Analysis of Stochastic Gradient Descent of Deep Neural Networks Under Heavy-Tailed Gradient Noise

Dmitry Galkin [1]

Abstract

Recently, it has been demonstrated that the gradient noise in several deep learning settings has heavy-tailed behavior. This suggests that the gradient noise can be modeled by using heavy-tailed distributions. It is suggested that the stochastic gradient noise (SGN) in the stochastic gradient descent (SGD) algorithm for deep neural networks converges to a heavy-tailed α-stable random vector, where tail-index determines the heavy-tailedness of the distribution. Under the α-stable SGN assumption it is possible to explore connection between the convergence rate of SGD to a wide local minimum and the tail-index value. This connection provide a different perspective on the belief that SGD prefers wide minima. The main goal of this term paper is to conduct series of experiments in a more accurate and extensive way than it was illustrated in some related papers.

Github repo: project github link

1. Introduction

The nature of SGD efficacy remains the subject for a lot of research. One of the most popular hypotheses is that SGD can escape sharp local minima on the landscape defined by the loss function and prefers wider minima (that generalize better than narrow ones). This may be explained by the phenomenon of heavy-tailedness of the stochastic gradient noise distribution (see (Thanh Huy Nguyen, 2019)). In this project, we calculate stochastic gradient noise for several deep neural networks and perform an extensive empirical analysis of the tail-index of the SGN distribution.

2. Theory

In this part we follow (Umut Simsekli, 2019).

Stochastic Gradient Descent (SGD) is the most popular method for deep learning optimizing tasks. It often shows good generalization ability and the computation is fast as we take a batch of data points at a time. For neural network's non-convex optimization problem we can define SGD iterations as follows

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \eta \nabla \tilde{f}_k(\mathbf{w}^k). \tag{1}$$

The stochastic gradient is formulated by averaging gradients on the subset of data points

$$\nabla \tilde{f}_k(\mathbf{w}) \triangleq \nabla \tilde{f}_{\Omega_k}(\mathbf{w}) \triangleq \frac{1}{b} \sum_{i \in \Omega_k} \nabla f^{(i)}(\mathbf{w}). \tag{2}$$

Here, $\Omega_k \subset \{1, \ldots, n\}$ is a random subset that is drawn with or without replacement.

Stochastic gradient noise is defined as $U_k(\mathbf{w}) \triangleq \nabla \tilde{f}_k(\mathbf{w}) - \nabla f(\mathbf{w})$. The simplest approach to describing SGD as a stochastic process is to suppose that

$$U_k(\mathbf{w}) \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}). \tag{3}$$

Under this assumption, (1) can be written as follows:

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \eta \nabla f(\mathbf{w}^k) + \sqrt{\eta}\sqrt{\eta\sigma^2} Z_k, \tag{4}$$

such that Z_k is a standard normal random vector. A popular approach for investigating the behavior of SGD is based on considering SGD as a discretization of a continuous-time process:

$$\mathrm{d}\mathbf{w}_t = -\nabla f(\mathbf{w}_t)\mathrm{d}t + \sqrt{\eta\sigma^2}\mathrm{dB}_t, \tag{5}$$

where B_t denotes the standard Brownian motion.

Recent approach (Umut Simsekli, 2019) substitutes the gaussianity assumption with the following: gradient noise distribution obeys symmetric-α-stable ($S\alpha S$) law:

$$[U_k(\mathbf{w})]_i \sim \mathcal{S}\alpha\mathcal{S}_i(\sigma_i(\mathbf{w})), \quad \forall i = 1, \ldots, n \tag{6}$$

Following(**?**), with this assumption the SGD iteration takes form:

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \eta \nabla f(\mathbf{w}^k) + \eta^{1/\alpha}\left(\eta^{\frac{\alpha-1}{\alpha}}\sigma\right) S_k, \tag{7}$$

where $S_k \in \mathbb{R}^d$ is a random vector with independent components distributed according to the $\sim \mathcal{S}\alpha\mathcal{S}(1)$ distribution.

The $\sim \mathcal{S}\alpha\mathcal{S}$ distribution itself could be defined by characteristic function

$$X \sim \mathcal{S}\alpha\mathcal{S}(\sigma) \iff \mathbb{E}[\exp(i\omega X)] = \exp(-|\sigma\omega|^{\alpha}) \quad (8)$$

Despite the fact that $\sim \mathcal{S}\alpha\mathcal{S}$ probability density function has no closed-form formula, their tails decay can be represented by power law $1/|x|^{\alpha+1}$ where $\alpha \in (1, 2]$. The behavior of the distribution can be defined by α tail-index. When tail-index gets smaller, then tails gets heavier ($\alpha < 2$) and when $\alpha = 2$ - it is Gaussian distribution.

In some sense (7) can be interpreted as a discretization of stochastic differential equation (SDE) driven by α-stable Lévy process. The discontinuities of the Lévy motion enables it to 'jump' from the narrow basin into the wide minima (which has better generalization abilities). Seems like, if gradient noise exhibits similar heavy-tailed behavior to an α-stable distribution, this result can be considered as a approximation which can help us to understand the wide-minima behavior of SGD.

3. Experiments

We investigate the tail behavior of the stochastic gradient noise in a variety of scenarios. The main contribution of this term paper is that we don't rely on the assumption that SGN must be isotropic and obey the same distribution across dimensions. The main argue against the approach of (Umut Simsekli, 2019) was proposed by (Abhishek Panigrahi, 2019). The estimator assumes that the coordinates of SGN vector are i.i.d. This assumption is invalid in the typical over-parameterized. (Umut Simsekli, 2019) setting computed SGN across n model parameters and regarded SGN as n samples drawn from a single-variant distribution. That is why only one tail-index for whole neural nets was explored in (Umut Simsekli, 2019).

In this term paper we compute stochastic gradient noise for each iteration for each network's parameter over all minibatches. This experiment setup allows us to estimate SGN's α tail-index for each parameter and to investigate the homogeneity of distribution of α index across the whole net. For α index dynamics exploration in some cases it is more convenient to average them (to get an estimate of SGN's tail-idex for the whole model in some sense).

To investigate SGN distribution in addition to tail-index estimation we build several plots that provide us with different extensive studies for particular net's parameters.

3.1. Experiments with fully-connected networks

We consider a fully-connected network (FCN) on the MNIST dataset. For this model, we vary the the number of layers in the set 2, 3, 4, 5, the amount of neurons per layer in the set 16, 32, 64 and the size of minibatch in the set 40, 50, 100, 150, 200, 300. For FCN we run each configuration with the negative-log-likelihood (i.e. cross entropy).

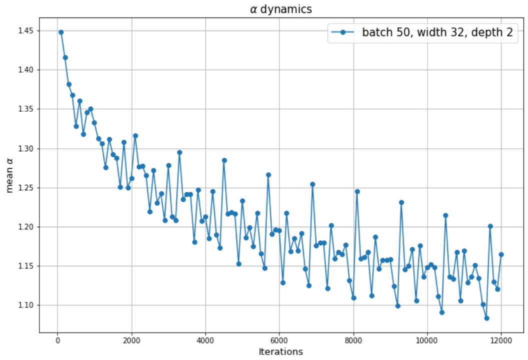

Figure 1. α dynamics for particular FCN with 12000 iterations

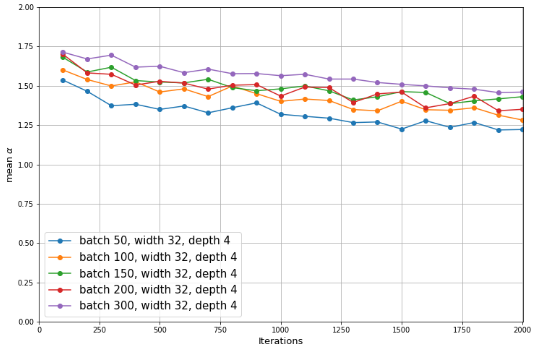

Figure 2. The iteration-wise behavior of α for FCN depending on batch size

Sizes of the networks that we stated above perform just good enough for our purposes. During experiments, we average the α measurements for 12000 iterations in order to observe dynamics.

Figure 1 shows that for first several thousands iterations α decreases more faster. And Figure 2 depicts that there is very clear and strong dependency between batch size and α tail-index rate of convergence to 1. Smaller batch size leads to more heavy-tailed distribution of SGN.

It is interesting to have a look on the SGN distribution itself. For the particular parameter in the beginning of training we observe quite Gaussian distribution (Figure 3). This is also true for the most of parameters for the first several dozens iterations. But as loss goes down and more iterations

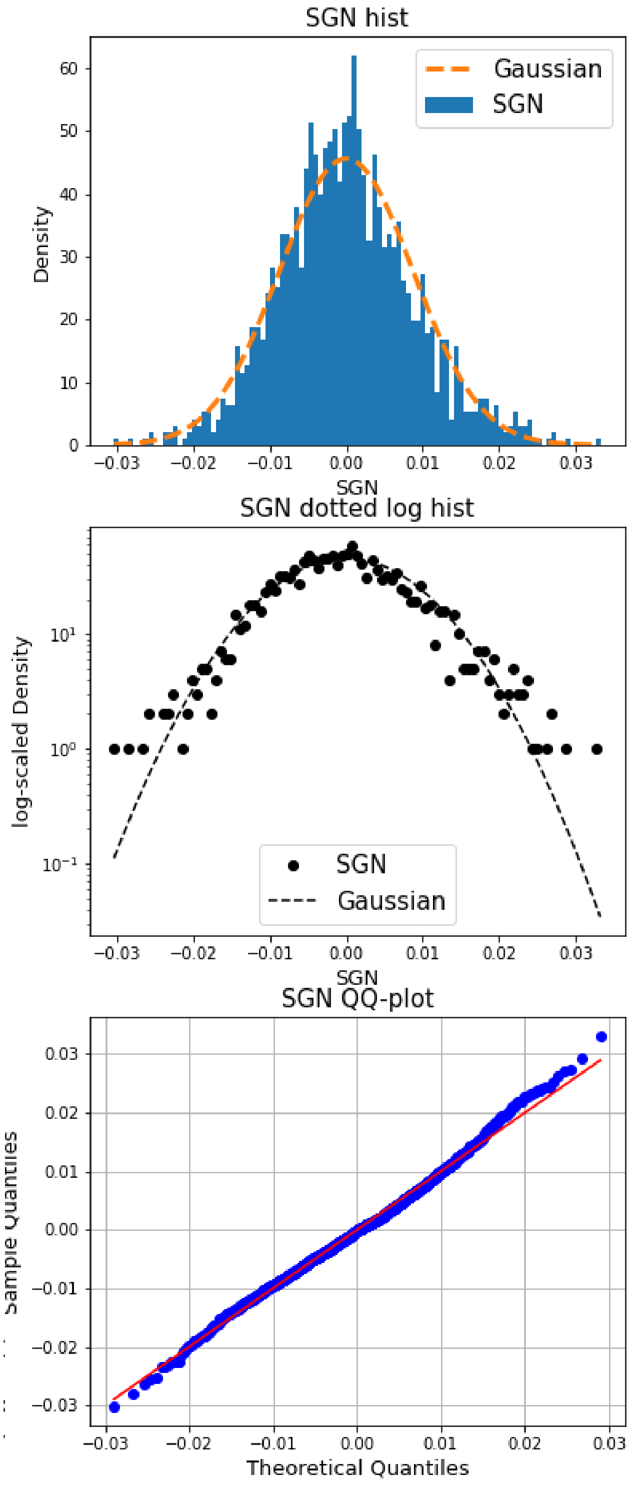

Figure 3. SGN distribution for FCN model with batch size 50 on 100 iteration. Upper plot: SGN histogram; Center plot: dotted SGN histogram in log scale; Lower plot: SGN QQ-plot with Gaussian distribution

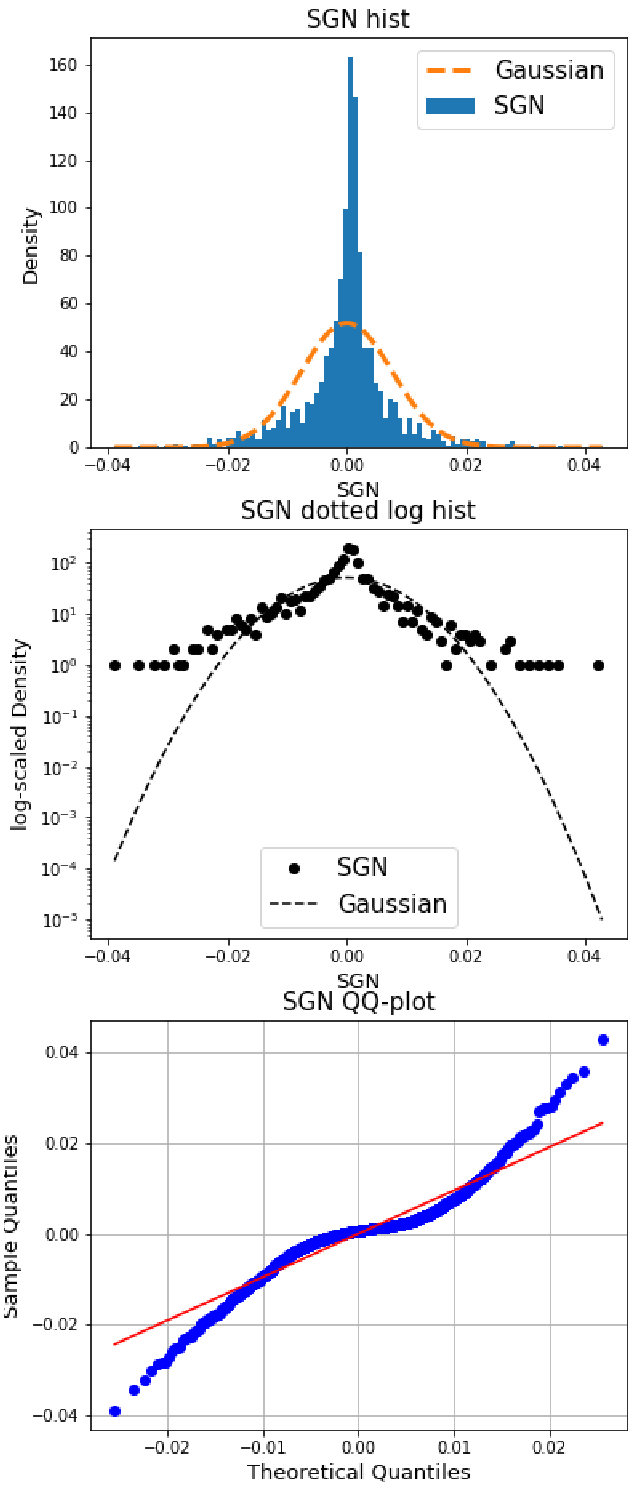

Figure 4. SGN distribution for FCN model with batch size 50 on 5000 iteration. Upper plot: SGN histogram; Center plot: dotted SGN histogram in log scale; Lower plot: SGN QQ-plot with Gaussian distribution

passed we monitor the increasing of kurtosis and heavier tails (Figure 4, Figure 5). Upper plots with histograms with high sharp distribution's picks are really far from fitted Gaussian's bell. Log-scaled histogram and Qintile-Quintile probability plots illustrate that SGN distribution's tails definitely far from theoretical Gaussian, especially for the last iterations of SGD optimization. Even these visual evidences could imply the hypothesis that gradient noise may be generated by some sort of generalised hyperbolic distribution, Gaussian mixture model or, as it is proposed above, α-stable distribution.

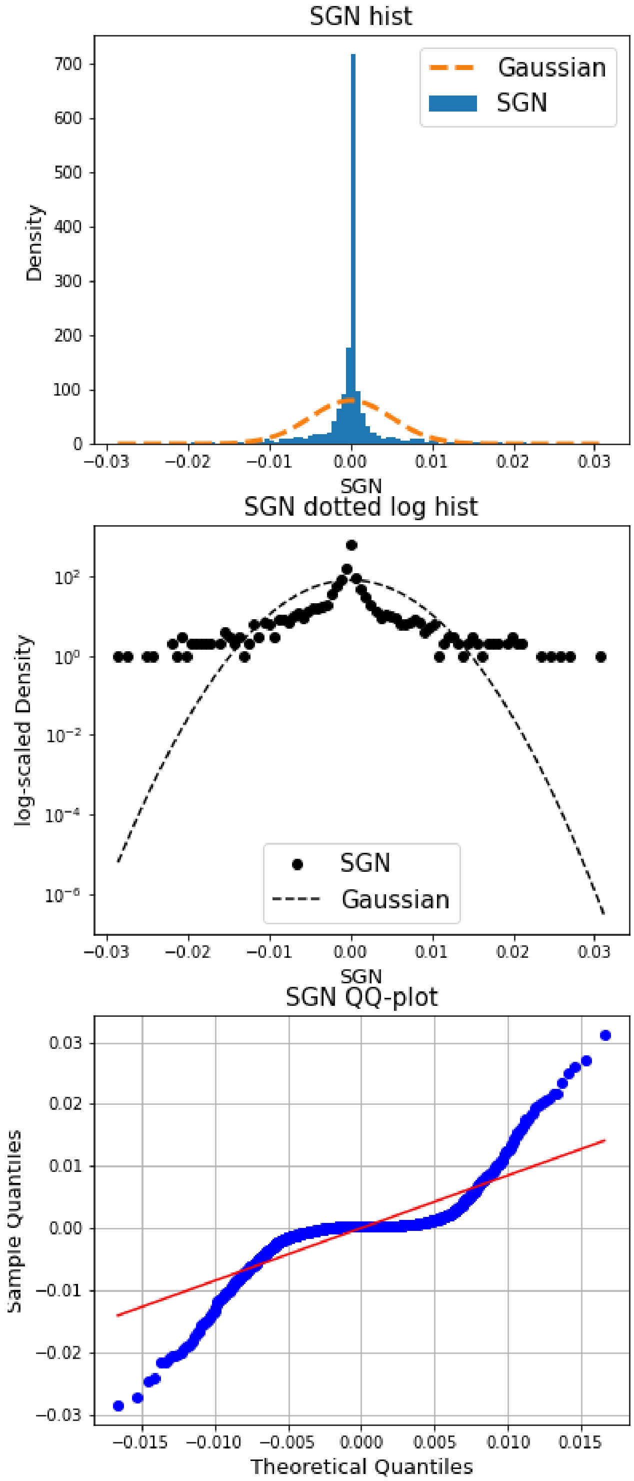

Figure 5. SGN distribution for FCN model with batch size 50 on 15000 iteration. Upper plot: SGN histogram; Center plot: dotted SGN histogram in log scale; Lower plot: SGN QQ-plot with Gaussian distribution

We estimate the α tail-index for varying the widths and depths of FCN. Heavy tails appear in all configurations of fully connected network since the estimated tail-index is far from 2. Figure 6 and Figure 7 shows that for the MNIST dataset tail-index decreases as depth and width parameters increase. This relation between value of α and net's parameters seems to be somehow connected with increasing of accuracy. Due to our suggestion about SGD driven by Levy process one can assume that increasing FCN size provide us with wider minima escaping from narrow ones.

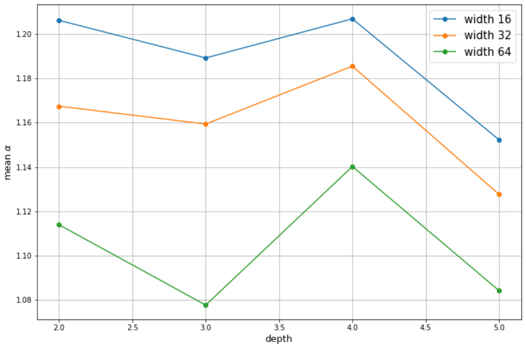

Figure 6. α estimation for FCN varying widths and depths for different widths

After evaluating the α estimation, we draw some plots with fitted α-stable distribution to demonstrate the level of proximity between SGN empirical distribution and α-stable.

For better and faster performance of fitting procedure it is more convenient to use R package ”alphastable”.

In Figure 8 and Figure 9 very clearly visible that α-stable distribution has much better approximation abilities since its tails are just as heavy as SGN's ones. Besides α-stable kurtosis has high and sharp peek which is also similar to stochastic gradient noise distribution and really far from Gaussian distribution.

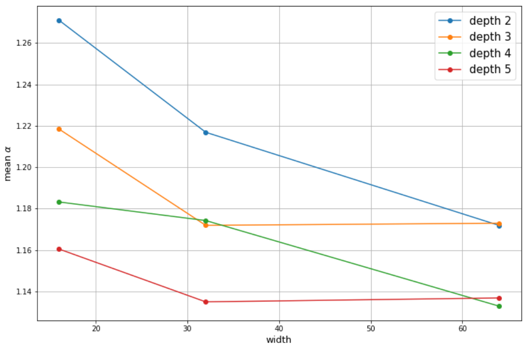

Figure 7. α estimation for FCN varying widths and depths for different depths

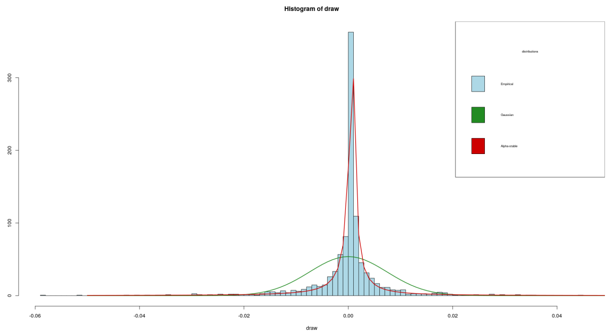

Figure 8. α Approximation of SGN by α-stable and Gaussin distributions.

3.2. Experiments with convolutional neural networks

The experiments were performed on Lenet convolutional network, as this network has a reputation of good performance on classification of MNIST digit images and is suitable for our constrained computational resources. The learning rate is 0.1, while the network consists of 2 convolutional layers with 6 and 16 filters respectively and 2 linear layers with 400 and 84 as hidden dimensions.The ReLU activation fuction was used after each layer, while maximum pooling of size 2 was implemented after each convolutional layer.

Main motivation of the experiments of alpha estimation performed for stochastic gradient noise calculated for convolutional neural network was to investigate the change of alpha during iterations in different scenarios.

We trained Lenet for 10 epochs with iteration over batches and saved stochastic gradient distribution for each 100th iteration. We trained model for different batch sizes: 50, 150, 200.

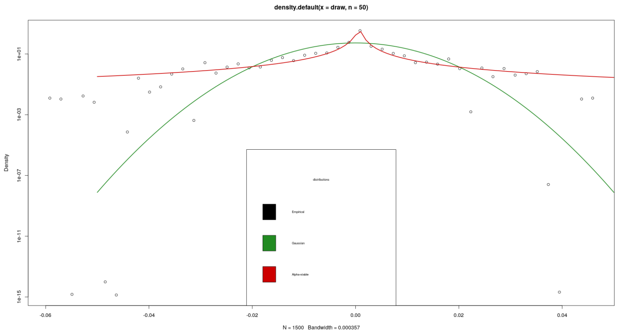

Figure 9. log-scaled approximation of SGN by α-stable and Gaussin distributions

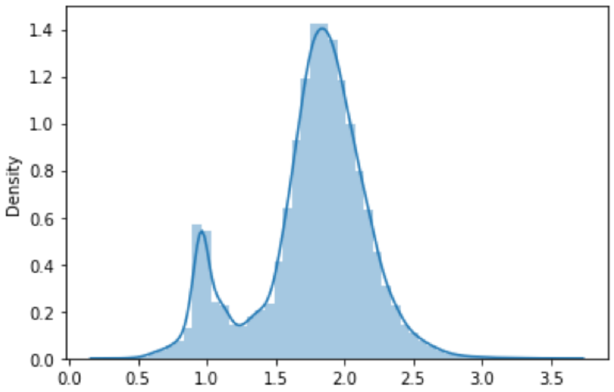

Figure 10. The distribution of alpha estimator for SGN for batch size 50 at iteration 100

The plots presented below depict the distribution of alpha estimator over all neurons of Lenet at 3 different iterations for batch size equal to 50. For initial iterations as illustrated by Figure 10, there exist alphas close to 1, but majority of alphas are located closer to 2, which implies that initial distribution of stochastic gradient noise is close normal for first iterations. Meanwhile, for later iterations alpha estimator values concentrate more around 1, which is showed by Figure 11 and emphasized by Figure 12, depicting the last iteration of mini-batch stochastic gradient descend.

The graphs below(Figures 13, 14, 15) illustrate the dynamics of mean alpha estimator for stochastic gradient noise over saved iterations. In general,the mean of alpha estimator falls after period of oscillation. Firstly, before figuring out the difference over the layers, let us evaluate the dependence on batch size of alpha estimator dynamics over training iterations for linear layers.

For the smallest batch size equal to 50, as depicted by Figure 13, the alpha estimator decreased with fluctuations from

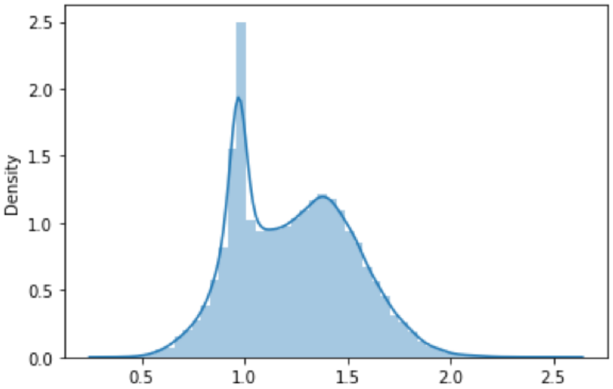

Figure 11. The distribution of alpha estimator for SGN for batch size 50 at iteration 5000

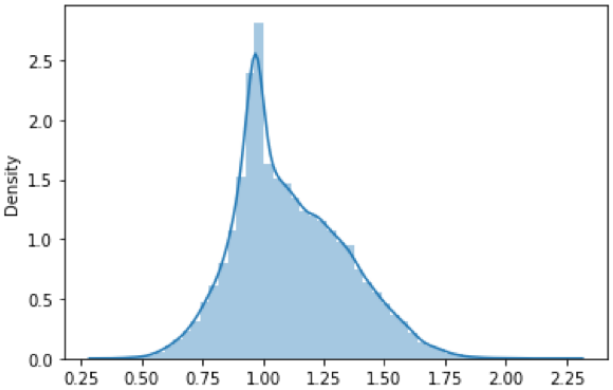

Figure 12. The distribution of alpha estimator for SGN for batch size 50 at last iteration (12000)

1.8 value of alpha up to 1.1, which implies that at the beginning of model training the distribution of stochastic gradient noise was characterised by less heavy tails, that resemble more like normal distribution, while as we trained the model, the stochastic gradient noise distribution was distancing from normal distribution and became more heavy-tailed. This negative trend of mean alpha estimator over iterations holds with fluctuations for all presented batch sizes. Nevertheless, for larger batch size the mean alpha estimator decreases more steadily and up to higher value of alpha estimator compare to smaller batch sizes. As showed by the Figure 14 for batch size equal to 150, mean alpha estimator during 10 epochs decreased up to around 1.5, which is lower that 2. Due to the fact that value of 2 corresponds to normal distribution, the distribution of SG noise for batch size 150 is distancing from normal, but to lower extend compare to SG noise for batch size 50. Therefore, we observe that with increase in batch size the distribution of alpha estimator

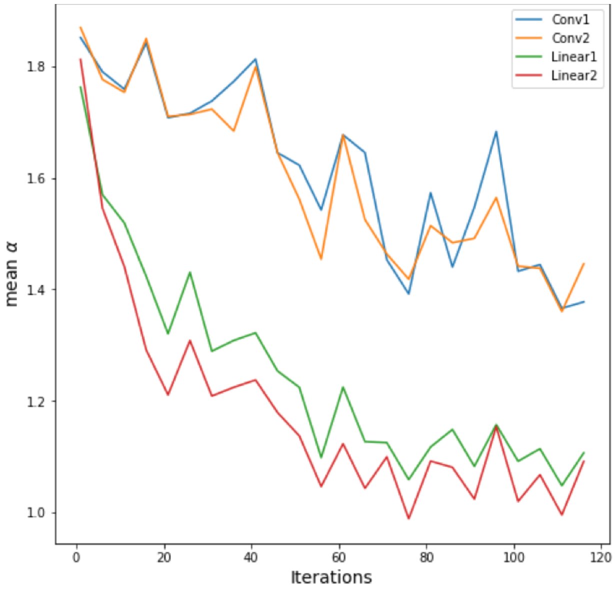

Figure 13. The dynamics mean of alpha estimator over saved iterations for SGN for batch size 50 for different layers of Lenet

is closer to normal. This can be attributed to the fact that the larger is batch size the lower is the variation between batches.

Intuitively, this phenomenon can be explained by the fact that at the beginning of training the distribution of difference between mini batch gradient and full gradient is distributed closer to normal as model weights are randomly initialised, while during the training procedure the as the model digests more data, the neural network finds the specific features of data in each batch and, thus, mini-batch stochastic gradient differs more from stochastic gradient on full data set, which results in heavier tails of distribution of stochastic gradient noise.

Then considering the difference in mean alpha estimator over layers of convolutional neural network Figures 13, 14, 15 show significant difference between alpha estimator for convolutional layers, that are located closer to network input, and following them linear layers. As depicted by Figures 13,14, 15 for convolutional layer the value of alpha is closer to 2, compare to linear layers, which implies that distribution of SG noise is very close to normal. Meanwhile, for linear layers closer to output the mean alpha estimator is closer to 1, reasoning heavy tails of stochastic gradient noise distribution. These Figures 13,14 indicate that the second linear layers mean alpha is higher from 1 for larger batch sizes.

The Figure 16 illustrates the distribution of alpha estimator for different layers. The distribution for linear layers is close

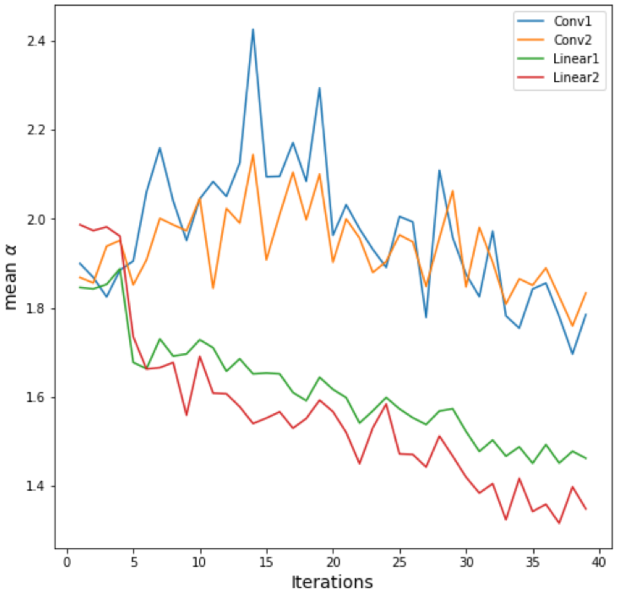

Figure 14. The dynamics of mean alpha estimator over saved iterations for SGN for batch size 150 for different layers of Lenet

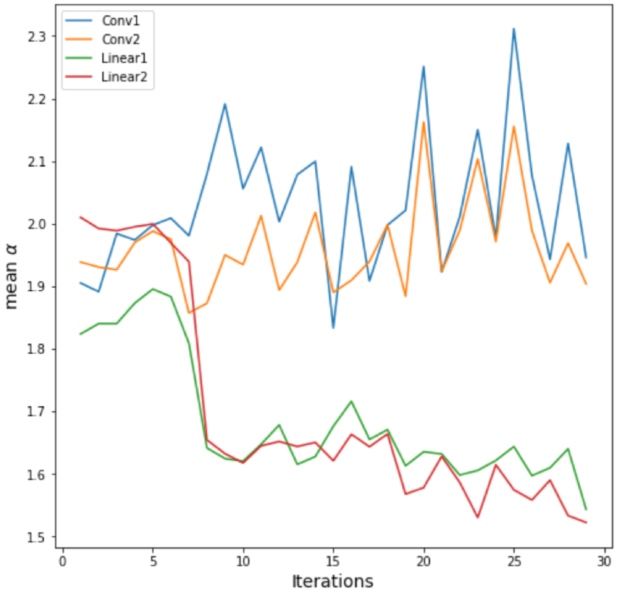

Figure 15. The dynamics mean of alpha estimator over saved iterations for SGN for batch size 200 for different layers of Lenet

to 1, convolutional layers is centered more between 1 and 2. This is in line with the results of paper (Umut Simsekli, 2019), that also showed that for layers closer to input layer the alpha is higher than alpha for layers closer to output layer. This corresponds to graphs depicting alpha estimator dependence on depth and width of neural network.

4. Conclusion

In this paper we investigated the nature of SGD in deep neural networks in a more reliable way than the proposed one in the original paper. We showed that the stochastic gradient noise distribution in FCN and CNN has heavy tails hence it is far from Gaussian distribution. Since that we consider SGN to be powered by α-stable distribution.

To estimate α we implemented estimator from the paper (Umut Simsekli, 2019), which showed stable results on tests. We fitted α-stable distribution to SGN and found it to be much closer to empirical distribution, especially in the sense of tail behavior. In the empirical analysis of the tail-index of the SG noise for FCN and CNN on the MNIST we observed that with increasing iterations, the tail index converged to 1. Moreover, varying batch size, depth and amount of neurons in layers appeared to be connected with tail index rate convergence.

Provided results suggest us to claim that, SGN distribution is close to α-stable. Assumption that stochastic gradient noise is α-stable leads us to interpret discritized Levy SDE as a proxy to understand the ”jump out” from the narrow minima of SGD.

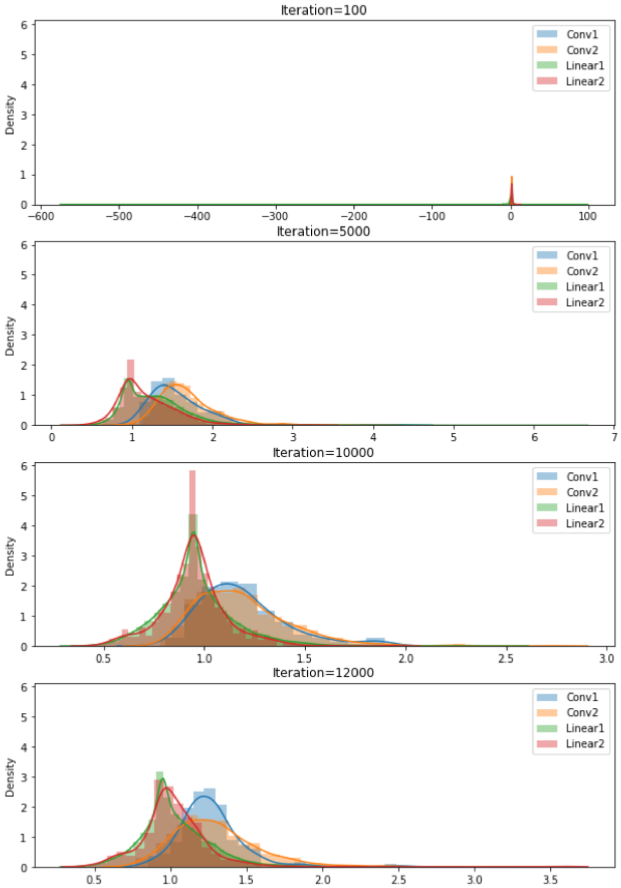

Figure 16. Distribution of alpha estimator for different layers and iterations for batch size 50

4.1. Citations and References

References

Abhishek Panigrahi, Raghav Somani, N. G. P. N. Non-gaussianity of stochastic gradient noise. *arXiv preprint arXiv:1910.09626*, 2019.

Thanh Huy Nguyen, Umut Simsekli, M. G. G. R. First exit time analysis of stochastic gradient descent underheavy-tailed gradient noise. *arXiv preprint arXiv:1906.09069*, 2019.

Umut Simsekli, Mert Gürbüzbalaban, T. H. N. G. R. L. S. On the heavy-tailed theory of stochastic gradient descent for deep neural networks. *arXiv preprint arXiv:1912.00018*, 2019.