# Edge of Stochastic Stability: Revisiting the Edge of Stability for SGD

**Anonymous authors**
Paper under double-blind review

## Abstract

Recent findings by Cohen et al. (2021) demonstrate that during the training of neural networks with full batch gradient descent at a step size of $\eta$, the sharpness—defined as the largest eigenvalue of the full batch Hessian—consistently stabilizes at $2/\eta$. These results have significant implications for generalization and convergence. Unfortunately, this was observed not to be the case of mini batch stochastic gradient descent (SGD), thus limiting the broader applicability of these findings. We show that SGD trains in a different regime we call Edge of Stochastic Stability. In this regime, what hovers at $2/\eta$ is, instead, the average over the batches of the largest eigenvalue of the Hessian of the mini batch loss—which is always bigger than the sharpness. This implies that the sharpness is generally lower when training with smaller batches or bigger learning rate, providing a basis for the observed implicit regularization effect of SGD towards flatter minima and a number of well established empirical phenomena.

## 1 Introduction

Training algorithms are a key ingredient to the success of deep learning. Stochastic gradient descent (SGD) (Robbins & Monro, 1951), a stochastic variant of gradient descent (GD), has been effective in finding parameters that yield good test performance despite the complicated nonlinear nature of neural networks.

Full batch GD and its adaptive versions have been shown to optimize in regime of instability Xing et al. (2018); Jastrzębski et al. (2020); Cohen et al. (2021; 2022). Precisely, if the step size is $\eta > 0$, the highest eigenvalue of the full batch Hessian, also called full batch sharpness–which we will denote as FullBS–grows until $2/\eta$ in a first phase of training called progressive sharpening and hovers around that value, subject to small oscillations. This regime is called Edge of Stability. This instability does not damage convergence when training neural networks. This is surprising as that would be the case of quadratics where if the curvature is higher than $2/\eta$ the optimization diverges.

This regime, called Edge of Stability by Cohen et al. (2021), implies that for a fixed learning rate, full batch sharpness is inherently limited by $2/\eta$, and constitutes a mode of instability.

**The work on EoS is about full batch methods.** The picture for full batch algorithms seems (empirically) clear, both for GD Cohen et al. (2021); Damian et al. (2023) and its adaptive and accelerated version Cohen et al. (2022). Unfortunately, neural networks are typically trained with the mini batch versions of these gradient based methods, and how Cohen et al. (2021) noticed and stated in the limitations of their work, what they observed is not the case of mini batch training. Precisely, to quote Cohen et al. (2021):

> [. . .] while the sharpness does not flatline at any value during SGD (as it does during gradient descent), the trajectory of the sharpness is heavily influenced by the step size and batch size (Jastrzebski et al., Jastrzębski et al. (2019; 2020)), which cannot be explained by existing optimization theory. Indeed, there are indications that the "Edge of Stability" intuition might generalize somehow to SGD, just in a way that does not center around the (full batch) sharpness.

> [. . .] In extending these findings to SGD, the question arises of how to model “stability” of SGD.

We believe we are the first ones to discover a notion of mini batch sharpness which acclimates to the learning rate and batch size in a way that reduces to the Edge of Stability in the particular case of full batch algorithms.

**What was empirically known for mini batch SGD.** In the case of mini batch algorithms, Jastrzębski et al. (2020) noticed that for SGD the phase transition happens earlier for smaller $\eta$ or smaller batch size $b$, but they did not quantify when **(i)**. Cohen et al. (2021) noticed that SGD somehow acclimates to the Hyperparameters. However, they did not characterize how if not negatively, **(ii)** it usually does not "flatline" at any value of FullBS and **(iii)** if it stabilizes, that always happens at a level they could not quantify which is below the $2/\eta$ threshold. This scenario leaves the most basic questions open: *In what way the location of convergence of SGD acclimates to the choice of hyperparameters? What are the key quantities involved?* To be more specific, can we characterize the training phenomena in **(i), (ii)** above? What determines them? Does SGD train in an unstable regime?

**Edge of Stochastic Stability.** We essentially respond to all the questions above with what we believe is a general and neat characterization of the SGD dynamics. In particular, we establish a quantity of the average *Mini Batch Sharpness* (MiniBS) - the highest eigenvalues of the hessian of mini batch loss - which constitutes a generalization and a "drop-in" replacement of full batch sharpness in the case of SGD. Precisely, we introduce a regime analogous to and generalizing EoS which governs the dynamics of SGD, *Edge of Stochastic Stability*: where the average MiniBS hovers around the level of $2/eta$, with an essential gap between it and the Full Batch Sharpness. This essentially forces the latter to plateau at a lower level which depends on the size of the batch, thus providing an explanation for the above phenomena.

## 2 Related Work

### 2.1 Progressive Sharpening and Edge of Stability

**Progressive Sharpening.** Early work noticed that the loss local shape of the landscape changes rapidly at the beginning of the training LeCun et al. (2012); Keskar et al. (2016); Achille et al. (2017); Jastrzębski et al. (2018); Fort & Ganguli (2019). A number of paper noticed growth of different estimators of the FullBS in the early training Keskar et al. (2016); Sagun et al. (2016); **?**); Jastrzębski et al. (2019). Later, Jastrzębski et al. (2020); Cohen et al. (2021; 2022) managed to make this precise, noticing that along the trajectories of SGD, gradient descent, and full batch Adam the FullBS usually steadily increases, often after a small number of steps of decrease. This phenomenon, was called progressive sharpening by Cohen et al. (2021). This is exactly what we reconfirm in our experiments across different models, losses, and datasets. The reason seems to be that in that regime the step size is comparably small and all these gradient based methods closely follow gradient flow, which steadily increases the FullBS Jastrzębski et al. (2020); Cohen et al. (2021). There is, thus, an empirical agreement on the fact that progressive sharpening is a feature of the loss landscape. To the knowledge of the authors there are no widely accepted explanations of progressive sharpening, we refer the reader to them for further discussion **?**Agarwala et al. (2023).

**A phase transition.** A number of early studies Goodfellow et al. (2016); Li et al. (2019); Jiang et al. (2019); Lewkowycz et al. (2020) showed that an initial large learning learning rate often improve generalization at the expenses of initial loss reduction. This was explained by Jastrzębski et al. (2020), who noticed that the initial progressive sharpening phase usually stops with a sudden phase transition, they named break-even point. This phase transition is considered to be, unlike progressive sharpening, a phenomenon induced by the gradient based algorithm, not by the landscape. Jastrzębski et al. (2020); Cohen et al. (2021; 2022) indeed showed that the phase transition comes at different points for different algorithms. Jastrzębski et al. (2020) showed that in the case of mini batch SGD this phase comes earlier for smaller step sizes and earlier for smaller batch sizes, without quantifying it. Cohen et al. (2021) showed indeed that it comes at the instability threshold for the optimization algorithm for full batch GD and, later Cohen et al. (2022) for full batch Adam. We manage to quantify the value of the instability threshold for mini batch SGD, thus characterizing when the phase transition happen for SGD.

**Full batch Edge of Stability.** At this points SGD, GD, and full batch Adam are understood to enter a different, oscillatory, regime that Cohen et al. (2021) called Edge of Stability and in which the FullBS sharpness stabilizes, entering a regime of small oscillations around a predefined value. The name is due to the fact that, in the case of full batch GD, the FullBS hovers at $2/\eta$ which is the stability threshold for optimizing quadratics Cohen et al. (2021). Cohen et al. (2021; 2022) observed that most of the training dynamics from that time on happen in this regime for full batch GD and full batch Adam, essentially defining the FullBS of the solution found. In section **??** we give 3 reasons why this EoS level at which FullBS is generally above $2/\eta$ and very rarely exactly at $2/\eta$. Precisely, (i) the gradient of the loss is not linear, thus requiring slightly bigger values or smaller values based on the higher order derivatives and (ii) the EoS is defined by the size of the Hessian along the gradients direction, not the size of FullBS alone.

**EoS and convergence.** There is a growing body of work analyzing the mechanism of EoS in training dynamics with GD. The idea is that when the gradients are a linear function of the parameters, if $\eta > \frac{2}{\lambda}$, even locally, you (locally) diverge. A very good exemplification of this fact is the case of one dimensional parabola, see Cohen et al. (2021). In the case of neural networks, surprisingly, convergence still happens even if $\eta \geq \frac{2}{\lambda}$, this is probably due to the non-standard geometry of the problem. Damian et al., Damian et al. (2023), proposed an explanation under some, empirically tested, assumptions of alignment of third derivatives and gradients. There has been a growing body of articles trying to figure this out and we refer the reader to **?**Ahn et al. (2022); **?**); Damian et al. (2023); Ahn et al. (2023); Zhu et al. (2023); Lyu et al. (2023) for more discussion about convergence in the regime of the Edge of Stability.

### 2.2 SGD, Hessian, and Generalization

**SGD finding flatter minima.** Our result is in nature a result about improved flatness by mini batch training. Indeed, the it explains why we can expect a smallest size for the eigenvalues of the full batch Hessian when training with a small batch. There has been a long line of work in this direction since Keskar et al. (2016) showed that SGD with smaller batch size finds minima with a smaller Hessian. A more recent paper, Jastrzębski et al. (2021), shows empirically that big learning rate SGD has an effect similar to penalizing the trace of the Fisher matrix, in image classification tasks. In similar settings, the Fisher matrix has been shown to approximate the Hessian during the training; in particular, there is an overlap between the top eigenspaces of the Hessian and its eigenspaces Jastrzębski et al. (2018); Martens (2020); Thomas et al. (2020). Furthermore, Jastrzębski et al. (2021) shows that, in practice, penalizing it consistently improves generalization, reduces memorization, and regularizes the trace of the final Hessian. Moreover, the advantages of penalizing the trace of the Fisher matrix are even stronger when in the presence of noisy labels. There exist multiple other studies along these lines, and in particular corroborate our findings.

**Sharpness and Generalization.** It has been observed that networks trained with SGD generalize better than GD, and smaller batch sizes often lead to better generalization performance (LeCun et al., 2012; Keskar et al., 2016; Goyal et al., 2017; Jastrzębski et al., 2018; **?**; Smith et al., 2021). Empirically, it has been observed that training with SGD results in flat minima (**?**Hochreiter & Schmidhuber, 1997). A number of works been argued that flatness of the minima is connected to the generalization performance (Neyshabur et al., 2017; Wu et al., 2017; Kleinberg et al., 2018; Xie et al., 2020; Jiang et al., 2019; Dinh et al., 2017), however we know only one theoretical result in that direction (**?**). Training algorithms aiming to find a flat minimum were developed and shown to perform well on a variety of tasks (**??**).

## 3 SGD trains at the Edge of Stochastic Stability

### 3.1 SGD does not train at the Edge of Stability

As Cohen et al. (2021) noticed, the highest eigenvalue of the Hessian (aka sharpness) stabilizes below $2/\eta$ for SGD, see also Figure 1. Precisely, it stabilizes lower and lower for smaller batches. This aligns with what empirically observed by Jastrzębski et al. (2020) who notice that SGD deviates from the trajectory of small learning rate GD or gradient flow earlier and earlier for larger learning rate orsmaller batch size. Precisely, what we observe is that (1) sharpness stabilizes also when training with mini batches, (2) sharpness for batch size $b_1$ stabilizes at a value that is always smaller than the one with batch size $b_2$, for $b_1 < b_2$. In particular, sharpness doesn't constitute a consistent metric that informs the dynamics of SGD.

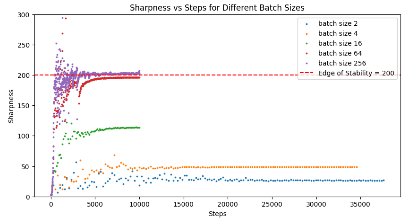

Figure 1: For SGD, sharpness plateaus under the EoS level: the lower the batch size, the lower it plateaus

### 3.2 Mini Batch Sharpness

In a similar manner to Cohen et al. (2021), we discover a different quantity that stabilizes during training with mini batch SGD. In particular, this quantity is the highest eigenvalue of the Hessian of the loss w.r.t. network parameters computed on the mini batch:

$$\lambda_{\max}\left[\nabla_\theta^2 L(\theta, B)\right] = \lambda_{\max}\left[\nabla_\theta^2\left(\frac{1}{b}\sum_{(x_i,y_i)\in B} L(f(\theta; x_i), y_i)\right)\right]$$

where $\lambda_{\max}[\cdot]$ denotes the largest eigenvalue, $\nabla_\theta^2 L(\theta, B)$ is the Hessian matrix of the loss computed over the mini batch $B$.

We call this quantity *Mini Batch Sharpness* (MiniBS), as opposed to Full Batch Sharpness (FullBS), referred to as just sharpness in Cohen et al. (2021). In particular, MiniBS is inherently of the same nature as FullBS, with the only difference being the "part" of the loss it is computed on - the mini batch loss for the former, and full-batch (aka full empirical) loss for the latter. Yet, this difference proves to be crucial in being descriptive of the SGD training dynamics. Furthermore, MiniBS stands as a generalization of FullBS, for when the batch size approaches the full dataset size, the MiniBS and FullBS become closer and closer - being equal if we are doing full batch SGD, aka GD. Lastly, FullBS serves as a strict lower bound for MiniBS - a property we will address below, together with additional properties of MiniBS. One of the specifics of MiniBS is its dependence on the draw of the batch, which introduces a lot of variance into the quantity. Instead, the quantity we are focusing on is the expectation over the batch draw of the MiniBS, referring to it as average MiniBS:

$$\mathbb{E}_B\left[\lambda_{\max}\left[\nabla_\theta^2 L(\theta, B)\right]\right]$$

### 3.3 Plateauing of MiniBS

The primary insight of our study is that the average MiniBS in SGD serves a role analogous to FullBS in GD. In both scenarios, these quantities plateau at a level around $2/\eta$, providing a comparative measure of stability under different optimization strategies. Figure 2 illustrates that the average MiniBS tends to stabilize slightly above $2/\eta$, akin to the behavior of FullBS in GD as detailed in Cohen et al. (2021).

As previously noted, FullBS in SGD does not provide significant insights, as its stabilization level is heavily dependent on the batch size and is much lower than $2/\eta$. Crucially, the average MiniBS's stability level exhibits some dependency on batch size, it is markedly less sensitive to the changes of batch size. Together with the fact that average MiniBS serves as a generalization of FullBS, these observations leads us to propose that MiniBS acts as a "critical" quantity in SGD, akin to the role of FullBS in GD.

### 3.4 Edge of Stochastic Stability

Building on the fact that MiniBS serves as an generalization of FullBS in the case of SGD, we can propose that SGD essentially trains in a regime analogous to Edge of Stability, which we call *Edge of Stochastic Stability* (EoSS). Precisely, we establish the following conjecture:

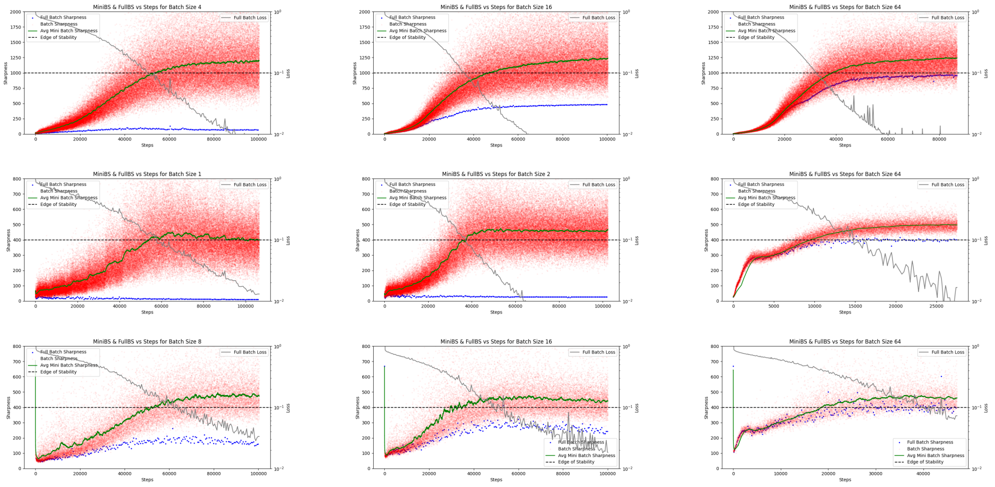

Figure 2: Comparison between: MiniBS (red dots), average MiniBS (green line), FullBS (blue dots). *Top:* MLP, 4 hidden layers, hidden size 512 *Middle:* CNN, 5 layers *Bottom:* ResNet-14; All trained on a 4k subset of CIFAR-10

*SGD tends to train in an instability regime we call Edge of Stochastic Stability. It is characterized by the fact that after a phase of progressive sharpening, the average Mini Batch Sharpness reaches a stability level of $2/\eta$ or slightly above, and hovers there.*

We further discuss the suitability of EoSS as an extension of EoS in Appendix B. In particular, we talk about the similarity of the EoSS and EoS from the point of view of "limiting behavior" and connection to quadratics, rather than just simple oscillations.

### 3.5 Average MiniBS and effect on FullBS

FullBS serves as a strict lower bound for MiniBS values, as further discussed in 6 and visible in the point cloud in Figure 2. In particular, depending on the exact mini batch size, there is often a significant gap between average MiniBS value and FullBS values, as clear from the plots. Precisely, the smaller the batch size, the bigger the gap between the two, 6. Combined with our conjecture that average MiniBS is plateauing at the EoSS level of slightly above $2/\eta$, this means that the FullBS is inherently being "forced" to plateau at a level below the EoS. This level is therefore being determined by the gap between the average MiniBS and FullBS, and is thus determined by the batch size. Precisely, the smaller the batch size, the lower the FullBS plateau - as confirmed by the FullBS lines on the Figure 2. This essentially explains the previously unexplained phenomena of why the FullBS plateaus lower for SGD and is dependent on the batch size.

## 4 The 2 Phases of Progressive Sharpening

Progressive sharpening (PS) was defined by Cohen et al. (2021) as *"When training neural networks, it seems to be a general rule that so long as the sharpness is small enough for gradient descent to be stable ($< 2/\lambda$, for vanilla gradient descent), gradient descent has an overwhelming tendency to continually increase the sharpness."* This concept, first posited as a general rule by Jastrzębski et al. (2020) in their Assumption 4, builds upon empirical findings from earlier studies Jastrzębski et al. (2018; 2019). We analyze and document here the phases of progressive sharpening. Not only we overwhelmingly empirically confirm the presence of this phase, but we provide a clear framework to understand this phase: we reveal that the PS phase consists of two smaller subphases. This particularly highlights the differences between full batch and mini batch stochastic gradient descent (SGD): progressive sharpening is not about the FullBS but it is inherently about the MiniBS. As expected, the description of full batch PS of Cohen et al. (2021) turns out to be a particular edge-case of our characterization.

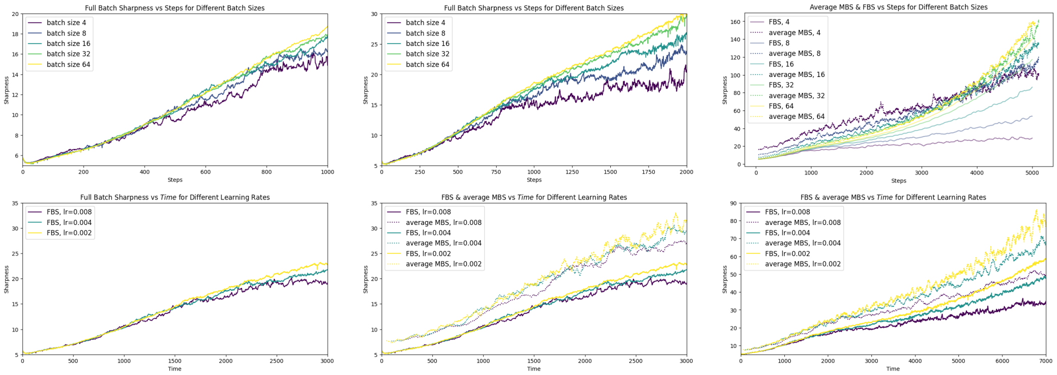

Figure 3: Progressive sharpening phases across different hyperparameter choices, highlighting how the sharpening is influenced by batch size and learning rate adjustments. *Setting:* MLP with 4 hidden layers and width 512, trained on 4k subset of Cifar10 with MSE. *Bottom:* The x axis is scaled according to lr used, thus "time"

### 4.1 First PS Phase

Our experiments indicate that the early training stages (ranging from 10 to 500 steps, depending on learning rate and initialization conditions) *exhibit a unified behavior across training dynamics for different hyperparameters*, irrespective of step size and batch size. More precisely, the trajectories display a consistent increase of both FullBS and MiniBS, often after a very small (5 to 50) number of steps in which they both slightly decline at the beginning of the training. These findings align with with the observations of Jastrzębski et al. (2019; 2020); Cohen et al. (2021).

### 4.2 Second PS Phase

Contrary to full batch case, where the entire progressive sharpening happens in the first phase, mini batch SGD exhibits a second distinct subphase of PS. Notably, the transition to this phase occurs earlier with smaller batch sizes and learning rates, as shown by Jastrzębski et al. (2020). From the case of full batch GD we would expect this phase transition to happen as the MiniBS approaches the EoSS. The trajectories, however, deviate from the full batch GD and gradient flow dynamics much earlier than when EoSS plateaus. In this intermediate phase between the initial progressive sharpening and the EoSS plateau, the rate of sharpness increase varies with batch size. The dynamics in this phase is inherently about the progressive sharpening of the MiniBS, as we can see in Figure fig. 3, top-right plot. Notice how it is the MiniBS that is evolving consistently no matter the batch size, while FullBSs significantly diverge, further indicating that FullBS is not the quantity driving the mini batch SGD dynamics, but rather MiniBS is. Further, notice in the bottom row the similarity of the effect of changes in learning rate on the average MiniBS and FullBS, thus showing that MiniBS responds to lr changes according to the theories established for FullBS by the aforementioned works.

The observed slower PS rate for smaller batch sizes corroborates mathematical theories from the implicit regularization literature that applies to big learning rate (Damian et al. (2021); Smith et al. (2021); Beneventano (2023)). After the completion of the first PS phase, the loss has typically reduced substantially and the manifold of minima is nearly reached (see also section 4.2), the implicit bias of mini batch SGD—which generally opposes the PS tendency seen in gradient flow—becomes more pronounced, particularly at smaller batch sizes.

## 5 Determining the Value of MiniBS when it Plateaus

We try to deduce here at what level the MiniBS plateaus depending on step size $\eta$.We highlight three reasons for why the MiniBS and FullBS sometimes plateau higher than at $2/\eta$.

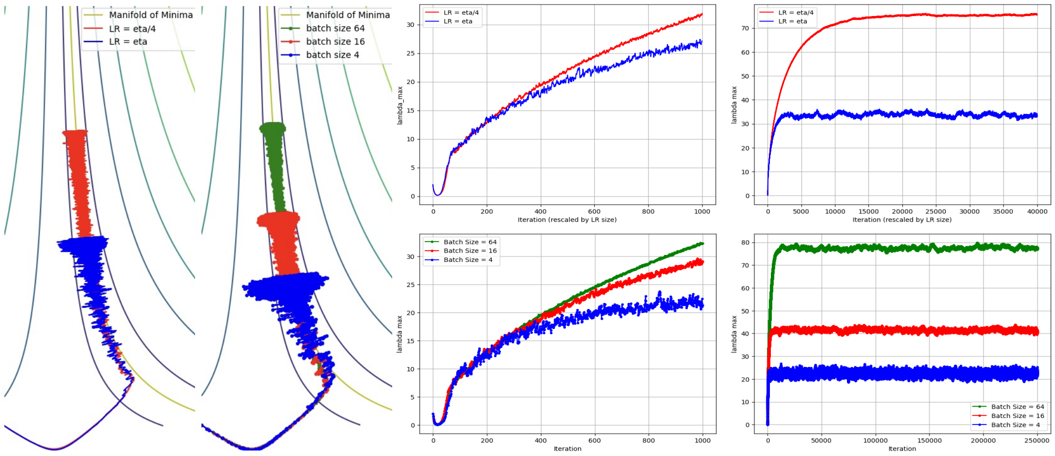

Figure 4: On the left we see how SGD with same batch and different learning rate and same learning rate but different batch acclimates in different areas along the manifold of minima. Precisely, traveling up along the manifold of minima, the sharpness grows quadratically. SGD with batch size $b$ acclimates the FullBS at a value that not only is higher for higher batch size, but as the batch size gets multiplied by $x$ it grows as $\sqrt{x}$. *Setting:* $2 \times 2$ matrix completion with $(a, b)$ and itself trasposed. This is the most simple model in which PS and EoSS are clearly visible and can be theoretically studied neatly.

### 5.1 Gradient-Hessian Alignment

Mini batch SGD is exactly gradient descent but on the landscape defined by the mini batch $B$. Expanding in Taylor the mini batch loss $L(\theta, B)$ at a parameter $\theta$ after one step of mini batch SGD performed with learning rate $\eta > 0$ we obtain

$$L(\theta_1, B) \quad = \quad L(\theta_0, B) - \eta \cdot \|\nabla L(\theta_0, B)\|^2 + \frac{\eta^2}{2} \cdot \nabla L(\theta_0, B)^\top \nabla^2 L(\theta_0, B) \nabla L(\theta_0, B) + O(\eta^3) \tag{1}$$

The Edge of Stability, as defined in Cohen et al. (2021), is the regime in which the last term on the right $\frac{\eta^2}{2} \cdot \nabla L(\theta_0, B)^\top \nabla^2 L(\theta_0, B) \nabla L(\theta_0, B)$ and the term in the middle of the RHS $\eta \cdot \|\nabla L(\theta_0, B)\|^2$ equalize. Training at the Edge of Stability does not mean that the higher eigenvalue of the Hessian is $2/\eta$, means that the quantity

$$\frac{2}{\eta} \quad = \quad \frac{\nabla L(\theta_0, B)^\top}{\|\nabla L(\theta_0, B)\|} \nabla^2 L(\theta_0, B) \frac{\nabla L(\theta_0, B)}{\|\nabla L(\theta_0, B)\|}. \tag{2}$$

Thus training at the EoS or EoSS is also a question of alignment, not only of size of the highest eigenvalue. Precisely, the highest eigenvalue of the Hessians typically coincides with the operatorial norm of the Hessian and is by definition bigger than the that product. Sometimes, the plateaus is slightly higher than $2/\eta$ for that reason.

### 5.2 The Model is not Linear

Moreover, when the model is linear and the loss is quadratic, then the loss is a quadratic function and the gradient is a linear function. In this case the EoS and EoSS happen when the Sharpness in Equation 2 is exactly $2/\eta$. Neural networks are not linear models and the thus to impose the LHS of Equation 1 $L(\theta_1, B)$ exactly equal to the value at initialization $L(\theta_0, B)$ it is usually needed to deal with the higher order terms of the Taylor expansion. **?** for example showed that for FullBS higher than $2/\eta$ gradient descent converges in shallow linear networks, although logarithmically fast. To establish both the convergence and the logarithmic speed it is necessary to investigate the higher order terms of the Taylor expansion. As an example, we show see Appendix **??** that for shallow linear networks example that we make below in Section **??**, the learning rate of the EoS step size is not $\lambda = 2/\eta$ but it is

$$\lambda \quad = \quad \frac{2}{\eta} + \eta \cdot L \pm \eta\sqrt{L}.$$

This may be the reason why we, and before us Cohen et al. (2021), see that very often the stabilization happens for MiniBS slightly bigger than $2/\eta$.

### 5.3 Fisher Matrix *vs* Hessian

In machine learning, the loss $L$ is generally a function of a batch of data $B$ and the parameterized model $f_\theta$. The Hessian of the loss can thus be decomposed as

$$\frac{1}{b}\sum_{(x_i,y_i)\in B}\underbrace{\nabla_\theta f_\theta(x_i)\cdot\nabla_f^2 L\big(f,(x_i,y_i)\big)\cdot\nabla_\theta f_\theta(x_i)^\top}_{\text{Relevant part}}+\underbrace{\nabla_f L\big(f,(x_i,y_i)\big)\cdot\nabla_\theta^2 f_\theta(x_i)}_{\text{Converges to zero as }\sqrt{L}.}\tag{3}$$

The right summand, when the loss is PL converges to zero together with the loss at the same speed as the square root of it. The left summand instead remains sizeable. For instance, in the case of MSE, the central term $\nabla_f^2 L\big(f,(x_i,y_i)\big)=1$, thus the whole left summand is $\frac{1}{b}\sum_{(x_i,y_i)\in B}\nabla_\theta f_\theta(x_i)\cdot\nabla_\theta f_\theta(x_i)^\top$. This explains how a smaller sharpness at convergence is related to a norm of the smaller Fisher matrix Jastrzębski et al. (2021); **?**. Jastrebsky et al., Jastrzębski et al. (2021), in an article which followed Jastrzębski et al. (2020), observed, along this line, how SGD implicitly regularizes the trace of the Fisher matrix.

The decomposition above in Equation 3 also allows us to study the alignment. Note that for one datapoint and MSE, the Fisher matrix above is

$$\nabla_\theta f_\theta(x_i)\cdot\nabla_\theta f_\theta(x_i)^\top.$$

This is an un-normalized projection matrix along the direction of the gradient of the function $\nabla_\theta f_\theta(x_i)$. Thus it has only one positive eigenvalue $\|\nabla_\theta f_\theta(x_i)\|^2$ with eigenvector the gradient and a kernel of dimension $n-1$. The gradient of the loss on one dapoint exactly aligns with the top eigenvector, indeed it is

$$\nabla_f L\big(f,(x_i,y_i)\big)\cdot\nabla_\theta f_\theta(x_i).$$

This alignment shows why the $\lambda$ needed to train at EoS/EoSS is generally close to $2/\eta$ and not much bigger. The gradient and the top eigenvalue indeed are the same on one datapoint and anyways have high cosine similarity later.

## 6 On Largest Eigenvalues of Sums of Matrices

In this section we establish mathematically:

- That MiniBS is always bigger than FullBS.
- How MiniBS scales with the batch size. Precisely:
  - That MiniBS increases as the batch size $b$ shrinks.
  - What we size can expect from the MiniBS-FullBS gap.

In particular, the following linear algebra results collectively enhance our understanding of the stability and scaling properties of the largest eigenvalues in the context of matrix sums.

### 6.1 Ordering the Largest Eigenvalues.

The largest singular value of the Hessian matrix derived from single data points is positive. This observation is crucial in establishing the following well-known property of matrix eigenvalues.

**Lemma 1.** *Let $m,n\in\mathbb{N}$ and consider $m$ matrices $M_1,M_2,\ldots,M_m\in\mathbb{R}^{n\times n}$ satisfying $\lambda_{\max}>|\lambda_{\min}|$. Then, the largest eigenvalue of their sum satisfies*

$$\lambda_{\max}\left(\sum_{i=1}^m M_i\right)\ \le\ \sum_{i=1}^m \lambda_{\max}\left(M_i\right)\tag{4}$$

*with equality only if all $M_i$ are identical.*

This lemma is a direct consequence of the convexity of the operator norm in matrices. In our setting, it implies that with non-identical matrices, the maximum eigenvalue of the sum is strictly less than the sum of the maximum eigenvalues of the individual matrices. To illustrate, consider eigenvalue sequences for batch sizes that are powers of four, though the result generalizes to any $b_1 < b_2$:

$$\lambda_{\max}^{1} \;<\; \lambda_{\max}^{4} \;<\; \lambda_{\max}^{16} \;<\; \lambda_{\max}^{64} \;<\; \lambda_{\max}^{128} \;<\; \lambda_{\max}^{512} \;<\; \ldots \tag{5}$$

### 6.2 Quantitative Analysis of Eigenvalue Scaling.

While the previous section establishes the order of MiniBS, it lacks quantification of their magnitudes. Random matrix theory help in bridging this gap at list for big batch sizes $b$.

**Lemma 2** (Matrix Bernstein Inequality)**.** *Let $n_1, n_2, b \in \mathbb{N}$, let $M_1, M_2, \ldots, M_b \in \mathbb{R}^{n_1 \times n_2}$ be independent random matrices satisfying $\mathbb{E}[M_i] = M$ and $\|M_i - M\| \le B$ for all $i$, let $v = \max\{\|\mathbb{E}[\sum_i M_i^\top M_i]\|, \|\mathbb{E}[\sum_i M_i M_i^\top]\|\}$ then for all $t > 0$*

$$\mathbb{P}\left(\left\|\tfrac{1}{b}\sum_i M_i - M\right\| \ge t\right) \quad \le \quad 2n \cdot \exp\left(-\frac{b^2 t^2/2}{v + Bbt/3}\right).$$

This lemma provides a probabilistic upper bound on the deviation of the largest eigenvalue as the batch size increases. For large batch sizes, where the Central Limit Theorem applies, the expected deviation diminishes as the inverse square root of the batch size.

**Lemma 3** (Informal, Scaling of Eigenvalue Deviations)**.** *For large batch sizes $b$ we have*

$$\lambda_{\max}^{b} \quad = \quad \lambda_{\max} \quad + \quad O\left(\frac{\log(n)}{\sqrt{b}}\right).$$

Note that the fact that the gap decreases as the square root of the batch size $\sqrt{b}$ agrees with our empirical observations and perfectly aligns with the observations

## 7 Implication: Noise injected full batch GD $\neq$ mini batch SGD

### 7.1 The Mini Batch Landscape Matters

Theoretical analyses of neural network optimization dynamics traditionally adopt two perspectives: (i) Online Case: Each sample serves as an independent, noised, unbiased estimator of the gradient from the theoretical loss. Here, stochastic gradient descent (SGD) is treated as performing noisy gradient descent on the actual expected risk derived from the theoretical distribution of the sample. (ii) Offline Case: A fixed finite dataset defines an empirical deterministic full batch landscape, with SGD executing noisy gradient descent upon this landscape.

Both scenarios presume the existence of a "real landscape" through which SGD navigates as a noisy, first-order method. If this was the case of the training dynamics of neural networks, then the quantity that would matter would be the FullBS, not influenced by the noise, while the gradients would show high levels of noise. However, an essential insight from our studies is that the specific landscape traversed at each step—the "mini batch landscape"—is crucial, rather than the generalized landscape (either full batch or theoretical). This distinction becomes evident when considering the variability of mini batch Hessians, their higher dimensional kernel, and their average sharpness set at $2/\eta$. If the sharpness chosen per batch is excessively high, it directs SGD trajectories sharply, highlighting that the average step taken across the mini batch landscape is more critical than the step on the average landscape.

### 7.2 Challenging SDEs as a Model for SGD

Our findings reveal that the dynamics of mini batch SGD with small batch sizes exhibit significant qualitative differences from those modeled by stochastic optimization trajectories on both full batch and theoretical landscapes. For instance, when employing continuous SDEs on these landscapes, the observed full batch Hessian showcases substantial discrepancies in: **(i)** The configuration of eigenvectors, as well as the magnitude and quantity of positive eigenvalues compared to those derived from mini batch processes. **(ii)** The orientation of these eigenvectors relative to the descent direction, which differs markedly from that observed in mini batch scenarios. This variation underscores a critical limitation of continuous SDE models in accurately predicting or replicating the nuanced behaviors of mini batch SGD, particularly when the batch size is small. The inherent differences highlighted by these observations strongly suggest that continuous time models may struggle to provide reliable insights into the pathways and outcomes of mini batch SGD unless implemented with considerably larger batch sizes.

The existing literature has already established that SDEs may not adequately model the dynamics of SGD. Notably, Yaida (2018) identified that the SDE approximation limits used are often ill-posed, indicating fundamental issues in their application to SGD dynamics. Later, Li et al. (2021) demonstrated that SDEs do not mimic SGD behaviors except under very specific conditions, such as extremely small learning rates in scale-invariant neural networks where variance dominates the gradient influence. On top of all the above, HaoChen et al. (2020) showed that various forms of noise—like those from Langevin dynamics or label noise—lead to convergence at different minima, suggesting diverse dynamics that SDE models may struggle to capture. Finally, Damian et al. (2021); Li et al. (2022); Shalova et al. (2024) have shown that the implicit regularization effects often attributed to diffusion are actually more likely to result from drifts caused by higher-order terms in Taylor expansions. In light of these findings, some recent theoretical efforts have begun to address the discrete nature and account for the different batch-specific Hessians, .e.g., Roberts (2021); Smith et al. (2021); Beneventano (2023), contrasting with much of the earlier literature that does not adequately capture these aspects.

However, the problems of continuous analysis as highlighted in the literature so fare were probabilistic in nature. While advancements in more sophisticated probabilistic tools and distributions, as suggested, e.g., by Ziyin et al. (2023), may offer potential solutions to them, the challenge of accurately incorporating the variability of batch-specific Hessians into the analyzes remains significant. This complexity underscores a critical gap in our understanding and modeling of SGD dynamics, raising questions about the feasibility of accurately predicting SGD behavior using existing analyzes.

## 8 Conclusions

This study advances our understanding of stochastic gradient descent (SGD) dynamics in training neural networks, particularly underlining the distinct behavior of mini batch SGD compared to full batch gradient descent. We introduce the concept of the Edge of Stochastic Stability (EoS), where the average largest eigenvalue of the mini batch Hessians—mini batch sharpness—stabilizes around $2/\eta$ unlike the full batch sharpness observed in previous studies. SGD's mini batch sharpness is consistently higher than full batch sharpness, underscoring its role in the implicit regularization that leads to flatter minima. Moreover, we properly characterize the behavior of SGD in the initial phase of the training, partly making more precise previous observations.

In summary, this work clarifies in what way mini-batch SGD does not align with full batch gradient descent behaviors. It challenges existing paradigms and encourages the development of new theories and methods that are more congruent with the stochastic nature of practical deep learning.

# 9 Sharpness Matters at a Batch Level

## 9.1 The landscape of mini batch loss

In the literature, it was generally assumed that, with a number of assumptions, SGD produces gradients that are unbiased estimates of the full gradients. After such an assumption, the analysis continues on the "true" loss landscape (and the empirical loss landscape is a good approximation of which). Now, in our analysis, we are also focusing on the loss landscape that is formed when we consider a the loss restricted to the instances of the mini batch. On one hand, this landscape is "temporary" and changes drastically from batch to batch, thus making it unsuitable for any sort of multi-step analysis. [This might consitture the reason why this loss landscape is rarely considered.] On the other hand, this mini batch loss landscape is the one that dictates the concrete step, and therefore constitutes an important object to consider for one-step analysis

## 9.2 The mini batch Sharpness

Now, although the aforementioned assumption that SGD produces an unbiased estimate of the empirical (and thus "true") is by definition true, this is not the case for the hessian of the loss. In particular, the aspect of the hessian that we care about is its highest eigenvalue, i.e. sharpness. Now, it turns out that the highest eigenvalue of the mini batch Hessian (the mini batch sharpness, MBS) doesn't provide an unbiased estimator of the highest eigenvalue of the full-batch Hessian (FBS). In particular, the expectation of MBS over the batches might be higher than the FBS: [the usual calculation goes here, side-by-side, comparing FBS and MBS] Notice that the first term is dominating, and would thus determine the sharpness of the network. [we need to add justification of why this is the case] Notice that the first term is a an average of (scaled) projections onto the gradients of the network. Now, notice that the operator norm of this operator (and therefore the sharpness of the loss) is non-increasing as we increase the batch size. This is simply follows from the fact that the norm of the sum is not greater than the sum of the norms. In one extreme case, if those gradients are perfectly collinear, then the norm of that operator does stay the same, no matter the batch size. On the other hand, if they are not perfectly collinear, the norm of the operator would reduce with the increase of batch size. In the extreme case of perfectly orthogonal vectors, the norm of the operator would drop as $1/batch_size$. Now, the more "non-collinear" the gradients are, the more the sharpness will drop as we increase the batch size.

## 9.3 Empirical evidence

As we can see, the network gradients are not highly collinear, although they are much less orthogonal than one would expect if the vectors were completely random

## 9.4 Sharpness vs second Taylor term

should we even include this? or should the whole thing be sent into appendix? One might notice that since we are doing one-step analysis of the batch loss landscape, and we have the access to the step direction, it might make sense to look at the product of gradient*Hessian*gradient. In particular, this would also take into account how aligned the gradient and the high eigenvalues of the Hessian.

The response to this is that by working with the sharpness (i.e. the spectral norm) of the hessian, we provide rather an upper bound for that second term, and work with an object that's consistent with full-batch sharpness. Now, by looking at the structure of the first term of Hessian and structyure of the gradient, one can notice that they are highly collinear, so working with the sharpness instead of the Taylor second term is a valid assumptiom.

Now, we also work with sharpness since it is a more "stable" notion, with less variation than gradient*hessian*gradient (something that would be even noisier because not only we have variability of the hessian, but there is also variability of direction)

We defer futher dicussion to the appendix

## 10 TRAINING DYNAMICS OF SGD

Just like in the case of gradient flow and GD, SGD also causes progressive sharpening, as illustrated as the increase of the sharpness as the training progresses in Figure. Now, as one can see from the same figure, it is clear that the in the the case of SGD, the sharpness of the loss landscape plateaus at a lowers level than with GD - and the smaller the batch size, the lower the that level is.

In particular, when we look at the GD, it trains at the edge of stability - i.e. the sharpness plateauing at the level of 2/eta. Yet, when we do SGD, the fact that sharpness plateaus at a lower level means that SGD doesn't train at the edge of stability.

Instead, we claim that SGD does indeed train at the edge of stability, with a modified, stochastic, notion of the Edge of Stability:

*SGD trains at the edge of stability which is defined by the average mini batch sharpness, rather than the full-batch sharpness of the empirical loss landscape*

That is, sharpness keeps increasing until it reaches a level such that the average mini batch sharpness is at edge of stability level of 2/eta, exactly as you can see in the scatter plot of MBSs in Figure ...

Combining this with the aforementioned result that the average MBS is higher than FBS, this provides a characterization for the phenomena that SGD seems not to train at the original notion of edge of stability. In particular, assuming that the average MBSs is the one that plateaus at the level 2/eta, and the FBS is strictly lower than the average MBS, it means that the FBS will plateau at a level lower than the edge of stability. Moreover, if combining with the previously established phenomenon that the gap between average MBS and FBS increases as we decrease the batch size, we get an explanation for the phenomenon that happens in the figure [] that the sharpness plateaus at lower levels as we decrease the batch size.

### 10.1 FURTHER EMPIRICAL EVIDENCE

Our original result was establish on a blah-blah trained on blah-blah. Yet, just like with the original EoS result, the results average MBS training at the edge of stability holds for a variety of architectures on a variety of datasets. In particular, we are working on MLP, simple CNN, as well as VGG and ResNet. It also holds for different learning rates, which would establish different levels of edge of stability, different batch sizes, which affects the average MBS size and different initalizations, which affects the initial sharpness. One can see the results in Figure [] and refer to Appendix [] for additional experiments. One can notice that in all the runs, the average MBS is the one that stays on the level of 2/eta, while the FBS is plateauing at a lower level, with the gap being determined by the batch size.

Now, it is important to note, just like with the original result of edge of stability, that our result is approximate. That is, just how in the case of GD in some settings the network would train at sharpnesses that are a slightly above the edge of stability level of 2/eta, in the case of SGD, it is not an exact average of MBSs that stays at the level of 2/eta.

### 10.2 EVEN FURTHER EVIDENCE

In the OG EoS paper, authors conduct additional experiements where after the training continues for a while at the edge of stability, they reduce the learning rate (thus increasing the EoS level), and the progressive sharpening continues until it reaches the new EoS level. We are conducting a reverse of this experiment, to showcase the effect that, from the point of view of edge of stability, increasing the batch size is somewhat equivalent to reducing the learning rate. That is, in the Figure you can notice that after the reduction of batch size (and thus reduction of average MBS), progressive sharpening continues until the new average MBS stays at the EoS level 2/eta. Overall, this indicates that batch size affects the sharpness specifically in the edge of stability regime, and not that in some initial phases

In reverse of these experiments, we also lower the batch size after we reach the EoS. This causes a divergence of loss [does it?], and just catapults the network to a completely different region, such that new (increased) average MBS is at or below the EoS. This proves that with SGD the average MBS indeed defines the edge of stability, in the same way as the FBS defines the EoS in GD - that is, if we start at a level that is significantly beyond the EoS, the training diverges, and gets catapulted to an entirely different region.

In combination to this, if we increase the learning rate so that 2/eta becomes below the average MBS, but still above the FBS, we see the same effect of the loss diverging [or the sharpness reducing]. This further supports the fact that it is specifically the average MBS training at the edge of stability

## References

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical Learning Periods in Deep Neural Networks, 2017. URL https://arxiv.org/abs/1711.08856v3.

Atish Agarwala, Fabian Pedregosa, and Jeffrey Pennington. Second-order regression models exhibit progressive sharpening to the edge of stability. In *Proceedings of the 40th International Conference on Machine Learning*, July 2023. URL https://proceedings.mlr.press/v202/agarwala23b.html.

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In *Proceedings of the 39th International Conference on Machine Learning*, June 2022. URL https://proceedings.mlr.press/v162/ahn22a.html.

Kwangjun Ahn, Sébastien Bubeck, Sinho Chewi, Yin Tat Lee, Felipe Suarez, and Yi Zhang. Learning threshold neurons via the "edge of stability", October 2023. URL http://arxiv.org/abs/2212.07469. arXiv:2212.07469 [cs, math].

Pierfrancesco Beneventano. On the Trajectories of SGD Without Replacement. *arXiv:2312.16143 [cs, math, stat]*, December 2023. doi: 10.48550/arXiv.2312.16143. URL http://arxiv.org/abs/2312.16143. arXiv:2312.16143 [cs, math, stat].

Jeremy M. Cohen, Simran Kaur, Yuanzhi Li, J. Zico Kolter, and Ameet Talwalkar. Gradient Descent on Neural Networks Typically Occurs at the Edge of Stability. *arXiv:2103.00065 [cs, stat]*, June 2021. URL http://arxiv.org/abs/2103.00065. arXiv: 2103.00065.

Jeremy M. Cohen, Behrooz Ghorbani, Shankar Krishnan, Naman Agarwal, Sourabh Medapati, Michal Badura, Daniel Suo, David Cardoze, Zachary Nado, George E. Dahl, and Justin Gilmer. Adaptive Gradient Methods at the Edge of Stability, July 2022. URL http://arxiv.org/abs/2207.14484. arXiv:2207.14484 [cs].

Alex Damian, Tengyu Ma, and Jason Lee. Label Noise SGD Provably Prefers Flat Global Minimizers. *arXiv:2106.06530 [cs, math, stat]*, June 2021. URL http://arxiv.org/abs/2106.06530. arXiv: 2106.06530.

Alex Damian, Eshaan Nichani, and Jason D. Lee. Self-Stabilization: The Implicit Bias of Gradient Descent at the Edge of Stability, April 2023. URL http://arxiv.org/abs/2209.15594. arXiv:2209.15594 [cs, math, stat].

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*, pp. 1019–1028. JMLR. org, 2017.

Stanislav Fort and Surya Ganguli. Emergent properties of the local geometry of neural loss landscapes, 2019. URL https://arxiv.org/abs/1910.05929v1.

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

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

Jeff Z. HaoChen, Colin Wei, Jason D. Lee, and Tengyu Ma. Shape Matters: Understanding the Implicit Bias of the Noise Covariance. *arXiv:2006.08680 [cs, stat]*, June 2020. URL http://arxiv.org/abs/2006.08680. arXiv: 2006.08680.

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

Stanisław Jastrzębski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three Factors Influencing Minima in SGD. *arXiv:1711.04623 [cs, stat]*, September 2018. URL `http://arxiv.org/abs/1711.04623`. arXiv: 1711.04623.

Stanisław Jastrzębski, Zachary Kenton, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. On the relation between the sharpest direc- tions of DNN loss and the SGD step length. 2019.

Stanisław Jastrzębski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The Break-Even Point on Optimization Trajectories of Deep Neural Networks. *arXiv:2002.09572 [cs, stat]*, February 2020. URL `http://arxiv.org/abs/2002.09572`. arXiv: 2002.09572.

Stanisław Jastrzębski, Devansh Arpit, Oliver Astrand, Giancarlo Kerg, Huan Wang, Caiming Xiong, Richard Socher, Kyunghyun Cho, and Krzysztof Geras. Catastrophic Fisher Explosion: Early Phase Fisher Matrix Impacts Generalization. *arXiv:2012.14193 [cs, stat]*, June 2021. URL `http://arxiv.org/abs/2012.14193`. arXiv: 2012.14193.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic Generalization Measures and Where to Find Them. *arXiv:1912.02178 [cs, stat]*, December 2019. URL `http://arxiv.org/abs/1912.02178`. arXiv: 1912.02178.

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. *arXiv preprint arXiv:1609.04836*, 2016.

Robert Kleinberg, Yuanzhi Li, and Yang Yuan. An Alternative View: When Does SGD Escape Local Minima?, August 2018. URL `http://arxiv.org/abs/1802.06175`. arXiv:1802.06175 [cs].

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

Sungyoon Lee and Cheongjae Jang. A new characterization of the edge of stability based on a sharpness measure aware of batch gradient distribution. In *International Conference on Learning Representations*, 2023. URL `https://api.semanticscholar.org/CorpusID:259298833`.

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:2003.02218 [cs, stat]*, March 2020. URL `http://arxiv.org/abs/2003.02218`. arXiv: 2003.02218.

Yuanzhi Li, Colin Wei, and Tengyu Ma. Towards explaining the regularization effect of initial large learning rate in training neural networks. In *Advances in Neural Information Processing Systems*, pp. 11669–11680, 2019.

Zhiyuan Li, Sadhika Malladi, and Sanjeev Arora. On the Validity of Modeling SGD with Stochastic Differential Equations (SDEs). *arXiv:2102.12470 [cs, stat]*, June 2021. URL `http://arxiv.org/abs/2102.12470`. arXiv: 2102.12470.

Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What Happens after SGD Reaches Zero Loss? –A Mathematical Framework. *arXiv:2110.06914 [cs, stat]*, February 2022. URL `http://arxiv.org/abs/2110.06914`. arXiv: 2110.06914.

Kaifeng Lyu, Zhiyuan Li, and Sanjeev Arora. Understanding the Generalization Benefit of Normalization Layers: Sharpness Reduction, January 2023. URL `http://arxiv.org/abs/2206.07085`. arXiv:2206.07085 [cs].

James Martens. New Insights and Perspectives on the Natural Gradient Method. 2020.

Konstantin Mishchenko, Ahmed Khaled, and Peter Richtarik. Random Reshuffling: Simple Analysis with Vast Improvements. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17309–17320. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/c8cc6e90ccbff44c9cee23611711cdc4-Abstract.html`.

Behnam Neyshabur, Srinadh Bhojanapalli, David Mcallester, and Nati Srebro. Exploring Generalization in Deep Learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 5947–5956. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/7176-exploring-generalization-in-deep-learning.pdf`.

Herbert Robbins and Sutton Monro. A Stochastic Approximation Method. 1951. URL `https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-22/issue-3/A-Stochastic-Approximation-Method/10.1214/aoms/1177729586.full`.

Daniel A. Roberts. SGD Implicitly Regularizes Generalization Error. *arXiv:2104.04874 [cs, stat]*, April 2021. URL `http://arxiv.org/abs/2104.04874`. arXiv: 2104.04874.

Levent Sagun, Leon Bottou, and Yann LeCun. Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond. November 2016. URL `https://openreview.net/forum?id=B186cP9gx`.

Anna Shalova, André Schlichting, and Mark Peletier. Singular-limit analysis of gradient descent with noise injection. *arXiv:2404.12293*, April 2024. URL `http://arxiv.org/abs/2404.12293`.

Samuel L. Smith, Benoit Dherin, David G. T. Barrett, and Soham De. On the Origin of Implicit Regularization in Stochastic Gradient Descent. *arXiv:2101.12176 [cs, stat]*, January 2021. URL `http://arxiv.org/abs/2101.12176`. arXiv: 2101.12176.

Valentin Thomas, Fabian Pedregosa, Bart Merriënboer, Pierre-Antoine Manzagol, Yoshua Bengio, and Nicolas Le Roux. On the interplay between noise and curvature and its effect on optimization and generalization. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pp. 3503–3513. PMLR, June 2020. URL `https://proceedings.mlr.press/v108/thomas20a.html`. ISSN: 2640-3498.

Lei Wu, Zhanxing Zhu, and Weinan E. Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes. *arXiv:1706.10239 [cs, stat]*, November 2017. URL `http://arxiv.org/abs/1706.10239`. arXiv: 1706.10239.

Zeke Xie, Issei Sato, and Masashi Sugiyama. A Diffusion Theory For Deep Learning Dynamics: Stochastic Gradient Descent Exponentially Favors Flat Minima. *arXiv:2002.03495 [cs, stat]*, November 2020. URL `http://arxiv.org/abs/2002.03495`. arXiv: 2002.03495.

Chen Xing, Devansh Arpit, Christos Tsirigotis, and Yoshua Bengio. A Walk with SGD, May 2018. URL `http://arxiv.org/abs/1802.08770`. arXiv:1802.08770 [cs, stat].

Sho Yaida. Fluctuation-dissipation relations for stochastic gradient descent. *arXiv preprint arXiv:1810.00004*, 2018.

Xingyu Zhu, Zixuan Wang, Xiang Wang, Mo Zhou, and Rong Ge. UNDERSTANDING EDGE-OF-STABILITY TRAINING DYNAMICS WITH A MINIMALIST EXAMPLE. 2023.

Liu Ziyin, Hongchao Li, and Masahito Ueda. Law of Balance and Stationary Distribution of Stochastic Gradient Descent, August 2023. URL `http://arxiv.org/abs/2308.06671`. arXiv:2308.06671 [cs, stat].

## A Determining the Value of FullBS when MiniBS Plateaus

We try to deduce here at what level the FullBS plateaus depending on step size $\eta$ and batch size $b$, assuming that MiniBS has plateaued at $2/\eta$.

## B On the Redefinitions of EoS for mini batch SGD

The question on how to redefine EoS phase for the mini batch case is not obvious. The defining features of the full-batch notion of Edge of Stability are that

1. There exists a phase in which FullBS is stable and present oscillations around the instability threshold.
2. **2.** The steps are such that the second and third order terms of the Taylor expansion of the loss $L(\theta)$ evolution cancel out. Indeed, $\theta_1$ after one step of full batch GD with step size $\eta > 0$ from $\theta_0$ is such that

$$L(\theta_1) \;=\; L(\theta_0) \;+\; \underbrace{\eta \cdot \|\nabla L(\theta_0)\| \;-\; \frac{\eta^2}{2} \cdot \nabla L(\theta_0)^\top \, \nabla^2 L(\theta_0) \, \nabla L(\theta_0)}_{\text{cancel out at the Edge of Stability}} + O(\eta^3). \quad (6)$$

3. The FullBS during this phase implicitly defines the sharpness of the minimum found by gradient descent.
4. **4.** This phenomenon is unexpected in the understood cases of optimization of quadratics.

Mini batch SGD with fixed step size is known to oscillate because of its inherent noise even in the case of quadratics at convergence. Precisely, at convergence, it oscillate around a minimum with size of the oscillations $\eta$ for the case with replacement or $\eta^2$ for the case without replacement, see Mishchenko et al. (2020) and references therein. SGD thus exhibits the properties 1. and 3. naturally, even for quandratics, not because of an instability due to the landscape, but because of an instability due to the noise. The only previous work connecting EoS and SGD present to our knowledge describe an effect which contains this natural effect due to SGD noise. The effect we discover and the generalization to mini batches that we give for edge of stability instead is about an instability due to the landscape, and thus about the properties **2.** and **4.** above. Edge of stability is about having a quantity that is around $2/\eta$; it is also about the canceling out in the equation above, but it is not only about the oscillations per se. We thus believe that this is the generalization of Edge of Stability for the mini batch setting that is interesting for understanding the training dynamics of deep learning and that naturally generalizes the one given by Cohen et al. (2021) for full batch GD and by Cohen et al. (2022) for full batch Adam. Indeed, EoSS reduces to EoS in the full batch case and our empirical observations are inherently of how our relevant notion of sharpness acclimates to learning rate and batch size. The only other work on extending the EoS for mini batch SGD is Lee & Jang (2023), but there the authors are rather presenting a non-consistent correction term to the GD EoS to get the EoS for SGD, without establishing its limiting factor or the influence on the dynamics of SGD.

## C Exemplification Through a Simplified Model

To elucidate what the size of the MiniBS, consider a simplified scenario involving a diagonal linear network trained on data from two orthogonal classes. Assume $(x, y) \in \mathbb{R}^2 \times \mathbb{R}$ is either $z_1 = \big((1,0),\ 1\big)$ or $z_2 = \big((0,1),\ -1\big)$ with probability $1/2$. We learn this data with a diagonal linear network and MSE, precisely where

$$f(x) = a^\top B \cdot x, \qquad a \in \mathbb{R}^2, \quad B \in \mathbb{R}^{2\times 2}.$$

Then with a diagonal initialization, gradient descent will converge almost surely to a neural network of the following kind

$$f(x) = (a_1, a_2) \cdot \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} \cdot x, \qquad \text{where} \quad |a_1 \cdot b_1| \;=\; |a_2 \cdot b_2| \;=\; 1.$$

At convergence, the spectrum of the Hessian on the data point $z_1$ is $\{\lambda_1, 0, 0, 0, 0, 0\}$, with $\lambda_1 := a_1^2 + b_1^2$, the Hessian on the data point $z_2$ is instead $\{\lambda_2, 0, 0, 0, 0, 0\}$, where $\lambda_2 := a_2^2 + b_2^2$, and the two eigenvectors for these two eigenvalues are orthogonal between each other. This implies that the Hessian of the full batch loss has spectrum $\{\lambda_1/2, \lambda_2/2, 0, 0, 0, 0\}$, while the Hessian on the mini batches of size one has either one of the spectra above.

This implies that

$$FullBS \quad = \quad \lambda_{\max}\left(\frac{1}{2}\mathcal{H}(z_1) + \frac{1}{2}\mathcal{H}(z_1)\right) \quad = \quad \max\left\{\frac{\lambda_1}{2},\, \frac{\lambda_2}{2}\right\} \tag{7}$$

This is smaller than the average largest eigenvalue of the mini batch Hessian which is

$$AMiniBS \quad = \quad \frac{1}{2}\lambda_{\max}(\mathcal{H}(z_1)) + \frac{1}{2}\lambda_{\max}(\mathcal{H}(z_2)) \quad = \quad \frac{\lambda_1}{2} + \frac{\lambda_2}{2}. \tag{8}$$

- **Smaller size:** Thus setting $FullBS$ equal to $\lambda$ means that the max between $\lambda_1$ and $\lambda_2$ is exactly $2\lambda$. Note that the fact that $a_1 \cdot b_1 = a_2 \cdot b_2 = 1$ and Cauchy-Schwartz imply that $\lambda_1, \lambda_2 \geq 2$. Setting $AMiniBS$ to $\lambda$ thus implies that the maximum between $\lambda_1$ and $\lambda_2$ is *at most* $2\lambda - 2$, generally smaller.
- **Higher alignment:** Moreover, we have that the gradient $\nabla f(z_i)$ on the data point $z_i$ exactly aligns with the eigenvector $v_i$ of the highest eigenvalue $\lambda_i$ of the Hessian in $z_i$. On the full batch, we are averaging them differently, precisely we have that there exist two constants $c_1, c_2$ such that the gradient is $\frac{c_1}{2}v_1 + \frac{c_2}{2}v_2$. Thus, where WLOG $\lambda_1 > \lambda_2$ we have the alignments

  $$\mathcal{H}(z_1) \cdot \nabla L(z_1) \;\sim\; c_1\lambda_1^2 v_1 \qquad \text{but} \qquad \mathcal{H} \cdot \nabla f \;\sim\; \frac{c_1}{2}\lambda_1^2 v_1 \tag{9}$$

  Thus one half of it (batch size divided by number of data points).

This shows that in the same point of the gradient, SGD perceives the largest eigenvalue of the Hessian bigger and more relevant to the gradient then GD.