Characterising Stochastic Gradient Descent Noise in Regularized Empirical Risk Minimization

Meshal Alharbi
MIT Computational Science and Engineering
meshal@mit.edu

Abstract

This project investigates the inherent noises in neural networks trained by Stochastic Gradient Descent (SGD) with weight decay regularization. We empirically investigate how SGD noise is affected by training hyperparameters such as batch size and learning rate. Further, we quantify the distribution of this noise when it is measured at the output of neural networks. Experiments are done on multiple network architectures and the MNIST and CIFAR10 datasets.

1 Introduction

It has been shown recently that there are inherent noises in neural networks trained using Stochastic Gradient Descent (SGD) and L2 regularization (weight decay), in the sense that the network's weights will not converge to a stationary point, even asymptotically [1]. In practice, neural networks are typically trained beyond the point of perfect accuracy on the training dataset [2, 3]. Thus, understanding the qualitative and quantitative features of this noise is crucial, especially when neural networks are used as input for other models or processes. This project aims to empirically investigate the "SGD noise" phenomenon and address some of the open questions in the literature.

The remaining of this section introduces our notation and formally defines the notion of SGD noise. Section 2 describes the setup of the empirical study. Section 3 contains our experiments and analysis. Finally, Section 4 concludes this report.

1.1 Preliminary

Let $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^k$ be the input and output spaces, respectively. A function f_W representing a neural network with trainable parameters W is a map from the input to the output space $\mathcal{X} \to \mathcal{Y}$. Given a balanced training dataset $S = \{(x_i, y_i)\}_{i=1}^m \in (\mathcal{X}, \mathcal{Y})^m$, we consider the problem of minimizing the regularized empirical risk:

$$\mathcal{L}_S^\lambda(f_W) = \frac{1}{m}\sum_{i=1}^{m} \ell(f_W(x_i), y_i) + \lambda \big\|W\big\|_2^2 \tag{1}$$

where ℓ is a loss function and $\lambda > 0$ is a predefined hyperparameter called weight decay. For ease of computation, we only consider binary classification tasks $y_i \in \{\pm 1\}$. Further, we focus on the minimization of the regularized squared loss [4]:

$$\mathcal{L}_S^\lambda(f_W) = \frac{1}{m}\sum_{i=1}^{m} (f_W(x_i) - y_i)^2 + \lambda \big\|W\big\|_2^2 \tag{2}$$

This objective is minimized using SGD. At each training iteration after initializing the weights randomly, we sample a subset $S_B = \{(x_i, y_i)\}_{i \in \mathcal{B}} \subset S$ uniformly at random, where $|\mathcal{B}| = B < m$ is the batch size. Then, the weights are updated according to:

$$W_{t+1} \leftarrow W_t - \eta \nabla_{W_t} \mathcal{L}_{S_B}^\lambda(f_{W_t}) \tag{3}$$

where η is the learning rate.

1.2 Stochastic Gradient Descent Noise

What we refer to by "SGD noise" is the inherent inability of the network's weights W trained using SGD to converge to a fixed stationary point, even asymptotically. This differs from the notion of "gradient noise" often associated with SGD [5, 6]. Galanti and Poggio [1] have shown that for networks trained with weight decay, SGD noise is persistent under a minimal set of assumptions. We rephrase their main results regarding SGD noise using our notation:

Proposition 1 ($\lambda > 0$) *Let ℓ be a differentiable loss function and f_W be ReLU neural network with trainable parameters W. Let W^* be a convergence point of mini-batch SGD for minimizing $\mathcal{L}_S^\lambda$. Then, either $f_{W^*} \equiv 0$ or $\{x_i\}_{i=1}^m$ are collinear vectors.*

Given that both convergence conditions are absurd, this concludes that SGD convergence is impossible in any practical scenario. Notice that this indicates that the recently observed phenomenon known as Neural Collapse (NC) [2] will not happen exactly when we train with weight decay, as SGD noise will prevent the within-class covariance from converging to zero. We experimentally validate this point in the following sections. Further, we want to study how this noise is affected by training hyperparameters such as batch size and learning rate.

2 Study Setup

In this section, we describe how we quantify the SGD noise and the procedure of the empirical study.

2.1 Characterising the Noise

It is computationally expensive to store all the weights W_t after each iteration in the training process. Thus, to study the convergence of the weights W_t and SGD noise, we keep track of three quantities that summarize relevant statistics about W_t:

Output Average: The first quantity track the average network's output on the training set:

$$\mu(f_{W_t}) = \frac{1}{m}\sum_{i=1}^{m} y_i f_{W_t}(x_i) \tag{4}$$

For the squared loss, the mean of this quantity should be 1 (notice the multiplication by y_i), and its variance should be an indicative measure of the amount of SGD noise. Moreover, we keep track of all $y_i f_{W_t}(x_i)$ for each sample in training dataset to study the SGD noise distribution.

Weights Average: Similarly, we measure the average of the weights W_t as follows:

$$\mu(W_t) = \frac{1}{p}\sum_{i=1}^{p} w_i \tag{5}$$

where w_i is each trainable parameter and p is the total number of parameters.

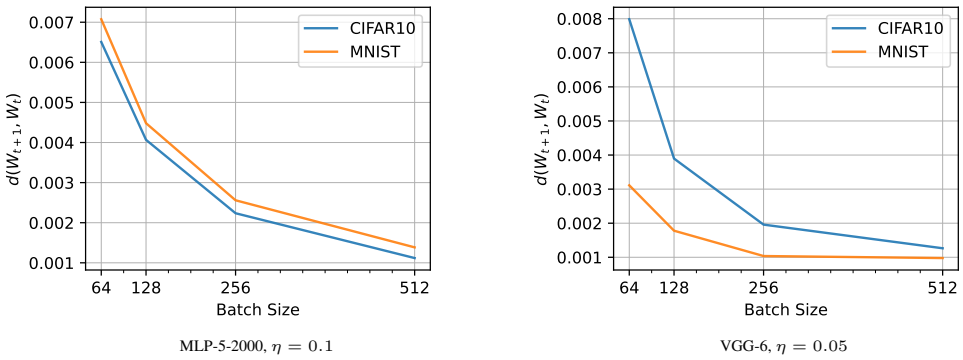
Weights Difference: Finally, we keep track of the Frobenius norm of the weights difference between each two consecutive epochs:

$$d(W_{t+1}, W_t) = \left\| W_{t+1} - W_t \right\|_F \tag{6}$$

We should expect this sequence to never converge exactly to zero due to SGD noise, and the sequence limit will correlate with the amount of noise.

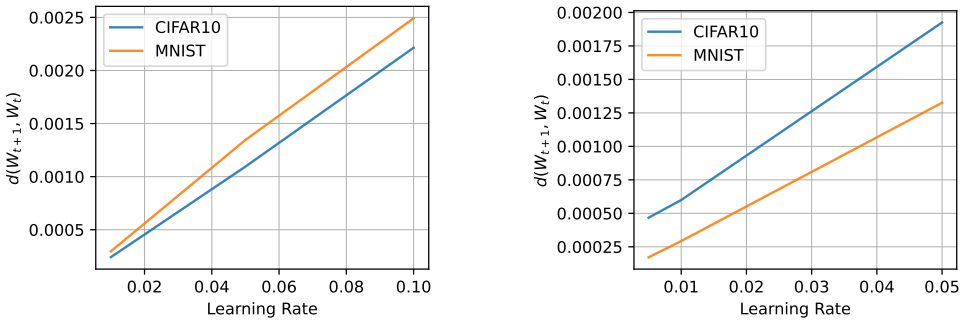
2.2 Network Architectures

We test two classes of neural networks. The first is a dense feedforward Multilayer Perception (MLP) with 5 layers of width 2000 neurons and ReLU activation. We call this model MLP-5-2000. The second class of neural networks, which we call VGG-6, is derived from the model proposed by Simonyan and Zisserman [7]. This model contains 6 layers in total with ReLU activation (4 are convolutional layers that are followed by 2 dense layers). The size of the kernels and the placement of the max pooling layers coincide with the first layers in VGG-11 detailed in [7].



MLP-5-2000, $\eta = 0.1$ VGG-6, $\eta = 0.05$

Figure 1: The relation between B and $d(W_{t+1}, W_t)$ for the MLP-5-2000 and VGG-6 models.



MLP-5-2000, $B = 256$ VGG-6, $B = 256$

Figure 2: The relation between η and $d(W_{t+1}, W_t)$ for the MLP-5-2000 and VGG-6 models.

2.3 Training Process and Datasets

Each model is trained for 4000 epochs with 5 different seeds. Training is done with a fixed learning rate without scheduling or annealing. We fixed the weight decay to $\lambda = 10^{-5}$. The models are trained on the MNIST and CIFAR10 datasets. To form the binary classification task, we select the first two classes from each dataset. The SGD noise is measured after separability is achieved (i.e., 100% classification accuracy on training) and when the norm of weights difference stops decaying.

3 Experiment Results

In this section, we present and analyze our simulation results. Overall, we used 8×RTX A4000 GPUs running for 48 hours to produce 140 simulations and gathered 28 GB of data. We include examples of the raw data in Figure 4 in the appendix.

3.1 Batch Size and SGD Noise

In this section, we examine the relationship between batch size and SGD noise. We measure the mean of $d(W_{t+1}, W_t)$ over the last 2000 epochs for a range of batch sizes $B \in \{64, 128, 256, 512\}$. Our results, shown in Figure 1, indicate that batch size has a significant influence on the amount of SGD noise. Furthermore, the relationship between batch size and noise appears to be exponential. We also observe that smaller batch sizes tend to lead to higher values of $d(W_{t+1}, W_t)$. These findings are consistent across both models and datasets. Given that the impossibility argument used in [1] relied on considering two batches that differ by one sample, such a strong relationship between the batch size and SGD is justified. Moreover, one hypothesis for the exponential relationship is the combinatorial nature of such pairs of batches. Finally, if one considers the case where the $B = m$ (thus, SGD becomes GD), one should expect the vanishing of SGD noise, which aligns with our observation.

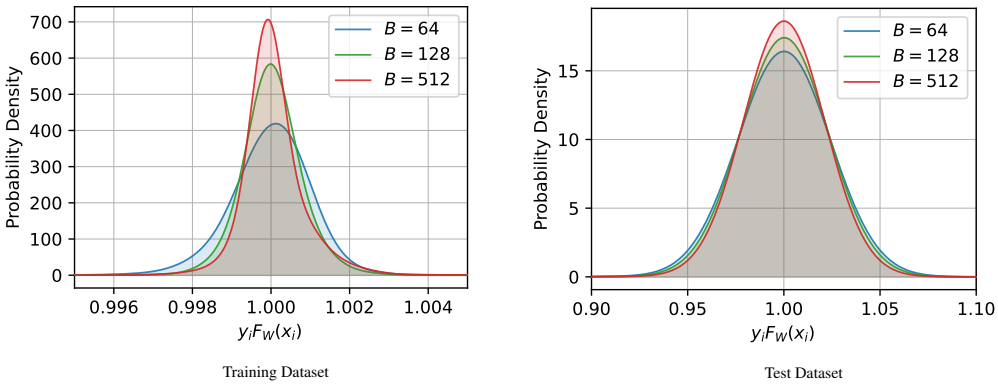


Figure 3: The distribution of $y_i f_W(x_i)$ for the trained MLP-5-2000 model on CIFAR10 dataset.

3.2 Learning Rate and SGD Noise

Next, we investigate the relationship between the learning rate and SGD noise. To do this, we tested a range of values for the learning rate $\eta \in [0.1, 0.001]$. We measured the mean of $d(W_{t+1}, W_t)$ over the last 1000 epochs to account for the slower convergence with smaller learning rates. These results are shown in Figure 2. As we can see from the figure, the learning rate also affects the amount of SGD noise, with larger learning rates leading to higher noise. However, the relationship between learning rate and noise is only linear, unlike the exponential relationship we observed between batch size and SGD noise. Based on these observations, we can conclude that the learning rate does not inherently influence SGD noise. Instead, the effect of the learning rate on SGD noise is determined by how it is used in the gradient update in Equation 3. This is supported by the fact that the slopes in Figure 2 are very agnostic to the training dataset.

3.3 SGD Noise Distribution

Finally, we study the distribution of SGD noise when measured at the network output. To do this, we focus on testing with different batch sizes, as this showed the strongest correlation with SGD noise. In Figure 3, we show the distribution of $y_i f_W(x_i)$ for the trained MLP-5-2000 model on both the training and test datasets. From the figure, we can see that the distribution of the network outputs appears to be Gaussian. Further, we can observe that these distributions are centered around 1, as anticipated for the square loss. Additionally, we can see that the distribution of the network outputs is wider for the test set than for the training set (variance of 10^{-3} in the test set versus 10^{-6} in the training set). Finally, these observations align with the results in Section 3.1, as we notice that smaller batch sizes lead to higher variance.

4 Conclusions

In this project, we empirically investigated how the batch size and learning rate affect the amount of SGD noise when neural networks are trained with weight decay. Our findings indicate that both hyperparameters have a clear relationship with the amount of SGD noise. In particular, we observed that smaller batch sizes lead to higher noise levels, while smaller learning rates lead to lower noise levels. Additionally, we found that the effect of batch size on SGD noise appears to be exponential, while the impact of the learning rate is linear. Finally, our investigation indicates that the distribution of the SGD noise, when measured at the model output, is Gaussian when the objective function we optimize for is the square loss.

Future research directions for this work include exploring whether the conclusions we have drawn about the relationship between batch size, learning rate, and SGD noise hold for different objective functions. Additionally, understanding the theoretical reason for the observed exponential relationship between batch size and SGD noise could be an exciting direction for further study. These investigations would provide valuable insights into the factors that influence SGD noise.

References

[1] Tomer Galanti and Tomaso Poggio. "SGD Noise and Implicit Low-Rank Bias in Deep Neural Networks". *Center for Brains, Minds and Machines (CBMM)*, 2022. URL https://arxiv.org/abs/2206.05794.

[2] Vardan Papyan, XY Han, and David L Donoho. "Prevalence of Neural Collapse During the Terminal Phase of Deep Learning Training". *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020. URL https://www.pnas.org/doi/10.1073/pnas.2015509117.

[3] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022. URL https://arxiv.org/abs/2201.02177.

[4] Katarzyna Janocha and Wojciech Marian Czarnecki. "On Loss Functions for Deep Neural Networks in Classification". In *Theoretical Foundations of Machine Learning (TFML)*, 2017. URL https://arxiv.org/abs/1702.05659.

[5] Samuel Smith, Erich Elsen, and Soham De. "On the Generalization Benefit of Noise in Stochastic Gradient Descent". In *International Conference on Machine Learning*, pages 9058–9067. PMLR, 2020. URL https://proceedings.mlr.press/v119/smith20a.html.

[6] Jingfeng Wu, Wenqing Hu, Haoyi Xiong, Jun Huan, Vladimir Braverman, and Zhanxing Zhu. On the Noisy Gradient Descent that Generalizes as SGD. In *International Conference on Machine Learning*, pages 10367–10376. PMLR, 2020. URL http://proceedings.mlr.press/v119/wu20c.html.

[7] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1409.1556.

A Appendix

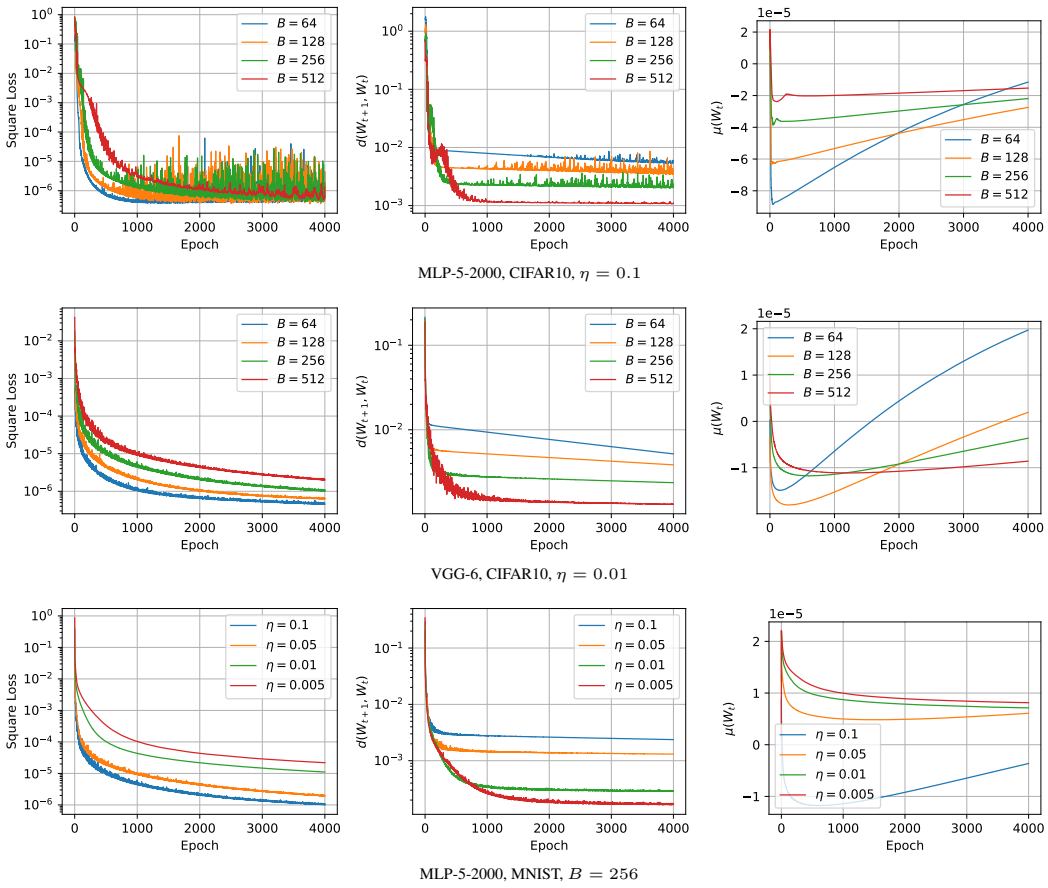


Figure 4: $\mathcal{L}_S^\lambda(f_W)$, $d(W_{t+1}, W_t)$, and $\mu(W_t)$ for different network architectures and datasets.