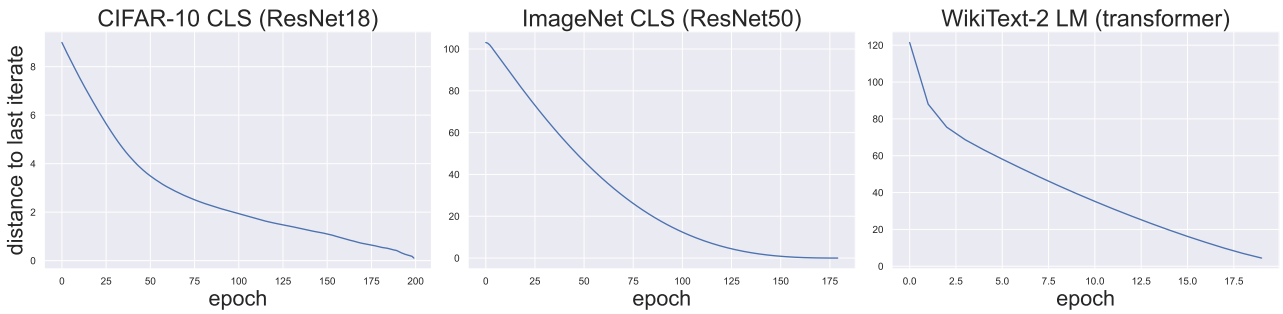

Figure 12. $\|w_t - w^\star\|_2$ over training in the three different settings of Figure 2.

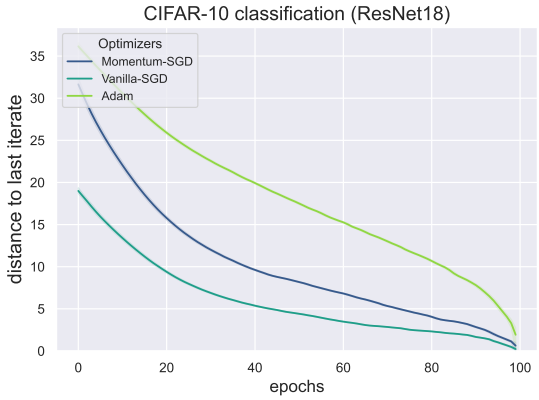

Figure 13. $\|w_t - w^\star\|_2$ over training of a ResNet18-1 on CIFAR-10, with different optimizers.

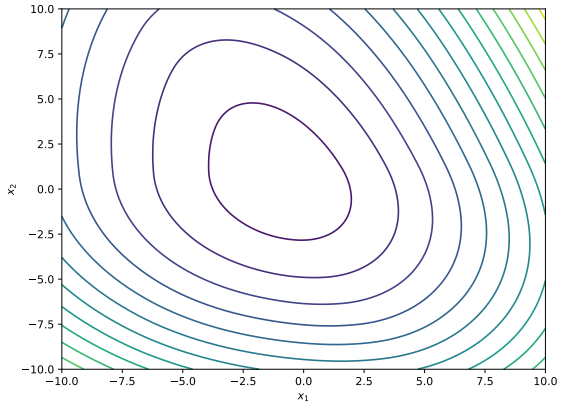

Figure 14. Level sets for the asymmetric linear model from Section 6, averaged over minibatches. The loss landscape is nicely convex, but the irregular behaviors observed in Figure 5 (left) result from the stochasticity induced by optimizing over minibatches.

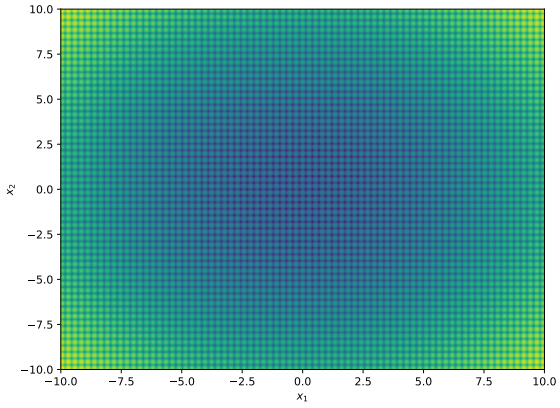

Figure 15. Level sets for the sinusoidal mixture from Section 6. The optimization is done deterministically (full batch), but the irregular behaviors observed in Figure 5 (left) result from the obvious non-convexity of the function.

D. Plausible causes

In this section, we examine potential factors that might be contributing to the remarkable geometric regularity observed via RSI and EB within the loss landscapes of neural networks. These are conjectural in nature, and we advocate for more rigorous investigation in future work to substantiate these propositions.

• **Architectural Characteristics:** The deep learning landscape has witnessed a plethora of architectural enhancements since its inception. Notably, ResNets incorporate advanced features such as skip-connections and batch normalization (Ioffe & Szegedy, 2015b), which were found to simplify the structure of the loss landscape (Santurkar et al., 2018a; Li et al., 2018). It is plausible that such favorable geometric attributes could play a significant role in the success of neural network architectures. Therefore, our observations may be more a byproduct of the selection of high-performing networks rather than an universal characteristic.

Nonetheless, Figure 16 provides an interesting comparison of cosine similarities derived from the training of a wide 4-layer Perceptron (MLP) with ReLU activations and a double-width ResNet-18. Despite a similar parameter count, these two architectures exhibit a considerable performance gap. Intriguingly, not only does the MLP not exhibit inferior geometrical properties, but it actually shows greater regularity in cosine similarity compared to the ResNet-18. This result suggests that the beneficial geometry of neural loss landscapes is not simply a consequence of extensive architectural tuning, but potentially a more intrinsic feature. Nonetheless, prior work concluded that skip connections significantly simplify the loss landscape at higher depth (Li et al., 2018).

• **High Dimensionality:** Our conjecture is that the primary contributor to the regular patterns observed by RSI, EB, and γ is the large dimensionality of neural loss landscapes. Assuming that a significant number of dimensions maintain a degree of independence, even when the gradient occasionally points in the 'wrong direction' in certain dimensions, this can be offset by averaging over a sufficiently vast number of dimensions. However, formalizing such an effect is challenging due to the evident dependencies between dimensions.

• **Properties of Real-World Data:** Lastly, the geometric simplicity of optimization paths might be influenced by inherent properties of real-world data distributions. For instance, it is commonly postulated that unstructured data from real-world applications resides within lower-dimensional manifolds. Analogous properties could be pivotal in shaping loss landscapes, which appear more benign than what worst-case scenarios might suggest.

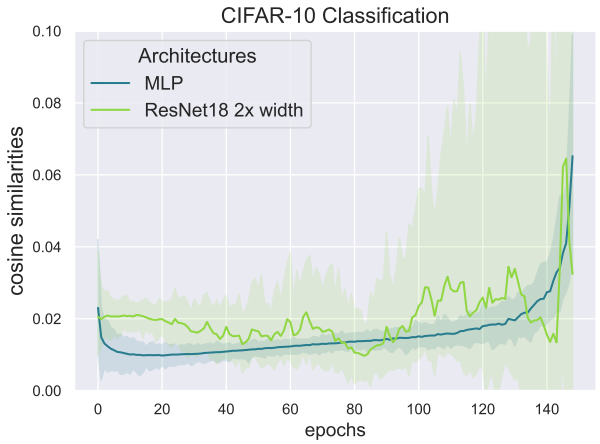

Figure 16. Comparison of cosine similarities derived from a CIFAR-10 classification task employing two distinct architectures: a straightforward Multi-Layer Perceptron (MLP) and a more complex ResNet-18 structure.

E. Additional Experiments

In this section, we introduce additional experimental results to support our claim made in the main paper.

E.1. Image Classification

To further investigate the performance of different architectures, MobileNet-V2 (Sandler et al., 2018) and VGG (Simonyan & Zisserman, 2014) were incorporated into the image classification task. For the CIFAR-10 dataset, models were trained for 400 epochs using Momentum SGD as the optimizer. The training parameters were set as follows: a batch size of 256 and a learning rate of 0.001.

VGG-11, VGG-19, MobileNet-V2 on CIFAR10:

The performance of VGG-11, VGG-19, and MobileNet-V2 on CIFAR10 aligns with our overall observations. VGG-19 and MobileNet-V2 exhibit highly stable positive γ values, with minimal variation between them. VGG-11, on the other hand, displays a slightly higher average γ value compared to VGG-19 and MobileNet-V2. However, VGG-11 demonstrates a significantly greater variance in cosine similarity across training epochs, with a wider gap between the minimum and maximum values observed compared to its counterparts. Despite this variance, VGG-11 remains relatively stable overall. Notably, in the final 5% of training epochs, VGG-11 experiences a few instances of negative cosine similarity, a phenomenon not observed in VGG-19 or MobileNet-V2.

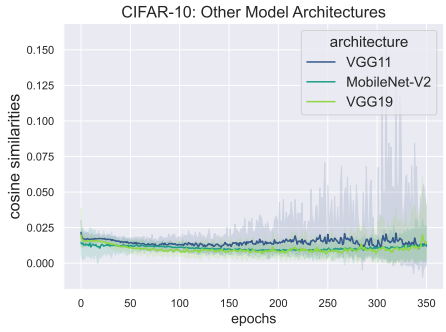

Figure 17. Comparison of cosine similarities derived from a CIFAR-10 classification task employing three additional architectures: VGG-11, VGG-19, and MobileNet-V2.

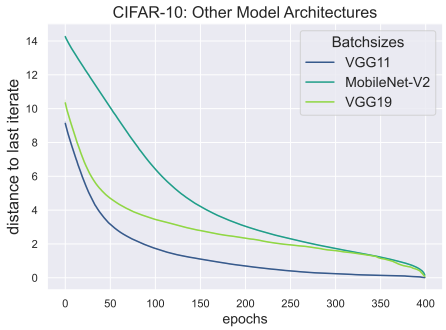

Figure 18. $\|w_t - w^\star\|_2$ over training of a VGG-11, VGG-19 and MobileNet-V2 on CIFAR-10.

MobileNet-V2 on ImageNet:

On ImageNet, MobileNet-V2 exhibits similar behavior to its ResNet counterparts, demonstrating consistent performance in line with our prior observations. Notably, MobileNet-V2 displays stable positive γ values metrics that remain consistent across minibatches but exhibit a gradual increase over training epochs.

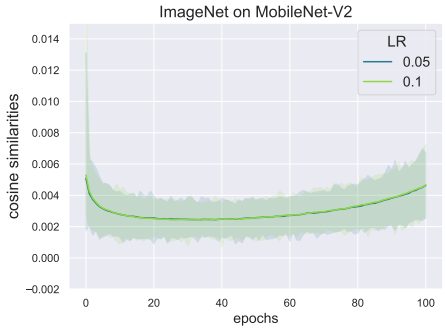

Figure 19. Comparison of cosine similarities derived from an ImageNet-1K classification task employing an additional architecture: MobileNet-V2.

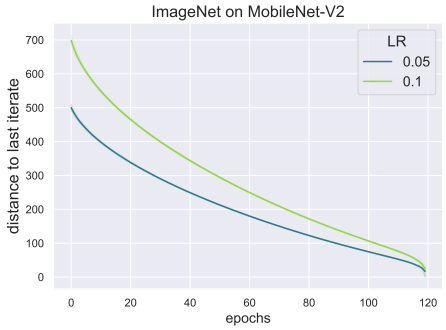

Figure 20. $\|w_t - w^\star\|_2$ over training of a MobileNet-V2 on ImageNet-1K.

E.2. Language Modeling

To further assess the performance of our language modeling approach, we introduce validation experiments on two additional datasets: the Penn Treebank dataset (Marcus et al., 1993) and WikiText-103 (Merity et al., 2016).

For the Penn Treebank dataset, we utilize the processed version provided by Mikolov et al. (2010). This dataset, widely recognized for evaluating sequence labeling models, is partitioned as follows: sections 0-18 for training (38,219 sentences, 912,344 tokens), sections 19-21 for validation (5,527 sentences, 131,768 tokens), and sections 22-24 for testing (5,462 sentences, 129,654 tokens).

WikiText-103, a collection of over 100 million tokens extracted from verified Wikipedia articles, offers a significantly larger dataset compared to WikiText-2, being more than 55 times larger. It also features a more extensive vocabulary and consists of complete articles, making it ideal for evaluating models that leverage long-term dependencies.

For both training scenarios, the Adam optimizer was employed with a fixed learning rate of 0.0001. The batch size was set to 256 for Penn Treebank training and 32 for WikiText-103 training. A training budget of 20 epochs was allocated for Penn Treebank and 2 epochs for WikiText-103.

Transformer for Language Modeling on Penntreebank:

On the Penn Treebank language modeling task, the transformer models exhibit performance characteristics consistent with those observed on Wikitext-2.

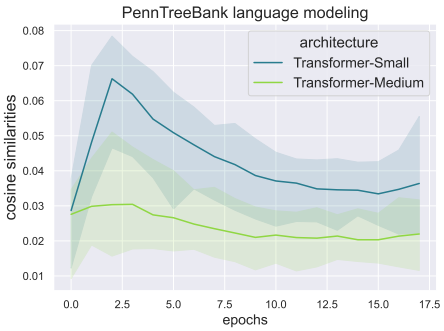

Figure 21. Comparison of cosine similarities derived from a language modeling task employing an additional dataset: Penn Treebank Dataset.

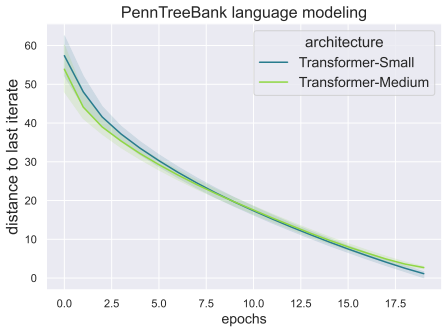

Figure 22. $\|w_t - w^\star\|_2$ over training of transformer model on Penn Treebank.

Transformer for Language Modeling on WikiText-103:

To further validate the findings of our numerical experiments on natural language processing tasks, we extend our analysis to include WikiText-103 in addition to WikiText-2.

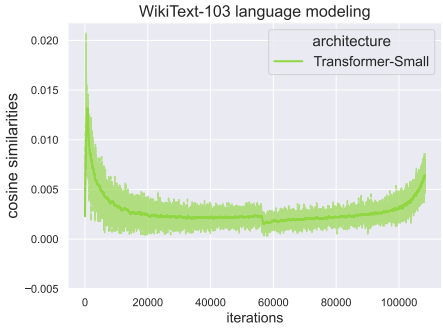

Figure 23. Comparison of cosine similarities derived from a language modeling task employing an additional dataset: WikiText-103 Dataset.

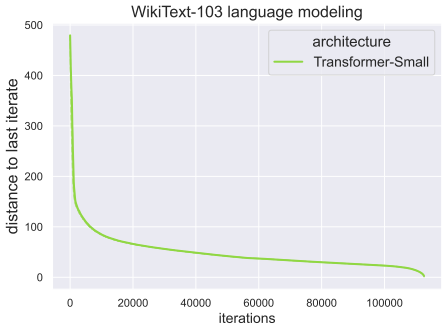

Figure 24. $\|w_t - w^\star\|_2$ over training of transformer model on WikiText-103.