Implicit Compressibility of Overparametrized Neural Networks Trained with Heavy-Tailed SGD

Yijun Wan [*1] Melih Barsbey [*2] Abdellatif Zaidi [1 3] Umut Şimşekli [4]

Abstract

Neural network compression has been an increasingly important subject, not only due to its practical relevance, but also due to its theoretical implications, as there is an explicit connection between compressibility and generalization error. Recent studies have shown that the choice of the hyperparameters of stochastic gradient descent (SGD) can have an effect on the compressibility of the learned parameter vector. These results, however, rely on unverifiable assumptions and the resulting theory does not provide a practical guideline due to its implicitness. In this study, we propose a simple modification for SGD, such that the outputs of the algorithm will be provably compressible without making any nontrivial assumptions. We consider a one-hidden-layer neural network trained with SGD, and show that if we inject additive heavy-tailed noise to the iterates at each iteration, for *any* compression rate, there exists a level of overparametrization such that the output of the algorithm will be compressible with high probability. To achieve this result, we make two main technical contributions: (i) we prove a "propagation of chaos" result for a class of heavy-tailed stochastic differential equations, and (ii) we derive error estimates for their Euler discretization. Our experiments suggest that the proposed approach not only achieves increased compressibility with various models and datasets, but also leads to robust test performance under pruning, even in more realistic architectures that lie beyond our theoretical setting.

1. Introduction

Obtaining compressible neural networks has become an increasingly important task in the last decade, and it has essential implications from both practical and theoretical perspectives. From a practical point of view, as the modern network architectures might contain an excessive number of parameters, compression has a crucial role in terms of deployment of such networks in resource-limited environments (O'Neill, 2020; Blalock et al., 2020). On the other hand, from a theoretical perspective, several studies have shown that compressible neural networks should achieve a better generalization performance due to their lower-dimensional structure (Arora et al., 2018; Suzuki et al., 2020a;b; Hsu et al., 2021; Barsbey et al., 2021; Sefidgaran et al., 2022).

Despite their evident benefits, it is still not yet clear how to obtain compressible networks with provable guarantees. In an empirical study, Frankle & Carbin (2018) introduced the "lottery ticket hypothesis", which indicated that a randomly initialized neural network will have a sub-network that can achieve a performance that is comparable to the original network; hence, the original network can be compressed to the smaller sub-network. This empirical study has formed a fertile ground for subsequent theoretical research, which showed that such a sub-network can indeed exist (see e.g., Malach et al., 2020; Burkholz et al., 2021; da Cunha et al., 2022). However, it is not clear how to develop an algorithm that can find it in a feasible amount of time.

Another line of research has developed methods to enforce compressibility of neural networks by using sparsity enforcing regularizers (see e.g., Papyan et al., 2018; Aytekin et al., 2019; Chen et al., 2020; Lederer, 2023; Kengne & Wade, 2023). While they have led to interesting algorithms, these typically require higher computational resources due to the increased complexity of the problem. On the other hand, due to the nonconvexity of the overall objective, it is also not trivial to provide theoretical guarantees for the compressibility of the resulting network weights.

Recently it has been shown that the training dynamics can have an influence on the compressibility of the algorithm output. In particular, motivated by the research that produced empirical and theoretical evidence that heavy tails

[*]Equal contribution [1]Paris Research Center, Huawei Technologies France [2]Boğaziçi University, İstanbul, Turkey [3]Université Gustave Eiffel, France [4]Inria, CNRS, Ecole Normale Supérieure, PSL Research University, Paris, France. Correspondence to: Yijun Wan <yijun.wan@hotmail.com>, Melih Barsbey <melih.barsbey@boun.edu.tr>.

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

might arise in stochastic optimization (see e.g., Martin & Mahoney, 2019; Simsekli et al., 2019; Şimşekli et al., 2019; Şimşekli et al., 2020; Zhou et al., 2020; Zhang et al., 2020; Camuto et al., 2021), Barsbey et al. (2021) and Shin (2021) showed that the network weights learned by stochastic gradient descent (SGD) will be compressible if we assume that they are heavy-tailed and that there exists a certain form of statistical independence within the network weights. These studies illustrated that, even *without* any modification to the optimization algorithm, the learned network weights can be compressible depending on the algorithm hyperparameters (such as the step size, i.e. learning rate, or the batch size). Even though the tail and independence conditions were recently relaxed by Lee et al. (2022), the resulting theory regards only the initialization and hence does not provide a fully practical guideline.

In this paper, we focus on single-hidden-layer neural networks with a fixed second layer (i.e., the setting used in previous work, De Bortoli et al., 2020) trained with vanilla SGD, and show that, when the iterates of SGD are simply perturbed by heavy-tailed noise with infinite variance (similar to the settings considered in Şimşekli, 2017; Nguyen et al., 2019; Şimşekli et al., 2020; Huang et al., 2021; Zhang & Zhang, 2023), the assumption made by Barsbey et al. (2021) in effect holds. More precisely, denoting the number of hidden units by n and the step size of SGD by η, we consider the *mean-field limit*, where n goes to infinity and η goes to zero. We show that in this limiting case, the columns of the weight matrix will be independent and identically distributed (i.i.d.) with a common *heavy-tailed* distribution. Then, we focus on the finite n and η regime and we prove that for *any* compression ratio (to be specified in the next section), there exists a number N, such that if $n \geq N$ and η is sufficiently small, the network weight matrix will be compressible with high probability. Figure 1 illustrates the overall approach and specifies our notion of compressibility.

To prove our compressibility result, we make two main technical contributions. We first consider the case where the step size $\eta \to 0$, for which the SGD recursion perturbed with heavy-tailed noise yields a *system* of heavy-tailed stochastic differential equations (SDE) with n particles. As our first technical contribution, we show that as $n \to \infty$ this particle system converges to a mean-field limit, which is a McKean-Vlasov-type SDE that is driven by a heavy-tailed process (Jourdain et al., 2007; Liang et al., 2021; Cavallazzi, 2023). We obtain a rate of convergence $n^{-1/2}$ in the presence of α-stable noises with $\alpha \in (1, 2)$, which is faster than the best-known rates, as recently proven by Cavallazzi (2023). This result indicates that a *propagation of chaos* phenomenon (Sznitman, 1991) emerges[1]: in the mean-field regime, the columns of the weight matrix will be i.i.d. and heavy-tailed due to the injected noise.

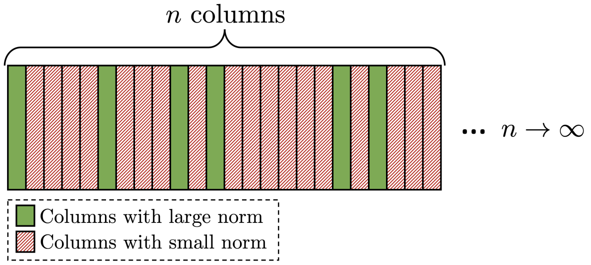


Figure 1: The illustration of the overall approach. We consider a one-hidden-layer neural network with n hidden units, which results in a weight matrix of n columns (first layer). We show that, when SGD is perturbed with heavy-tailed noise, as $n \to \infty$, each column will follow a multivariate heavy-tailed distribution in an i.i.d. fashion. This implies that a small number of columns will have significantly larger norms compared to the others; hence, the norm of the overall weight matrix will be determined by such columns (Gribonval et al., 2012). As a result, the majority can be removed (i.e., set to zero), which we refer to as compressibility.

Next, we focus on the Euler discretizations of the particle SDE to be able to obtain a practical, implementable algorithm. As our second main technical contribution, we derive *strong-error* estimates for the Euler discretization (Kloeden et al., 1992) and show that for sufficiently small η, the trajectories of the discretized process will be close to the one of the continuous-time SDE, in a precise sense. This result is similar to the ones derived for vanilla SDEs (e.g., Mikulevičius & Xu, 2018) and enables us to incorporate the error induced by using a finite step size η to the error of the overall procedure.

Equipped with these results, we finally prove a high-probability compression bound by invoking (Gribonval et al., 2012; Amini et al., 2011), which essentially shows that an i.i.d. sequence of heavy-tailed random variables will have a small proportion of elements that will dominate the whole sequence in terms of absolute values (to be stated formally in the next section). This establishes our main contribution. Here, we shall note that similar mean-field regimes have already been considered in machine learning (see e.g., Mei et al., 2018; Chizat & Bach, 2018; Rotskoff & Vanden-Eijnden, 2018; Jabir et al., 2019; Mei et al., 2019; De Bortoli et al., 2020; Sirignano & Spiliopoulos, 2022). However, these studies all focused on particle SDE systems that either converge to deterministic systems or that are driven by Brownian motion. While they have introduced interesting analysis tools, we cannot directly benefit from their analysis in this paper, since the heavy tails are crucial for obtaining compressibility, and the Brownian-driven SDEs cannot produce heavy-tailed solutions in general. Hence, as we

[1] Here, *chaos* refers to statistical independence: when the particles are initialized independently, they stay independent through the whole process although their common distribution may evolve.

consider heavy-tailed SDEs in this paper, we need to use different techniques to prove mean-field limits, compared to the prior art in machine learning.

To validate our theory, we conduct experiments with various neural networks and datasets. Our results show that, even with a minor modification to SGD (i.e., injecting heavy-tailed noise), the proposed approach can achieve compressibility with negligible computational overhead and with a slight compromise from the training and test error. Our findings further demonstrate that our methodology generalizes beyond our theoretical results, and produces models that are not only compressible, but robust in terms of test performance in fully connected neural networks with single or multiple hidden layers, and convolutional neural networks, implying that our approach is indeed a promising one in terms of its practical implications.

2. Preliminaries and Technical Background

Notation. For a vector $u \in \mathbb{R}^d$, denote by $\|u\|$ its Euclidean norm, and by $\|u\|_p$ its ℓ_p norm. For a continuous function $f \in C(\mathbb{R}^{d_1}, \mathbb{R}^{d_2})$, denote by $\|f\|_\infty := \sup_{x\in\mathbb{R}^{d_1}} \|f(x)\|$ its L^∞ norm. For a family of n (or ∞) vectors, the indexing $\cdot^{i,n}$ (or $\cdot^{i,\infty}$) denotes the i-th vector in the family. In addition, for random variables, $\overset{\text{(d)}}{=}$ means equality in distribution, and the space of probability measures on $\mathbb{R}^d$ is denoted by $\mathcal{P}(\mathbb{R}^d)$. For a matrix $A \in \mathbb{R}^{d_1\times d_2}$, its Frobenius norm is denoted by $\|A\|_F = \sqrt{\sum_{i=1}^{d_1}\sum_{j=1}^{d_2} |a_{i,j}|^2}$. Unless otherwise noted, $\mathbb{E}$ denotes the expectation over all the randomness taken into consideration.

Alpha-stable processes. A centered random variable $X \in \mathbb{R}^d$ is called *α-stable* with the stability parameter $\alpha \in (0, 2]$, if X_1, X_2, $\dots$ are independent copies of X, then $n^{-1/\alpha}\sum_{j=1}^n X_j \overset{\text{(d)}}{=} X$ for all $n \geq 1$ (Samoradnitsky, 2017). Stable distributions appear as limiting distributions in the generalized central limit theorem (CLT) (Gnedenko & Kolmogorov, 1954). In the one-dimensional case ($d = 1$), we call X a symmetric α-stable random variable if its characteristic function is of the following form: $\mathbb{E}[\exp(i\omega X)] = \exp(-|\lambda\omega|^\alpha)$ for $\omega \in \mathbb{R}$ and some $\lambda \in \mathbb{R}_+$.

For symmetric α-stable distributions, the case $\alpha = 2$ corresponds to the Gaussian distribution, while $\alpha = 1$ corresponds to the Cauchy distribution. An important property of α-stable distributions is that in the case $\alpha \in (1, 2)$, the p-th moment of an α-stable random variable is finite if and only if $p < \alpha$; hence, the distribution is heavy-tailed. In particular, $\mathbb{E}[|X|] < \infty$ and $\mathbb{E}[|X|^2] = \infty$, which can be used to model phenomena with heavy-tailed observations.

In the experiments, we consider the three most common types of α-stable random vectors that have been used in finance (Mandelbrot, 1963; Cont, 2001), statistical physics (Montroll & Bendler, 1984), and engineering literature (Nikias & Shao, 1995). We first describe these random vectors and then provide some intuition regarding their behavior.

- **Type-I.** Let $Z \in \mathbb{R}$ be a symmetric α-stable random variable. We then construct the random vector X such that all the coordinates of X is equated to Z. In other words $X = \mathbf{1}_d Z$, where $\mathbf{1}_d \in \mathbb{R}^d$ is a vector of ones. With this choice, X admits the following characteristic function: $\mathbb{E}\left[\exp(i\langle u, X\rangle\right] = \exp(-|\langle u, \mathbf{1}_d\rangle|^\alpha)$;
- **Type-II.** X has i.i.d. coordinates, such that each component of X is a symmetric α-stable random variable in $\mathbb{R}$. This choice yields the following characteristic function: $\mathbb{E}\left[\exp(i\langle u, X\rangle\right] = \exp(-\sum_{i=1}^d |u_i|^\alpha)$;
- **Type-III.** X is a rotationally invariant α-stable random vector with the characteristic function $\mathbb{E}\left[\exp(i\langle u, X\rangle\right] = \exp(-\|u\|^\alpha)$.

Notice that when added to a parameter vector (e.g. corresponding to a neuron), Type-I noise disturbs all parameters in the same direction and magnitude, e.g. acting like a random bias node scaled by the input. In contrast, Type-II noise constitutes an i.i.d. perturbation that affects each parameter separately, allowing some of the noise components to be very large while others are small, and/or in opposite directions. Lastly, due to the heavy-tailed distribution of its norm, Type-III noise vectors are likely to include elements that are simultaneously large or small in magnitude, yet these elements can vary among themselves in magnitude and direction. Also note that the Type-II and Type-III noises reduce to a Gaussian distribution when $\alpha = 2$, i.e., the characteristic function becomes $\exp(-\|\lambda u\|^2)$.

We define a more general class of random process, called the *α-stable Lévy process*, which extends the Brownian motion. Formally, α-stable processes are stochastic processes $(\mathrm{L}^\alpha_t)_{t\geq 0}$ with independent and stationary α-stable increments, and have the following definition:

- $\mathrm{L}^\alpha_0 = 0$ almost surely,
- For any $0 \leq t_0 < t_1 < \cdots < t_N$, the increments $\mathrm{L}^\alpha_{t_n} - \mathrm{L}^\alpha_{t_{n-1}}$ are independent,
- For any $0 \leq s < t$, the difference $\mathrm{L}^\alpha_t - \mathrm{L}^\alpha_s$ and $(t - s)^{1/\alpha}\mathrm{L}^\alpha_1$ have the same distribution,
- L^α_t is stochastically continuous, i.e. for any $\delta > 0$ and $s \geq 0$, $\mathbb{P}(\|\mathrm{L}^\alpha_t - \mathrm{L}^\alpha_s\| > \delta) \to 0$ as $t \to s$.

To fully characterize an α-stable process, we further need to specify the distribution of L^α_1. Along with the above properties, the choice for L^α_1 will fully determine the process.

For this purpose, we will again consider the previous three types of α-stable vectors: We will call the process L_t^α a Type-I process if L_1^α is a Type-I α-stable random vector. We define the Type-II and Type-III processes analogously. Note that, when $\alpha = 2$, Type-II and Type-III processes reduce to the Brownian motion. For notational clarity, occasionally, we will drop the index α and denote the process by L_t.

Compressibility of heavy-tailed processes. One interesting property of heavy-tailed distributions in the one-dimensional case is that they exhibit a certain compressibility property. Informally, if we consider a sequence of i.i.d. random variables coming from a heavy-tailed distribution, a small portion of these variables will likely have a very large magnitude due to the heaviness of the tails, and they will dominate all the other variables in magnitude (Nair et al., 2022). Therefore, if we only keep this small number of variables with large magnitude, we can “compress” (in a lossy way) the whole sequence of random variables by representing it with this small subset.

Concurrently, Amini et al. (2011); Gribonval et al. (2012) provided formal proofs for these explanations. Formally, Gribonval et al. (2012) characterized the family of probability distributions whose i.i.d. realizations are compressible. They introduced the notion of ℓ_p-compressibility - in terms of the error made after pruning a fixed portion of small (in magnitude) elements of an i.i.d. sequence, whose common distribution has diverging p-th order moments. More precisely, let $X_n = (x_1, \ldots, x_n)$ be a sequence of i.i.d. random variables such that $\mathbb{E}\left[|x_1|^\alpha\right] = \infty$ for some $\alpha \in \mathbb{R}_+$. Then, for all $p \geq \alpha$ and $0 < \kappa \leq 1$ denoting by $X_n^{(\kappa n)}$ the $\lfloor \kappa n \rfloor$ largest ordered statistics[2] of X_n, the following asymptotic on the relative compression error holds almost surely:

$$\lim_{n\to\infty} \frac{\|X_n^{(\kappa n)} - X_n\|_p}{\|X_n\|_p} = 0$$

Built upon this fact, Barsbey et al. (2021) proposed structural pruning of neural networks (the procedure described in Figure 1) by assuming that the network weights provided by SGD will be asymptotically independent. In this study, instead of making this assumption, we will directly prove that the network weights will be asymptotically independent in the two-layer (i.e. single-hidden-layer) neural network setting with additive heavy-tailed noise injections to SGD.

3. Problem Setting and the Main Result

We consider a single-hidden-layer overparametrized network of n units and use the setup provided in (De Bortoli et al., 2020). Our goal is to minimize the expected loss in a supervised learning regime, where for each data $z = (x, y)$ distributed according to $\pi(\mathrm{d}x, \mathrm{d}y)$,[3] the feature x is in $\mathcal{X} \subset \mathbb{R}^d$ and the label y is in $\mathcal{Y}$. We denote by $\theta^{i,n} \in \mathbb{R}^p$ the parameter for the i-th unit, and the parametrized model is denoted by $h_x : \mathbb{R}^p \to \mathbb{R}^l$. The mean-field network is the average over models for n units:

$$f_{\Theta^n}(x) = (1/n) \sum\nolimits_{i=1}^n h_x(\theta^{i,n}),$$

where $\Theta^n = (\theta^{i,n})_{i=1}^n \in \mathbb{R}^{p\times n}$ denotes the collection of parameters in the network and $x \in \mathcal{X}$ is the feature variable for the data point. In particular, the mean-field network corresponds to a two-layer neural network with the weights of the second layer fixed to be $1/n$ and Θ^n are the parameters of the first layer. While this model is less realistic than the models used in practice, we believe that it is desirable from a theoretical point of view, and this defect can be circumvented upon replacing $h_x(\theta^{i,n})$ by $h_x(c^{i,n}, \theta^{i,n}) = c^{i,n} h_x(\theta^{i,n})$, where $c^{i,n}$ and $\theta^{i,n}$ represent weights of the last layer and previous layers respectively. However, to obtain similar results in this setup as in our paper, stronger assumptions are inevitable and the proof should be more involved, which is left for future work, e.g., our framework has the potential to be extended to cases when the number of units in different layers goes to infinity one by one as in (Sirignano & Spiliopoulos, 2022).

Given a loss function $\ell : \mathbb{R}^l \times \mathcal{Y} \to \mathbb{R}^+$, the goal (for each n) is to minimize the expected loss taken over the distribution over the whole dataset π,

$$R(\Theta^n) = \mathbb{E}_{(x,y)\sim\pi}\left[\ell\left(f_{\Theta^n}(x), y\right)\right]. \tag{1}$$

One of the most popular approaches to minimize this loss is the stochastic gradient descent (SGD) algorithm. In this study, we consider a simple modification of SGD, where we inject a stable noise vector to the iterates at each iteration. For notational clarity, we will describe the algorithm and develop the theory over gradient descent, where we will assume that the algorithm has access to the true gradient ∇R at every iteration. However, since we are already injecting a heavy-tailed noise with *infinite variance*, our techniques can be adapted for handling the stochastic gradient noise (under additional assumptions, e.g., De Bortoli et al., 2020), which typically has a milder behavior compared to the α-stable noise[4].

Let us set the notation for the proposed algorithm. Let $\hat{\theta}_0^{i,n}$, $i = 1, \ldots, n$, be the initial values of the iterates, which are n random variables in $\mathbb{R}^d$ distributed independently

[2] In other words, $X_n^{(\kappa n)}$ is obtained by keeping only the largest (in magnitude) κn elements of X_n and setting all the other elements to 0.

[3] Note that for a finite dataset, π can be seen as a measure supported on finitely many points.

[4] In (Simsekli et al., 2019) the authors argued that the stochastic gradient noise in neural networks can be modeled by using stable distributions. Under such an assumption, the effect of the stochastic gradients can be directly incorporated into L_t^α.

according to a given initial probability distribution μ_0. Then, we consider the gradient descent updates with step size ηn, which is perturbed by i.i.d. α-stable noises $\sigma \cdot \eta^{1/\alpha} X_k^{i,n}$ for each unit $i = 1, \dots, n$, $\alpha \in (1, 2)$ and some $\sigma > 0$:

$$\hat{\theta}_{k+1}^{i,n} = \hat{\theta}_k^{i,n} - \eta n \left[\partial_{\theta^{i,n}} R(\Theta_k^n) \right] + \sigma \cdot \eta^{1/\alpha} X_k^{i,n} \tag{2}$$

where the scaling factor $\eta^{1/\alpha}$ in front of the stable noise enables the discrete dynamics of the system to homogenize to SDEs as $\eta \to 0$. Here σ is fixed to be a constant. In practice, we tune the step size η according to the number of neurons n, hence influencing the noise level. At this stage, we do not have to determine which type of stable noise (e.g., Type-I, II, or III) we shall consider as they will all satisfy the requirements of our theory. However, our empirical findings will illustrate that the choice will affect the overall performance.

We now state the assumptions that will imply our theoretical results. The following assumptions are rewritings with a certain degree of relaxation (in terms of the order of moments) of (De Bortoli et al., 2020, Assumption A1).

Assumption 3.1.

- Regularity of the model: for each $x \in \mathcal{X}$, the function $h_x : \mathbb{R}^p \to \mathbb{R}^l$ is two-times differentiable, and there exists a function $\Psi : \mathcal{X} \to \mathbb{R}_+$ such that for any $x \in \mathcal{X}$,

$$\|h_x(\cdot)\|_\infty + \|\nabla h_x(\cdot)\|_\infty + \|\nabla^2 h_x(\cdot)\|_\infty \le \Psi(x).$$

- Regularity of the loss function: there exists a function $\Phi : \mathcal{Y} \to \mathbb{R}_+$ such that

$$\|\partial_1 \ell(\cdot, y)\|_\infty + \|\partial_1^2 \ell(\cdot, y)\|_\infty \le \Phi(y)$$

- Moment bounds on $\Phi(\cdot)$ and $\Psi(\cdot)$: there exists a positive constant B such that

$$\mathbb{E}_{(x,y)\sim\pi}[\Psi^2(x)(1 + \Phi^2(y))] \le B^2.$$

Let us remark that Assumption 3.1 includes the smoothness and boundedness assumptions that have been made in the mean field literature (Mei et al., 2018; 2019) and are satisfied by several smooth activation functions, including the sigmoid and hyper-tangent functions.

We now proceed to our main result. Let $\hat{\Theta}_k^n \in \mathbb{R}^{p\times n}$ be the matrix with columns being the parameters $\hat{\theta}_k^{i,n}$, $i = 1, \dots, n$ obtained by the recursion (2) after k iterations. We will now compress $\hat{\Theta}_k^n$ by pruning its columns with small norms. More precisely, fix a compression ratio $\kappa \in (0, 1)$, compute the norms of the columns of $\hat{\Theta}_k^n$, i.e., $\|\hat{\theta}_k^{i,n}\|$. Then, keep the $\lfloor \kappa n \rfloor$ columns, which have the largest norms, and set all the other columns to zero in entirety. Finally, denote by $\hat{\Theta}_k^{(\kappa n)} \in \mathbb{R}^{p\times n}$, the pruned version of $\hat{\Theta}_k^n$.

Theorem 3.1. *Suppose that Assumption 3.1 holds. For any $\alpha \in (1, 2)$, if we fix $k > 0$, $\kappa \in (0, 1)$ and $\epsilon > 0$ sufficiently small, with probability $1 - \epsilon$, there exists $N \in \mathbb{N}_+$ such that for all $n \ge N$ and η such that $\eta \le n^{-\alpha/2-1}$, the following upper bound on the relative compression error holds:*

$$\frac{\left\| \hat{\Theta}_k^{(\kappa n)} - \hat{\Theta}_k^n \right\|_F}{\left\| \hat{\Theta}_k^n \right\|_F} \le \epsilon. \tag{3}$$

This bound shows that, thanks to the heavy-tailed noise injections, the weight matrices will be compressible at *any* compression rate, as long as the network is sufficiently overparametrized and the step size is sufficiently small. We note that (3) can be extended to an error bound on the outputs of the neural network by using similar techniques as in (Barsbey et al., 2021, Lemma S1). Furthermore, Theorem 3.1 also enables us to directly obtain a generalization bound by invoking (Barsbey et al., 2021, Theorem 4)[5].

4. Proof Strategy and Intermediate Results

In this section, we gather the main technical contributions with the purpose of demonstrating Theorem 3.1. We begin by rewriting (2) in the following form:

$$\hat{\theta}_{k+1}^{i,n} - \hat{\theta}_k^{i,n} = \eta b(\hat{\theta}_k^{i,n}, \hat{\mu}_k^n) + \sigma \cdot \eta^{1/\alpha} X_k^{i,n} \tag{4}$$

where $\hat{\mu}_k^n = \frac{1}{n}\sum_{i=1}^n \delta_{\hat{\theta}_k^{i,n}}$ is the empirical distribution of parameters at iteration k and δ is the Dirac measure, and the drift is given by $b(\theta_k^{i,n}, \mu_k^n) = -\mathbb{E}[\partial_1 \ell(\mu_k^n(h_x(\cdot)), y) \nabla h_x(\theta_k^{i,n})]$, where ∂_1 denotes the partial derivative with respect to the first parameter and

$$\mu_k^n(h_x(\cdot)) := \frac{1}{n}\sum_{i=1}^n h_x(\theta_k^{i,n}) = f_{\Theta_k^n}(x).$$

It is easy to check that $b(\theta_k^{i,n}, \mu_k^n) = -n\partial_{\theta^{i,n}} R(\Theta^n)$. By looking at the dynamics from the perspective of empirical distributions, we can treat the evolution of the parameters as a system of evolving probability distributions μ_k^n: the empirical distribution of the parameters during the training process will converge to a limit as η goes to 0 and n goes to infinity.

We start by linking the recursion (2) to its limiting case where $\eta \to 0$. The limiting dynamics can be described by the following system of SDEs:

$$\mathrm{d}\theta_t^{i,n} = b(\theta_t^{i,n}, \mu_t^n)\mathrm{d}t + \sigma \mathrm{d}\mathrm{L}_t^{i,n} \tag{5}$$

[5] This bound would imply that with high probability, the generalization gap for (slightly modified) $0-1$ loss function is roughly of order $\mathcal{O}(\sqrt{\kappa n p \frac{\log N}{N}})$, where N is the number of data points and p is the data dimension.

where $\mu_t^n = \frac{1}{n}\sum_{i=1}^n \delta_{\theta_t^{i,n}}$ and $(\mathrm{L}_t^{i,n})_{t\geq 0}$ are independent α-stable processes such that $\mathrm{L}_1^{i,n} \overset{(\mathrm{d})}{=} X_1^{i,n}$. We can now see the recursion (2) as an Euler discretization of (5) and then we have the following strong uniform error bound for the discretization.

Lemma 4.1. *Let $(\theta_t^{i,n})_{t\geq 0}$ be the solutions to SDE (5) and $(\hat{\theta}_k^{i,n})_{k\in\mathbb{N}_+}$ be given by SGD (2) with the same initial condition $\theta_0^{i,n}$ and α-stable Lévy noise $\mathrm{L}_{\cdot}^{i,n}$, i=1,…,n. Under Assumption 3.1, for any $T > 0$, if $\eta k \leq T$, there exists a constant C depending on B, T, α such that*

$$\mathbb{E}\left[\sup_{i\leq n} \|\theta_{\eta k}^{i,n} - \hat{\theta}_k^{i,n}\|\right] \leq C(\eta n)^{1/\alpha}.$$

In comparison to the standard error estimates in the Euler-Maruyama scheme concerning only the step size η, the additional n-dependence is because here we consider the supremum of the approximation error over all $i \leq n$, which involves the expectation of the supremum of the modulus of n independent α-stable random variables.

Next, we start from the system (5) and consider the case where $n \to \infty$. In the limit, we obtain the following McKean-Vlasov-type stochastic differential equation:

$$\mathrm{d}\theta_t^\infty = b(\theta_t^\infty, [\theta_t^\infty])\mathrm{d}t + \mathrm{d}\mathrm{L}_t \tag{6}$$

where $(\mathrm{L}_t)_{t\geq 0}$ is an α-stable process and $[\theta_t^\infty]$ denotes the distribution of θ_t^∞. The existence and uniqueness of a strong solution to (6) are given by Cavallazzi (2023). Moreover, for any positive T, $\mathbb{E}\left[\sup_{t\leq T}\|\theta_t^\infty\|^\alpha\right] < +\infty$. This SDE with measure-dependent coefficients turns out to be a useful mechanism for analyzing the behavior of neural networks and provides insights into the effects of noise on the learning dynamics.

In this step, we will link the system (5) to its limit (6), which is a strong uniform propagation of chaos result for the weights. The next result shows that, when n is sufficiently large, the trajectories of weights asymptotically behave i.i.d. according to (6).

Lemma 4.2. *Following the existence and uniqueness of strong solutions to (5) and (6), let $(\theta_t^{i,\infty})_{t\geq 0}$ be solutions to the McKean-Vlasov equation (6) and $(\theta_t^{i,n})_{t\geq 0}$ be solutions to (5) associated with the same realization of α-stable processes $(\mathrm{L}_t^i)_{t\geq 0}$ for each i. Suppose that $(\mathrm{L}_t^i)_{t\geq 0}$ are independent. Then there exists C depending on T, B such that*

$$\mathbb{E}\left[\sup_{t\leq T}\sup_{i\leq n}|\theta_t^{i,n} - \theta_t^{i,\infty}|\right] \leq \frac{C}{\sqrt{n}}$$

Our result differs from the existing literature by taking the supremum over the indices i before taking the expectation, which is obviously stronger than taking the supremum over i outside the expectation. It is also worth mentioning that the $O(n^{-1/2})$ decreasing rate here is better, if $\alpha < 2$, than the state of the art (Cavallazzi, 2023) with classical Lipschitz assumptions on the coefficients of SDEs. The reason is that here, thanks to Assumption 3.1, we can benefit from the one-hidden-layer neural network structure.

Finally, we are interested in the distributional properties of solutions to the McKean-Vlasov equation (6). The following result establishes that their marginal distributions have diverging second-order moments, hence, they are heavy-tailed.

Lemma 4.3. *Let $(\mathrm{L}_t)_{t\geq 0}$ be an α-stable process. For any time t, let θ_t be the solution to (6) with initialization θ_0 which is independent of $(\mathrm{L}_t)_{t\geq 0}$ such that $\mathbb{E}\left[\|\theta_0\|\right] < \infty$, then the following holds for $t > 0$,*

$$\mathbb{E}\left[\|\theta_t^\infty\|^2\right] = +\infty.$$

We remark that the result is weak in the sense that details on the tails of θ_t with respect to α and t are implicit. However, it renders sufficient for our compressibility result in Theorem 3.1. Now, having proved all the necessary ingredients, Theorem 3.1 is obtained by accumulating the error bounds proven in Lemmas 4.1 and 4.2, and applying (Gribonval et al., 2012, Proposition 1) along with Lemma 4.3.

Note that, due to the multiplicative noise in plain SGD (without additional injected noise), the iterates might already possess a heavy-tailed behavior (Gurbuzbalaban et al., 2021; Hodgkinson & Mahoney, 2021; Pavasovic et al., 2023). However, this behavior does not directly result in compressibility as some notion of independence between the columns of $\hat{\Theta}_k^n$ is necessary (Amini et al., 2011; Gribonval et al., 2012; Silva & Derpich, 2015).

Additional theoretical results. In the Appendix, we investigate two other properties of the considered scheme. In Appendix A, we prove that when the injected noise is not heavy-tailed (i.e., α is set to 2 and the noise becomes Gaussian) and when the step-size goes to zero, the obtained network weights *cannot* be compressible in terms of the notion we defined in Theorem 3.1. This shows that heavy tails are instrumental in order to guarantee compressibility in our specific compression definition.

In Appendix B, we investigate the effects of the heavy tails on the training loss. In particular, we upper-bound the expected gradient norm, i.e., $\mathbb{E}\|\nabla R(\hat{\Theta}_k^n)\|^2$ and show that the gradient norm will be bounded by two terms: (i) one term that linearly goes to zero as K increases, (ii) another term, that scales up with the noise scale σ. This result highlights the fact that injecting heavy-tailed noise introduces a trade-off: while the noise is beneficial in terms of compressibility, it might hurt the optimization performance. In

α	Test Acc.	Pruning Ratio	Test Acc. a.p.
none	94.15 ± 0.51	11.42 ± 0.04	94.12 ± 0.85
1.75	94.15 ± 0.51	49.38 ± 22.72	90.50 ± 10.64
1.8	94.15 ± 0.51	37.22 ± 14.51	93.26 ± 1.85
1.9	94.15 ± 0.51	25.90 ± 11.02	93.56 ± 1.41

Table 1: ECG5000, Type-I noise, $n = 2$K.

α	Test Acc.	Pruning Ratio	Test Acc. a.p.
none	94.16 ± 0.28	11.44 ± 0.01	93.75 ± 1.77
1.75	94.16 ± 0.28	54.72 ± 18.46	90.12 ± 5.42
1.8	94.16 ± 0.28	44.90 ± 11.04	91.45 ± 6.36
1.9	94.16 ± 0.28	30.86 ± 8.11	93.21 ± 1.02

Table 2: ECG5000, Type-I noise, $n = 10$K.

the next section, we investigate this trade-off in a number of experiments.

5. Empirical Results

In this section, we validate our theory with empirical results. Our goal is to investigate the effects of the heavy-tailed noise injection in SGD in terms of compressibility and the train/test performance. For our experiments we use the ECG5000 (Baim et al., 2000), MNIST (LeCun et al., 2010), CIFAR-10, and CIFAR-100 (Krizhevsky, 2009) datasets. By slightly stretching the scope of our theoretical framework, we also train the weights of the second layer instead of fixing them to $1/n$. We start our experiments with a single-hidden-layer neural network with ReLU activations and the cross-entropy loss, applied on classification tasks. We then examine how well our results generalize to more complex architectures by conducting experiments using fully connected neural networks (FCN) with more hidden layers, as well as using convolutional neural networks (CNN).

For SGD, the step size is chosen to be small enough to approximate the continuous dynamics given by the McKean-Vlasov equation in order to stay close to the theory, but also not too small so that SGD converges in a reasonable amount of time. We fix the batch size to be as large as possible within memory constraints. For all experiments, the training was continued until reaching 95% accuracy on the training set. As for the noise level σ, we try a range of values for each dataset and n, and we choose the largest σ such that the perturbed SGD converges, without a dramatic performance cost to the pruned model. Intuitively, we can expect that smaller α with heavier tails will lead to lower relative compression error. However, it does not guarantee better test performance: we will investigate the trade-offs between compression error and test performance more in detail below. All the experimentation details are given in Appendix E, in addition to the extended versions of the results presented here, and our source code includes the relevant implementation details[6].

5.1. Experiments with ECG5000

We first consider the ECG5000 dataset and investigate the effects of α and n on compressibility and performance. We repeat the experiments 10 times and report means and standard deviations in Tables 1-4. Here, for different cases, we monitor the training and test accuracies before and after pruning (a.p.), as well as the pruning ratio: the percentage of the weight matrix that can be pruned while keeping the 90% of the squared norm of the original matrix[7]. For brevity we present training accuracies in Appendix E.

Using Type-I noise, the results of our first experiment (Table 1) show that even for a moderate number of neurons $n = 2$K, the heavy-tailed noise results in a significant improvement in the compression capability of the neural network. For $\alpha = 1.9$, we can see that the pruning ratio increases to 25.90%, whereas vanilla SGD can only be compressible with a rate 11.42%, with only a slight decrease in pruned test performance for the noise-added model. We also observe that decreasing α (i.e., increasing the heaviness of the tails) results in a better compression rate; yet, there is a trade-off between this rate and the test performance. In Table 2, we repeat the same experiment for $n = 10$K. We observe that the previous conclusions become even clearer in this case, as our theory applies to large n. For the case where $\alpha = 1.75$, we obtain a pruning ratio of 54.72% with test accuracy 90.12%, whereas for vanilla SGD the ratio is only 11.44% with a test accuracy of 93.75%.

We also investigate the impact of noise type, where we set $n = 10$K and use the same setting as in Table 2. Tables 3-4 illustrate the results. We observe that the choice of the noise type moderately impacts compressibility and accuracy. Type-III noise seems to demonstrate a similar pattern to Type-I, while achieving a worse compression rate overall. On the other hand, although Type-II noise improves on Type-I in its performance under $\alpha = 1.75$, it loses on performance and/or compression in the other two α values. Accordingly, we conclude Type-I noise to obtain a better trade-off overall, and proceed to the remaining experiments with it.

5.2. Experiments with MNIST

In our next experiment, we consider the MNIST dataset, set $n = 5$K and use Type-I noise. Table 5 illustrates the results as the average and the standard deviation of 5 runs. Similar to the previous results, we observe that the injected

[6] `https://github.com/mbarsbey/imp_comp`

[7] The pruning ratio has the same role of $1 - \kappa$, where we fix the compression error to 0.1 and find the smallest κ that satisfies this error threshold.

α	Test Acc.	Pruning Ratio	Test Acc. a.p.
1.75	94.17 ± 0.51	54.40 ± 18.92	93.41 ± 2.24
1.8	94.72 ± 1.22	39.85 ± 13.66	91.68 ± 3.13
1.9	94.62 ± 0.36	22.05 ± 9.87	93.31 ± 2.05

Table 3: ECG5000, Type-II noise, $n = 10$K.

α	Test Acc.	Pruning Ratio	Test Acc. a.p.
1.75	94.35 ± 0.64	51.25 ± 16.55	91.93 ± 3.13
1.8	94.10 ± 0.49	38.09 ± 16.67	92.74 ± 2.23
1.9	94.51 ± 0.47	21.92 ± 9.56	93.18 ± 2.83

Table 4: ECG5000, Type-III noise, $n = 10$K.

noise has a visible benefit on compressibility. When $\alpha = 1.9$, our approach doubles the compressibility of the vanilla SGD (from 10.58% to 23.82%), while pruned test accuracy decreases only by $\sim 1\%$. On the other hand, when we decrease α, the pruning ratio goes up to 40.63%, while only compromising $\sim 3\%$ of pruned test accuracy.

5.3. Experiments with CIFAR-10 and CIFAR-100

We now test our approach with datasets and model architectures that are relatively more realistic in a machine learning setting (see Appendix E for full details). First, we conduct experiments with the CIFAR-10 dataset using the architecture in the MNIST experiments above, where we set $n = 5$K and use Type-I noise. We present our results as the average and standard deviation of 5 runs in Table 6. We observe that the results are similarly positive for CIFAR-10, where dramatic improvements in compressibility are obtained for a small cost to pruned test performance.

Importantly, in most practical discussions of compressibility (Blalock et al., 2020), it is also desired that the compressed network is *robust* to compression in terms of performance: That is, the pruned network is expected to maintain its test performance in the face of pruning. To compare the networks trained under our approach to vanilla SGD in terms of robustness, we progressively prune more of the columns of each model, and examine the models' test accuracy under increasing pruning ratios (e.g. $0.1, 0.2, \dots$). The results are presented in Figure 2's column (a). Here we plot models' absolute and relative accuracy as a function of pruning ratio, where relative test accuracy refers to the test accuracy of a pruned model in proportion to its unpruned test accuracy. Our findings unequivocally demonstrate the advantage of our approach: Networks trained with heavy-tailed noise (of all three αs) are not only more compressible, but are also more robust to pruning in terms of performance.

Robustness with More Complex Architectures Inspired by the robustness results presented, we then test whether our results generalize to more complex, and arguably more realistic architectures. Though this means venturing beyond our theoretical setting, we find it crucial from a practical point of view to examine whether our methodology obtains robustness in such contexts. For this purpose, we train a CNN model, a slightly modified version of the VGG11 model (Simonyan & Zisserman, 2015) as described in Appendix E, and conduct training on CIFAR-10 dataset with noiseless and noise-added networks. The results in column (b) of Figure 2 again demonstrate the advantage of our approach: the noise-added networks are much more robust to pruning compared to those trained with noiseless SGD. We present similar results with a larger CNN model in Appendix E.

α	Test Acc.	Pruning Ratio	Test Acc. a.p.
none	96.00 ± 0.48	10.58 ± 0.01	95.95 ± 0.47
1.75	95.01 ± 0.15	40.63 ± 8.55	92.89 ± 1.70
1.8	94.95 ± 0.16	36.05 ± 6.53	93.27 ± 1.33
1.9	95.44 ± 0.24	23.82 ± 5.89	94.94 ± 0.81

Table 5: MNIST, Type-I noise, $n = 5$K.

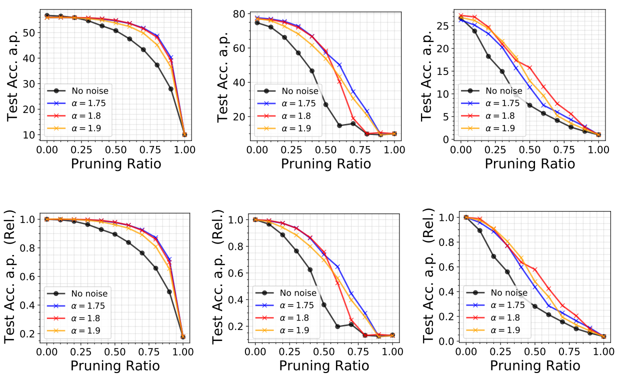
α	Test Acc.	Pruning Ratio	Test Acc. a.p.
none	56.71 ± 0.38	11.60 ± 0.09	56.31 ± 0.50
1.75	51.60 ± 0.22	49.67 ± 2.30	51.48 ± 0.27
1.8	52.36 ± 0.31	41.01 ± 1.36	52.03 ± 0.27
1.9	52.60 ± 0.41	30.25 ± 1.95	52.65 ± 0.36

Table 6: CIFAR-10, Type-I noise, $n = 5$K.

Lastly, we test our approach using a more challenging classification dataset, CIFAR-100. To match the complexity of the task, this time we utilize an FCN with 4 hidden layers of width 2048. We again conduct training until 95% training accuracy. The results can be seen in column (c) of Figure 2, and are consistent with the preceding results: noise-added networks are consistently more robust to pruning than their clean-trained counterpart.

6. Conclusion

We provided a methodological and theoretical framework for provably obtaining compressibility in mean-field neural networks. Our approach requires minimal modification for vanilla SGD and has the same computational complexity. By proving discretization error bounds and propagation of chaos results, we showed that the resulting algorithm is guaranteed to provide compressible parameters. We tested our approach through several experiments, where we showed that in most cases the proposed approach achieves high compressibility, while only slightly compromising accuracy. Moreover, we showed that our methodology produces models that are more robust to pruning in terms of test performance, even with architectures that are beyond our theoretical setting, speaking to the promise of our approach



(a) (b) (c)

Figure 2: Absolute and relative test accuracies for various models and datasets after pruning, as a function of pruning ratio. Column (a): CIFAR-10, FCN with a single hidden layer, $n = 5\text{K}$. Column (b): CIFAR-10, CNN. Column (c): CIFAR-100, FCN with 4 hidden layers, $n = 2048$.

from a practical point of view.

The limitations of our approach are as follows: (i) we consider mean-field networks, it would be of interest to generalize our theoretical results to more sophisticated architectures, (ii) adaptive optimizers are frequently used in various fields of machine learning, thus extending our results from standard SGD to such optimization schemes would be beneficial, (iii) making the dependence between k and n explicit to have a clearer understanding of the required level of overparametrization would be of great interest, (iv) a more detailed understanding of how data distribution, learning rate, noise type, and architecture interact to produce compressibility would be helpful for extending our results to various application domains.

Impact Statement

Due to the mostly theoretical nature of our paper, it does not have a direct negative social impact. Any potential downstream practical applications are to lead to decreased resource consumption for storage and inference.

Acknowledgments

We thank Alain Durmus for valuable discussions. Yijun Wan acknowledges the support from the French Agence Nationale de la Recherche (ANR) under reference ANR-21-CE40-0020 (CONVERGENCE project). Umut Şimşekli is partially supported by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute) and the European Research Council Starting Grant DYNASTY – 101039676.

References

Amini, A., Unser, M., and Marvasti, F. Compressibility of deterministic and random infinite sequences. *IEEE Transactions on Signal Processing*, 59(11):5193–5201, 2011.

Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. Stronger generalization bounds for deep nets via a compression approach. In *International Conference on Machine Learning*, volume 80, pp. 254–263. PMLR, 2018.

Aytekin, C., Cricri, F., and Aksu, E. Compressibility loss for neural network weights. *arXiv preprint arXiv:1905.01044*, 2019.

Baim, D. S., Colucci, W. S., Monrad, E. S., Smith, H. S., Wright, R. F., Lanoue, A., Gauthier, D. F., Ransil, B. J., Grossman, W., and Braunwald, E. The BIDMC Congestive Heart Failure Database, 2000.

Barsbey, M., Sefidgaran, M., Erdogdu, M. A., Richard, G., and Simsekli, U. Heavy tails in SGD and compressibility of overparametrized neural networks. *Advances in Neural Information Processing Systems*, 34:29364–29378, 2021.

Blalock, D., Ortiz, J. J. G., Frankle, J., and Guttag, J. What is the state of neural network pruning? *arXiv preprint arXiv:2003.03033*, 2020.

Burkholz, R., Laha, N., Mukherjee, R., and Gotovos, A. On the existence of universal lottery tickets. *arXiv preprint arXiv:2111.11146*, 2021.

Camuto, A., Wang, X., Zhu, L., Holmes, C., Gürbüzbalaban, M., and Şimşekli, U. Asymmetric heavy tails and implicit bias in Gaussian noise injections. In *International Conference on Machine Learning*. PMLR, 2021.

Cavallazzi, T. Well-posedness and propagation of chaos for Lévy-driven Mckean-Vlasov SDEs under Lipschitz assumptions. *arXiv preprint arXiv:2301.08594*, 2023.

Chen, T., Ji, B., Shi, Y., Ding, T., Fang, B., Yi, S., and Tu, X. Neural network compression via sparse optimization. *arXiv preprint arXiv:2011.04868*, 2020.

Chizat, L. and Bach, F. On the global convergence of gradient descent for over-parameterized models using optimal transport. *Advances in Neural Information Processing Systems*, 31, 2018.

Cont, R. Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative Finance*, 1(2): 223–236, 2001.

da Cunha, A., Natale, E., and Viennot, L. Proving the strong lottery ticket hypothesis for convolutional neural networks. In *ICLR 2022-10th International Conference on Learning Representations*, 2022.

De Bortoli, V., Durmus, A., Fontaine, X., and Simsekli, U. Quantitative propagation of chaos for SGD in wide neural networks. In *Advances in Neural Information Processing Systems*, 2020.

Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Gnedenko, B. V. and Kolmogorov, A. N. *Limit Distributions For Sums Of Independent Random Variables*. Addison-Wesley, Cambridge, 1954.

Gribonval, R., Cevher, V., and Davies, M. E. Compressible distributions for high-dimensional statistics. *IEEE Transactions on Information Theory*, 58(8):5016–5034, 2012.

Gurbuzbalaban, M., Simsekli, U., and Zhu, L. The heavy-tail phenomenon in sgd. In *International Conference on Machine Learning*, pp. 3964–3975. PMLR, 2021.

Hodgkinson, L. and Mahoney, M. Multiplicative noise and heavy tails in stochastic optimization. In *International Conference on Machine Learning*, pp. 4262–4274. PMLR, 2021.

Hsu, D., Ji, Z., Telgarsky, M., and Wang, L. Generalization bounds via distillation. In *International Conference on Learning Representations*, 2021.

Huang, L.-J., Majka, M. B., and Wang, J. Approximation of heavy-tailed distributions via stable-driven SDEs. *Bernoulli*, 27(3):2040–2068, 2021.

Jabir, J.-F., Šiška, D., and Szpruch, Ł. Mean-field neural ODEs via relaxed optimal control. *arXiv preprint arXiv:1912.05475*, 2019.

Jourdain, B., Méléard, S., and Woyczynski, W. Nonlinear SDEs driven by Lévy processes and related PDEs. *arXiv preprint arXiv:0707.2723*, 2007.

Kengne, W. and Wade, M. Sparse-penalized deep neural networks estimator under weak dependence. *arXiv preprint arXiv:2303.01406*, 2023.

Kloeden, P. E., Platen, E., Kloeden, P. E., and Platen, E. *Stochastic Differential Equations*. Springer, 1992.

Krizhevsky, A. Learning multiple layers of features from tiny images, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf. accessed on October 10, 2022.

LeCun, Y., Cortes, C., and Burges, C. Mnist handwritten digit database, 2010. URL http://yann.lecun.com/exdb/mnist. accessed on October 10, 2022.

Lederer, J. Statistical guarantees for sparse deep learning. *AStA Advances in Statistical Analysis*, pp. 1–28, 2023.

Lee, H., Ayed, F., Jung, P., Lee, J., Yang, H., and Caron, F. Deep neural networks with dependent weights: Gaussian process mixture limit, heavy tails, sparsity and compressibility. *arXiv preprint arXiv:2205.08187*, 2022.

Liang, M., Majka, M. B., and Wang, J. Exponential ergodicity for SDEs and Mckean–Vlasov processes with Lévy noise. In *Annales de l'Institut Henri Poincare (B) Probabilites et statistiques*, volume 57, pp. 1665–1701. Institut Henri Poincaré, 2021.

Malach, E., Yehudai, G., Shalev-Schwartz, S., and Shamir, O. Proving the lottery ticket hypothesis: Pruning is all you need. In *International Conference on Machine Learning*, pp. 6682–6691. PMLR, 2020.

Mandelbrot, B. The variation of certain speculative prices. *The Journal of Business*, 36(4):394–419, 1963.

Martin, C. H. and Mahoney, M. W. Traditional and heavy-tailed self regularization in neural network models. *arXiv preprint arXiv:1901.08276*, 2019.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, volume 54, pp. 1273–1282, 2017.

Mei, S., Montanari, A., and Nguyen, P.-M. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33): E7665–E7671, 2018.

Mei, S., Misiakiewicz, T., and Montanari, A. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. In *Conference on Learning Theory*, pp. 2388–2464. PMLR, 2019.

Mikulevičius, R. and Xu, F. On the rate of convergence of strong Euler approximation for SDEs driven by Lévy processes. *Stochastics*, 90(4):569–604, 2018.

Montroll, E. and Bendler, J. On Lévy (or stable) distributions and the Williams-Watts model of dielectric relaxation. *J Stat Phys*, 34:129162, 1984.

Nair, J., Wierman, A., and Zwart, B. *The fundamentals of heavy tails: Properties, emergence, and estimation*, volume 53. Cambridge University Press, 2022.

Nguyen, T. H., Simsekli, U., and Richard, G. Non-asymptotic analysis of fractional Langevin Monte Carlo for non-convex optimization. In *International Conference on Machine Learning*, pp. 4810–4819. PMLR, 2019.

Nikias, C. and Shao, M. *Signal processing with alpha-stable distributions and applications*. Wiley-Interscience, 1995.

O'Neill, J. An overview of neural network compression. *arXiv preprint arXiv:2006.03669*, August 2020.

Papyan, V., Romano, Y., Sulam, J., and Elad, M. Theoretical foundations of deep learning via sparse representations: A multilayer sparse model and its connection to convolutional neural networks. *IEEE Signal Processing Magazine*, 35(4):72–89, 2018.

Pavasovic, K. L., Durmus, A., and Simsekli, U. Approximate heavy tails in offline (multi-pass) stochastic gradient descent. *arXiv preprint arXiv:2310.18455*, 2023.

Ramage, D. and McMahan, B. Federated learning: Collaborative machine learning without centralized training data, 2017. URL https://ai.googleblog.com/2017/04/federated-learning-collaborative.html.

Rotskoff, G. M. and Vanden-Eijnden, E. Trainability and accuracy of neural networks: An interacting particle system approach. *arXiv preprint arXiv:1805.00915*, 2018.

Samoradnitsky, G. *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*. CRC Press, 2017.

Sefidgaran, M., Gohari, A., Richard, G., and Simsekli, U. Rate-distortion theoretic generalization bounds for stochastic learning algorithms. In *Conference on Learning Theory*, pp. 4416–4463. PMLR, 2022.

Shin, J. Y. Compressing heavy-tailed weight matrices for non-vacuous generalization bounds. *arXiv preprint arXiv:2105.11025*, 2021.

Silva, J. F. and Derpich, M. S. On the Characterization of $\mathrm{l_p}$-Compressible Ergodic Sequences. *IEEE Transactions on Signal Processing*, 63(11):2915–2928, 2015.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2015.

Şimşekli, U. Fractional Langevin Monte Carlo: Exploring Lévy driven stochastic differential equations for Markov chain Monte Carlo. In *International Conference on Machine Learning*, pp. 3200–3209. PMLR, 2017.

Şimşekli, U., Gürbüzbalaban, M., Nguyen, T. H., Richard, G., and Sagun, L. On the Heavy-Tailed Theory of Stochastic Gradient Descent for Deep Neural Networks. *arXiv preprint arXiv:1912.00018*, 2019.

Simsekli, U., Sagun, L., and Gurbuzbalaban, M. A tail-index analysis of stochastic gradient noise in deep neural networks. In *International Conference on Machine Learning*, pp. 5827–5837. PMLR, 2019.

Şimşekli, U., Zhu, L., Teh, Y. W., and Gürbüzbalaban, M. Fractional underdamped Langevin dynamics: Retargeting SGD with momentum under heavy-tailed gradient noise. In *International Conference on Machine Learning*. PMLR, 2020.

Sirignano, J. and Spiliopoulos, K. Mean field analysis of deep neural networks. *Mathematics of Operations Research*, 47(1):120–152, 2022.

Suzuki, T., Abe, H., Murata, T., Horiuchi, S., Ito, K., Wachi, T., Hirai, S., Yukishima, M., and Nishimura, T. Spectral pruning: Compressing deep neural networks via spectral analysis and its generalization error. In *International Joint Conference on Artificial Intelligence*, pp. 2839–2846, 2020a.

Suzuki, T., Abe, H., and Nishimura, T. Compression based bound for non-compressed network: unified generalization error analysis of large compressible deep neural network. In *International Conference on Learning Representations*, 2020b.

Sznitman, A.-S. Topics in propagation of chaos. In *Ecole d'été de probabilités de Saint-Flour XIX1989*, pp. 165–251. Springer, 1991.

Zhang, J., Karimireddy, S. P., Veit, A., Kim, S., Reddi, S., Kumar, S., and Sra, S. Why are adaptive methods good for attention models? In *Advances in Neural Information Processing Systems*, volume 33, pp. 21285–21296, 2020.

Zhang, X. and Zhang, X. Ergodicity of supercritical SDEs driven by α-stable processes and heavy-tailed sampling. *Bernoulli*, 29(3):1933–1958, 2023.

Zhou, P., Feng, J., Ma, C., Xiong, C., Hoi, S. C. H., and E, W. Towards theoretically understanding why SGD generalizes better than Adam in deep learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21285–21296, 2020.

Implicit Compressibility of Overparametrized Neural Networks via Heavy-Tailed Noisy Gradient Descent

APPENDIX

The supplementary material is organized as follows.

- In Appendix A, we discuss incompressibility under injection of Gaussian noise.
- In Appendix B, we provide insights about the local convergence properties of the SGD algorithm described in the paper.
- In Appendix C, we provide technical lemmas that are useful for proving Theorem 3.1, Lemma 4.1, and Lemma 4.2.
- In Appendix D, we provide proofs of the theoretical results in the main paper.
- In Appendix E, we present experimental details and extended results for our experiments.
- In Appendix F, implications of our work on federated learning are discussed.

A. Incompressibility of the Brownian Case

In this section, we show that injection of Gaussian noise rather than heavy-tailed noise does not result in compressibility in the sense of the notion defined in Theorem 3.1. More precisely, consider the following SDE, which is equivalent to (5) if $\alpha = 2$:

$$\mathrm{d}\theta_t^{i,n} = b(\theta_t^{i,n}, \mu_t^n)\mathrm{d}t + \sigma \mathrm{dB}_t^{i,n} \tag{7}$$

where $\mathrm{B}_t^{i,n}$, $i = 1, \dots, n$, denote n independent Brownian motion. The next result shows that the neural network trained according to (7) is not compressible, in the sense that there exists at least one value of the compression level κ for which the conclusion of Theorem 3.1 does not hold.

Proposition A.1. *Suppose that Assumption 3.1 holds and let $\hat{\Theta}_t^n \in \mathbb{R}^{p\times n}$ be the matrix with columns being all the parameters $\theta_t^{i,n}$, $i = 1, \dots, n$ obtained by the recursion (7). Then, there exists a relative compression error ε such that for any $\kappa > 0$,*

$$\limsup_{n\to\infty} \frac{\left\|\hat{\Theta}_t^{(\kappa n)} - \hat{\Theta}_t^n\right\|_F}{\left\|\hat{\Theta}_t^n\right\|_F} > \varepsilon.$$

Proof. In the mean-field scaling regime, (De Bortoli et al., 2020, Theorem 10) showed that when the SDEs are driven by Brownian motions, the iterates of SGD have finite second-order moments. Whereas independent samples of Gaussian random variables are not compressible in the sense of Theorem 3.1, see (Gribonval et al., 2012, Proposition 1). This completes the proof. □

B. Local Convergence of SGD with α-Stable Noise

In this part, we provide insights about the local convergence properties of the SGD algorithm described by (2), as guided by (Şimşekli et al., 2019, Theorem 5).

Proposition B.1. *Let n be the number of neurons in the one-hidden-layer neural network and recall that $\hat{\Theta}_k^n$ represents the matrix with columns being the individual neuron weights at iteration k of the SGD described by (2). Under Assumption 3.1, for some $0 < \gamma < \alpha - 1$ and if $\sigma\eta^{1/\alpha-1} > 1$, we have*

$$\min_{0\le k\le K-1} \mathbb{E}\|\nabla R(\hat{\Theta}_k^n)\|^2 \le \frac{R(\hat{\Theta}_0^n) - R^*}{Kn\eta} + \frac{2(2B)^{2+\gamma}\sigma^{1+\gamma}}{(1+\gamma)n^2}\eta^{\frac{\gamma+1-\alpha}{\alpha}}.$$

If particular, if η is chosen such that $\eta \in (n^{2\alpha/(\gamma+1-\alpha)+\epsilon}, \sigma^{\alpha/(\alpha-1)})$ for some $\epsilon > 0$ small enough, the upper bound goes to 0 as K and n go to infinity.

Proof. It suffices to show that A3 and A7 in (Şimşekli et al., 2019) holds with step size ηn, $M = 2B/n$ (in A3) and $\sigma_\gamma^{1+\gamma} = 2(2B/n)^{1+\gamma}(\sigma\eta^{1/\alpha-1})^{1+\gamma}$ (in A7), where B is the constant in Assumption 3.1.

Since bounded Lipschitz-functions are γ-Holder continuous for every $\gamma \in (0, 1]$ (see Lemma B.1), the γ-Holderness A3 follows from Lemma C.1 follows directly from Assumption 3.1 by taking $\gamma \in (0, \alpha - 1)$.

It is not hard to see that the noise $\sigma\eta^{1/\alpha-1}X_k^{i,n}$ of the noisy stochastic gradient noise in (2) is unbiased. Then it remains to verify that the gradient descent updates satisfy certain moment bounds. To this end, note that we have

$$\begin{aligned}\mathbb{E}\left[\|\nabla R(\Theta_k^n) - \sigma\eta^{1/\alpha-1}X_k^{i,n}/n\|^{1+\gamma}|\hat{\Theta}_k^n\right] \leq& 2^{1+\gamma}\mathbb{E}\left[\nabla R(\Theta_k^n)\|^{1+\gamma}|\hat{\Theta}_k^n\right] + 2^{1+\gamma}\mathbb{E}\left[\|\sigma\eta^{1/\alpha-1}X_k^{i,n}/n\|^{1+\gamma}\right]\\ \leq&(2B)^{1+\gamma}(\sigma\eta^{1/\alpha-1}/n)^{1+\gamma} + (2B/n)^{1+\gamma}\\ \leq&2(2B/n)^{1+\gamma}(\sigma\eta^{1/\alpha-1})^{1+\gamma}\end{aligned}$$

Finally, using (Şimşekli et al., 2019, Theorem 5) completes the proof. □

Lemma B.1. *If f is an L-Lipschitz-function which is bounded by M, then for every $\gamma \in (0, 1]$,*

$$\|f(x) - f(y)\| \leq \max(L, (2M)^{1-\gamma}L^\gamma)\|x - y\|^\gamma.$$

Proof. For any x, y in the domain of definition of f, if $\|x - y\| \leq 1$, since f is

$$\|f(x) - f(y)\| \leq L\|x - y\| = L\|x - y\|^{1-\gamma}\|x - y\|^\gamma \leq L\|x - y\|^\gamma.$$

Otherwise if $\|x - y\| \geq 1$,

$$\|f(x) - f(y)\| \leq \|f(x) - f(y)\|^{1-\gamma}\|f(x) - f(y)\|^\gamma \leq (2M)^{1-\gamma}L^\gamma\|x - y\|^\gamma.$$

This completes the proof. □

C. Technical Lemmas

Lemma C.1. *Under Assumption 3.1, it holds that*

$$\|b(\theta_1, \mu_1) - b(\theta_2, \mu_2)\| \leq B \cdot \left(\|\theta_1 - \theta_2\| + \mathbb{E}_{x\sim\pi}\left[|\mu_1(h_x(\cdot)) - \mu_2(h_x(\cdot))|^2\right]^{\frac{1}{2}}\right).$$

Moreover, $\|b(\cdot, \cdot)\|_\infty \leq B$, and if $\mu_1 = \frac{1}{n}\sum_{i=1}^n \delta_{\theta_1^i}$, $\mu_2 = \frac{1}{n}\sum_{i=1}^n \delta_{\theta_2^i}$,

$$\|b(\theta_1, \mu_1) - b(\theta_2, \mu_2)\| \leq B\|\theta_1 - \theta_2\| + \frac{B}{n}\sum_{i=1}^n \|\theta_1^i - \theta_2^i\|.$$

Proof. Recall that

$$b(\theta, \mu) = -\mathbb{E}\left[\partial_1\ell(\mu(h_x(\cdot)), y)\nabla h_x(\theta)\right].$$

Then it follows from the triangular inequality that

$$\|b(\theta_1, \mu_1) - b(\theta_2, \mu_2)\| \leq \|b(\theta_1, \mu_1) - b(\theta_2, \mu_1)\| + \|b(\theta_2, \mu_1) - b(\theta_2, \mu_2)\|. \tag{8}$$

The first term is upper bounded by

$$\begin{aligned}\|b(\theta_1, \mu_1) - b(\theta_2, \mu_1)\| \leq&\mathbb{E}\left[\|\partial_1\ell(\cdot, y)\|_\infty \cdot \|\nabla^2 h_x\|_\infty\right] \cdot \|\theta_2 - \theta_1\|\\ \leq&\mathbb{E}\left[\Phi(y)\Psi(x)\right] \cdot \|\theta_1 - \theta_2\|\\ \leq&\left(\mathbb{E}\left[\Phi^2(y)\Psi^2(x)\right]\right)^{1/2} \cdot \|\theta_1 - \theta_2\|\\ \leq&B \cdot \|\theta_1 - \theta_2\|.\end{aligned} \tag{9}$$

The second term is upper bounded by

$$\begin{aligned}\|b(\theta_2,\mu_1)-b(\theta_2,\mu_2)\| \leq &\mathbb{E}\left[\|\partial_1^2\ell(\cdot,y)\|_\infty\cdot\|\nabla h_x(\cdot)\|_\infty\cdot|\mu_1(h_x(\cdot))-\mu_2(h_x(\cdot))|\right]\\ \leq &\left(\mathbb{E}\left[\Phi^2(y)\Psi^2(x)\right]\right)^{1/2}\mathbb{E}\left[|\mu_1(h_x(\cdot))-\mu_2(h_x(\cdot))|^2\right]^{1/2}\\ \leq &B\cdot\mathbb{E}\left[|\mu_1(h_x(\cdot))-\mu_2(h_x(\cdot))|^2\right]^{1/2}.\end{aligned} \tag{10}$$

The first inequality then follows by combining (8), (9) and (10).

For the boundedness of b in the norm infinity, it is not difficult to see that

$$\begin{aligned}b(\theta,\mu)&=-\mathbb{E}[\partial_1\ell(\mu(h_x(\cdot)),y)\nabla h_x(\theta)]\\ &\leq\mathbb{E}\left[\Phi(y)\Psi(x)\right]\\ &\leq B.\end{aligned}$$

The proof of the last inequality follows by using the first bound and the Cauchy-Schwarz inequality as

$$\begin{aligned}\|b(\theta_1,\mu_1)-b(\theta_2,\mu_2)\| \leq &B\|\theta_1-\theta_2\|+\frac{1}{n}\mathbb{E}_{x\sim\pi}\left[\left(\sum_{i=1}^n h_x(\theta_1^i)-h_x(\theta_2^i)\right)^2\right]^{1/2}\\ \leq &B\|\theta_1-\theta_2\|+\frac{1}{n}\mathbb{E}_{x\sim\pi}\left[\|\nabla h_x\|_\infty\left(\sum_{i=1}^n\|\theta_1^i-\theta_2^i\|\right)^2\right]^{1/2}\\ \leq &B\|\theta_1-\theta_2\|+\frac{1}{n}\mathbb{E}_{x\sim\pi}[\Psi^2(x)]^{1/2}\cdot\sum_{i=1}^n\|\theta_1^i-\theta_2^i\|\\ \leq &B\|\theta_1-\theta_2\|+\frac{B}{n}\sum_{i=1}^n\|\theta_1^i-\theta_2^i\|.\end{aligned}$$

This completes the proof. □

C.1. Propagation of Chaos

Lemma C.2. *Let $(\mathrm{L}_t)_{t\geq 0}$ be an α-stable Lévy process and let $(\mathcal{F}_t)_{t\geq 0}$ be the filtration generated by $(\mathrm{L}_t)_{t\geq 0}$. Then under Assumption 3.1, given the the initial condition $X_0=\xi$, there exists a unique adapted process $(X_t)_{t\in[0,T]}$ for all integrable datum $\xi\in L^1(\mathbb{R}^p)$ such that*

$$X_t=\xi+\int_0^t b(X_t,[X_t])\mathrm{d}t+\mathrm{L}_t.$$

Moreover the first moment of the supremum of the process is bounded

$$\mathbb{E}\left[\sup_{t\leq T}\|X_t\|\right]<+\infty.$$

Proof. The proof follows from Theorem 1 in (Cavallazzi, 2023) by Lemma C.1 where β is set to 1. □

C.2. Compression

Lemma C.3. *Consider a non-integrable probability distribution μ taking values in $\mathbb{R}_+$ such that $\mathbb{E}_{X\sim\mu}[X]=+\infty$. Let $X_1,\ldots,X_n$ be n i.i.d. copies distributed according to μ. Then for any C positive,*

$$\mathbb{P}\left[\frac{1}{n}\sum_{i=1}^n X_i\leq C\right]\xrightarrow{n\to\infty}0.$$

Proof. Using the assumption that μ is non-integrable, let K be a cutoff level for μ such that

$$\mathbb{E}_{X\sim\mu}[\max(X,K)] = C+1.$$

Therefore by the law of large numbers, when goes to infinity,

$$\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\max(X_i,K) = C+1 \qquad \text{almost surely.}$$

Finally, note that

$$\frac{1}{n}\liminf_{n\to\infty}\sum_{i=1}^{n}X_i \geq \frac{1}{n}\lim_{n\to\infty}\sum_{i=1}^{n}\max(X_i,K),$$

which is lower bounded by $(C+1)$ almost surely. Thus the probability that $\frac{1}{n}\sum_{i=1}^{n}X_i$ be smaller than C vanishes for large (infinite) values of n. □

D. Proofs

D.1. Proof of Lemma 4.3

Proof. Recall that $\theta_t = \theta_0 + \int_0^t b(\theta_s,[\theta_s])\mathrm{d}s + \mathrm{L}_t$, then

$$\begin{aligned}
\mathbb{E}\left[\|\theta_t\|^2\right] =&\mathbb{E}\left[\left\langle \theta_0 + \int_0^t b(\theta_s,[\theta_s])\mathrm{d}s + \mathrm{L}_t, \theta_0 + \int_0^t b(\theta_s,[\theta_s])\mathrm{d}s + \mathrm{L}_t\right\rangle\right]\\
=&\mathbb{E}\left[\left\|\theta_0 + \int_0^t b(\theta_s,[\theta_s])\mathrm{d}s\right\|^2\right] + 2\mathbb{E}\left[\left\langle \theta_0 + \int_0^t b(\theta_s,[\theta_s])\mathrm{d}s, \mathrm{L}_t\right\rangle\right]\\
&+\mathbb{E}\left[\|\mathrm{L}_t\|^2\right]\\
\geq&\mathbb{E}\left[\|\mathrm{L}_t\|^2\right] - 2\mathbb{E}\left[\|\theta_0\|\cdot\|\mathrm{L}_t\|\right] - 2\mathbb{E}\left[t\|b(\cdot)\|_\infty\cdot\|\mathrm{L}_t\|\right]\\
\geq&\mathbb{E}\left[\|\mathrm{L}_t\|^2\right] - 2\mathbb{E}\left[\|\theta_0\|\right]\mathbb{E}\left[\|\mathrm{L}_t\|\right] - 2Bt\cdot\mathbb{E}\left[\|\mathrm{L}_t\|\right],
\end{aligned}$$

where the last inequality follows from the independence between the initialization θ_0 and the diffusion noise $(\mathrm{L}_t)_{t\geq 0}$ and by using Lemma C.1. The proof is completed by noticing that

$$\mathbb{E}\left[\|\mathrm{L}_t\|^2\right] = \infty \quad \text{and} \quad \mathbb{E}\left[\|\theta_0\|\right], \mathbb{E}\left[\|\mathrm{L}_t\|\right] < \infty.$$

□

D.2. Proof of Lemma 4.2

Proof. By identification of the diffusion process $(\mathrm{L}_t^{i,n})_{t\geq 0}$ in (5) and (6), the difference of their solutions $\theta_t^{i,n}$ and $\theta_t^{i,\infty}$ for all $t\in[0,T]$ satisfies

$$\theta_t^{i,n} - \theta_t^{i,\infty} = \int_0^t [b(\theta_s^{i,n},\mu_s^n) - b(\theta_s^{i,\infty},[\theta_s^{i,\infty}])]\mathrm{d}s,$$

where $\mu_t = \frac{1}{n}\sum\limits_{i=1}^{n}\delta_{\theta_t^{i,n}}$ and $[\theta_t^{i,\infty}]$ denotes the distribution of $\theta_t^{i,\infty}$. Using Lemma C.1,

$$\begin{aligned}
\|\theta_t^{i,n} - \theta_t^{i,\infty}\| \leq& B\int_0^t \|\theta_s^{i,n} - \theta_s^{i,\infty}\|\mathrm{d}s + B\int_0^t \mathbb{E}_{x\sim\pi}\left[|\mu_s^n(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2\right]^{1/2}\mathrm{d}s\\
\leq& B\int_0^t \sup_{i\leq n}\|\theta_s^{i,n} - \theta_s^{i,\infty}\|\mathrm{d}s + B\int_0^t \mathbb{E}_{x\sim\pi}\left[|\mu_s^n(h_x(\cdot)) - \bar{\mu}_s^n(h_x(\cdot))|^2\right]^{1/2}\mathrm{d}s \qquad (11)\\
&+B\int_0^t \mathbb{E}_{x\sim\pi}\left[|\bar{\mu}_s^n(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2\right]^{1/2}\mathrm{d}s
\end{aligned}$$

where $\bar{\mu}^n_s := \frac{1}{n}\sum\limits_{i=1}^{n} \delta_{\theta_s^{i,\infty}}$, the empirical measure of $\theta_s^{i,\infty}$ for $i = 1, \dots, n$. the last inequality follows from Cauchy-Schwarz inequality. Moreover we have

$$\begin{aligned}\mathbb{E}_{x\sim\pi}[|\mu^n_s(h_x(\cdot) - \bar{\mu}^n_s(h_x(\cdot))|^2]^{1/2} \leq & \mathbb{E}_{x\sim\pi}\left[\left|\frac{\|\nabla h_x\|\infty}{n}\sum_{i=1}^{n}\|\theta_s^{i,n} - \theta_s^{i,\infty}\|\right|^2\right]^{1/2} \\ \leq & \mathbb{E}_{x\sim\pi}[\Psi^2(x)]^{1/2} \cdot \frac{1}{n}\sum_{i=1}^{n}\|\theta_s^{i,n} - \theta_s^{i,\infty}\| \\ \leq & B\sup_{i\leq n}\|\theta_s^{i,n} - \theta_s^{i,\infty}\|.\end{aligned}$$

Plugging the above estimate into (11) yields

$$\|\theta_t^{i,n} - \theta_t^{i,\infty}\| \leq B(1+B)\int_0^t \sup_{i\leq n}\|\theta_s^{i,n} - \theta_s^{i,\infty}\|\mathrm{d}s + B\int_0^t \mathbb{E}_{x\sim\pi}\left[|\bar{\mu}^n_s(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2\right]^{1/2}\mathrm{d}s. \tag{12}$$

Taking the supremum over $i = 1, \dots, n$ and t, and using the fact that

$$\sup\int_{\cdot}(\cdot) \leq \int_{\cdot}\sup(\cdot),$$

we get

$$\begin{aligned}\sup_{t\leq T}\sup_{i\leq n}\|\theta_t^{i,n} - \theta_t^{i,\infty}\| \leq & B(1+B)\int_0^T \sup_{t\leq s}\sup_{i\leq n}\|\theta_t^{i,n} - \theta_t^{i,\infty}\|\mathrm{d}s \\ & + B\int_0^t \mathbb{E}_{x\sim\pi}\left[|\bar{\mu}^n_s(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2\right]^{1/2}\mathrm{d}s.\end{aligned} \tag{13}$$

Let us now estimate $\mathbb{E}\left[|\bar{\mu}^n_s(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2|x\right]^{1/2}$, the expectation under the stable diffusion, rather than the expectation over the data distribution, where the $1/\sqrt{n}$ convergence rate comes from. Indeed for fixed x, $h_x(\theta_s^{i,\infty})$, $i = 1, \dots, n$ are bounded i.i.d. random variables with mean value $[\theta_s^{i,\infty}](h_x(\cdot))$. Therefore

$$\begin{aligned}\mathbb{E}\left[|\bar{\mu}^n_s(h_x(\cdot)) - [\theta_s^{i,\infty}](h_x(\cdot))|^2|x\right]^{1/2} = & \mathbb{E}\left[\left|\frac{1}{n}\sum_{i=1}^{n}h_x(\theta_s^{i,\infty}) - [\theta_s^{i,\infty}](h_x(\cdot))\right|^2\middle| x\right]^{1/2} \\ \leq & \frac{1}{\sqrt{n}}\|h_x(\cdot)\|_\infty \leq \frac{\Psi(x)}{\sqrt{n}}.\end{aligned} \tag{14}$$

Finally, combining (13), (14), the integrability condition Lemma C.2 and using Fubini's theorem, we get

$$\mathbb{E}\left[\sup_{r\leq t}\sup_{i\leq n}\|\theta_r^{i,n} - \theta_r^{i,\infty}\|\right] \leq B(1+B)\int_0^t \mathbb{E}\left[\sup_{r\leq s}\sup_{i\leq n}\|\theta_r^{i,n} - \theta_r^{i,\infty}\|\right]\mathrm{d}s + \frac{Bt\mathbb{E}_{x\sim\pi}[\Psi(x)]}{\sqrt{n}}.$$

Finally, by Gronwall's inequality we get

$$\mathbb{E}\left[\sup_{t\leq T}\sup_{i\leq n}\|\theta_t^{i,n} - \theta_t^{i,\infty}\|\right] \leq (1+B)\left(\frac{BT}{\sqrt{n}} + \frac{B^2T^2\exp(BT(1 + \mathbb{E}_{x\sim\pi}[\Psi(x)]))}{2\sqrt{n}}\right).$$

This completes the proof of Lemma 4.2. □

D.3. Proof of Lemma 4.1

Proof. Similar to in the proof of Lemma 4.2, we have

$$
\begin{aligned}
\sup_{i\le n}\|\theta_\eta^{i,n}-\hat{\theta}_1^{i,n}\| \le& \sup_{i\le n}\int_0^\eta \|b(\theta_t^{i,n},\mu_t^n)-b(\hat{\theta}_0^{i,n},\mu_0^n)\|\mathrm{d}t\\
\le& B\int_0^\eta \sup_{i\le n}\|\theta_t^{i,n}-\theta_0^{i,n}\|+\frac{1}{n}\sum_{j=1}^n\|\theta_t^{j,n}-\theta_0^{j,n}\|\mathrm{d}t\\
\le& B\int_0^\eta 2\|b\|_\infty\cdot t+\sup_{i\le n}\|\mathrm{L}_t^{i,n}\|+\frac{1}{n}\sum_{j=1}^n\|\mathrm{L}_t^{j,n}\|\mathrm{d}t
\end{aligned}
$$

Recall that $\|b\|_\infty\le B$, therefore by taking the expectation and the scaling of the stable process $\mathrm{L}_t^{i,n}$, we get

$$
\begin{aligned}
\mathbb{E}\left[\sup_{i\le n}\|\theta_\eta^{i,n}-\hat{\theta}_1^{i,n}\|\right] \le& B\int_0^\eta\left(2Bt+t^{1/\alpha}\cdot\mathbb{E}\left[\sup_{i\le n}\|\mathrm{L}_1^{i,n}\|+\frac{1}{n}\sum_{j=1}^n\|\mathrm{L}_1^{j,n}\|\right]\right)\mathrm{d}t\\
\le& B^2\eta^2+\frac{B\alpha\cdot\mathbb{E}\left[\sup_{i\le n}\|\mathrm{L}_1^{i,n}\|+\|\mathrm{L}_1^{\alpha}\|\right]}{\alpha+1}\eta^{1+1/\alpha}.
\end{aligned}
\tag{15}
$$

Denote by $C':=\mathbb{E}\left[\sup_{i\le n}\|\mathrm{L}_1^{i,n}\|+\|\mathrm{L}_1^{\alpha}\|\right]$ and $\psi_t(\xi)$ the solution of (5) at time t with initial condition $\xi\in\mathbb{R}^{n\times d}$, which is the matrix of n vectors $\psi_t^{i,n}(\xi)\in\mathbb{R}^d$, $i=1,\dots,n$. At time T which is a multiple of η,

$$
\theta_T^{i,n}-\hat{\theta}_{T/\eta}^{i,n}=\sum_{k=0}^{T/\eta-1}\psi_{T-\eta k}^{i,n}(\hat{\Theta}_k^n)-\psi_{T-\eta(k+1)}^{i,n}(\hat{\Theta}_{k+1}^n),
\tag{16}
$$

where $\hat{\Theta}_k^n$ is the matrix of $\hat{\theta}_k^{i,n}$. Similarly, for each of the terms inside the summation above,

$$
\begin{aligned}
\psi_{T-\eta k}^{i,n}(\hat{\Theta}_k^n)-\psi_{T-\eta(k+1)}^{i,n}(\hat{\Theta}_{k+1}^n)=&\left[\int_{\eta k}^{\eta(k+1)}b^{i,n}(\psi_{t-\eta k}(\hat{\Theta}_k^n))\mathrm{d}t+\mathrm{d}\mathrm{L}_t^{i,n}-(\hat{\theta}_{k+1}^{i,n}-\hat{\theta}_k^{i,n})\right]\\
&-\int_{\eta(k+1)}^{T}\left(b^{i,n}(\psi_{t-\eta k}(\hat{\Theta}_k^n))-b^{i,n}(\psi_{t-\eta(k+1)}(\hat{\Theta}_{k+1}^n))\right)\mathrm{d}t.
\end{aligned}
\tag{17}
$$

Note that the first term in the big bracket is the difference of one-step increment started from $\hat{\Theta}_k^n$. It follows from (15) that

$$
\mathbb{E}\left[\sup_{i\le n}\left\|\int_{\eta k}^{\eta(k+1)}b^{i,n}(\psi_t(\hat{\Theta}_k^n))\mathrm{d}t+\mathrm{d}\mathrm{L}_t^{i,n}-(\hat{\theta}_{k+1}^{i,n}-\hat{\theta}_k^{i,n})\right\|\right]\le B^2\eta^2+\frac{B\alpha\cdot C'}{\alpha+1}\eta^{1+1/\alpha}.
\tag{18}
$$

For the second integral term, similarly we have

$$
\begin{aligned}
&\mathbb{E}\left[\sup_{i\le n}\|b^{i,n}(\psi_{t-\eta k}(\hat{\Theta}_k^n))-b^{i,n}(\psi_{t-\eta(k+1)}(\hat{\Theta}_{k+1}^n))\|\right]\\
\le& B\cdot\mathbb{E}\left[\sup_{i\le n}\|\psi_{t-\eta k}^{i,n}(\hat{\Theta}_k^n)-\psi_{t-\eta(k+1)}^{i,n}(\hat{\Theta}_{k+1}^n)\|\right]+\frac{B}{n}\sum_{j=1}^n\mathbb{E}\left[\|\psi_{t-\eta k}^{j,n}(\hat{\Theta}_k^n)-\psi_{t-\eta(k+1)}^{j,n}(\hat{\Theta}_{k+1}^n)\|\right]
\end{aligned}
\tag{19}
$$

Combining (17), (18) and (19) we get

$$
\begin{aligned}
&\mathbb{E}\left[\sup_{i\le n}\|\psi_{T-\eta k}^{j,n}(\hat{\Theta}_k^n)-\psi_{T-\eta(k+1)}^{j,n}(\hat{\Theta}_{k+1}^n)\|\right]\\
\le& B^2\eta^2+\frac{2B\alpha\cdot C'}{\alpha+1}\eta^{1+1/\alpha}+2B\cdot\int_{\eta(k+1)}^{T}\mathbb{E}\left[\sup_{i\le n}\|\psi_{t-\eta k}^{i,n}(\hat{\Theta}_k^n)-\psi_{t-\eta(k+1)}^{i,n}(\hat{\Theta}_{k+1}^n)\|\right]\mathrm{d}t.
\end{aligned}
$$

Next it follows from Gronwall's inequality that

$$\mathbb{E}\left[\sup_{i\leq n}\|\psi^{j,n}_{T-\eta k}(\hat{\Theta}^n_k)-\psi^{j,n}_{T-\eta(k+1)}(\hat{\Theta}^n_{k+1})\|\right]\leq \exp(2BT)\left(B^2\eta^2+\frac{2B\alpha\cdot C'}{\alpha+1}\eta^{1+1/\alpha}\right).$$

Finally, combining with (16) we obtain

$$\mathbb{E}\left[\sup_{i\leq n}\|\theta^{i,n}_T-\hat{\theta}^{i,n}_{T/\eta}\|\right]\leq T\exp(2BT)\left(B^2\eta+\frac{2B\alpha\cdot C'}{\alpha+1}\eta^{1/\alpha}\right).$$

Then it follows by Lemma D.1 that for some constant C'_α that depends on α, we have

$$C'=\mathbb{E}\left[\sup_{i\leq n}\|\mathrm{L}^{i,n}_1\|+\|\mathrm{L}^{\alpha}_1\|\right]\leq C'_\alpha(n^{1/\alpha}+1).$$

This completes the proof of Lemma 4.1. □

Lemma D.1. *Take n i.i.d. α-stable random variables X^i such that there exists $C_\alpha>0$, for t sufficiently large and $i=1,\dots,n$, $\mathbb{P}[\|X^i\|\geq t]\geq C_\alpha t^{-\alpha}$. If $1<\alpha<2$, then there exists C'_α such that for n sufficiently large,*

$$\mathbb{E}\left[\sup_{i\leq n}\|X^i\|\right]\leq C'_\alpha n^{1/\alpha}$$

Proof. It is not difficult to see from the condition $\mathbb{P}[\|X^i\|\geq t]\geq C_\alpha t^{-\alpha}$ that for large t, it holds that

$$\mathbb{P}\left[\sup_{i\leq n}\|X^i\|\geq t\right]=1-\prod_{i=1}^{n}\mathbb{P}[\|X^i\|<t]\leq 1-\left(1-C_\alpha t^{-\alpha}\right)^n\leq C_\alpha n t^{-\alpha}.$$

Then, for large n we get

$$\begin{aligned}\mathbb{E}\left[\sup_{i\leq n}\|X^i\|\right]&=\int_0^\infty \mathbb{P}\left[\sup_{i\leq n}\|X^i\|\geq t\right]\mathrm{d}t\\&=\sum_{k=-1}^{-\infty}\int_{(n/2^{k+1})^{1/\alpha}}^{(n/2^k)^{1/\alpha}}\mathbb{P}\left[\sup_{i\leq n}\|X^i\|\geq t\right]\mathrm{d}t+\int_0^{n^{1/\alpha}}\mathbb{P}\left[\sup_{i\leq n}\|X^i\|\geq t\right]\mathrm{d}t\\&\leq n^{1/\alpha}\sum_{k=-1}^{-\infty}2^{-k/\alpha}\mathbb{P}\left[\sup_{i\leq n}\|X^i\|\geq (n/2^{k+1})^{1/\alpha}\right]+n^{1/\alpha}\\&\leq C_\alpha n^{1/\alpha}\sum_{k=-1}^{-\infty}2^{k+1-k/\alpha}+n^{1/\alpha}\\&\leq C'_\alpha n^{1/\alpha}\end{aligned}$$

where in the last inequality we set $C'_\alpha=1+2^{1+1/\alpha}/(2-2^{1/\alpha})$. This completes the proof of Lemma D.1. □

D.4. Proof of Theorem 3.1

Definition D.1 (k-term approximation error (Gribonval et al., 2012))**.** The best k-term approximation error $\overline{\sigma}_k(\mathbf{x})$ of a vector $\mathbf{x}$ is defined by

$$\sigma_k(\mathbf{x})=\inf_{\|\mathbf{y}\|_0\leq k}\|\mathbf{x}-\mathbf{y}\|,$$

where $\|\mathbf{y}\|_0$ is the l^0-norm of $\mathbf{y}$, which counts the non-zero coefficients of $\mathbf{y}$. Without mentioned explicitly, $\|\mathbf{x}\|$ denotes the square norm of $\mathbf{x}$.

Proof. Denote by $\hat{\mathbf{w}}_t^n = (\|\hat{\theta}^{1,n}_{\lfloor t/\eta \rfloor}\|, \dots, \|\hat{\theta}^{n,n}_{\lfloor t/\eta \rfloor}\|)$ and $\mathbf{w}_t^* = (\|\theta_t^{1,\infty}\|, \dots, \|\theta_t^{n,\infty}\|)$, where the components $\theta_t^{i,\infty}$ are independent solutions to (6) in Lemma 4.2. Note that the definition of Frobenius matrix norm $\|\cdot\|_F$ gives that

$$\|\hat{\Theta}^{\{\kappa n\}}_{\lfloor t/\eta \rfloor} - \hat{\Theta}^n_{\lfloor t/\eta \rfloor}\|_F = \|\sigma_{\langle \kappa n \rangle}(\hat{\mathbf{w}}_t^n)\|, \quad \|\hat{\Theta}^n_{\lfloor t/\eta \rfloor}\|_F = \|\mathbf{w}_t^\star\|, \tag{20}$$

Therefore it suffices to prove Theorem 3.1 for $\hat{\mathbf{w}}_t^n$. It follows from Lemma 4.2 and Lemma 4.1 that there exists a constant C independent of n for which

$$\mathbb{E}\left[\sup_{i \le n} \|\hat{\theta}^{i,n}_{\lfloor t/\eta \rfloor} - \theta_t^{i,\infty}\|\right] \le \frac{C}{3\sqrt{n}}$$

Then by the Markov's inequality we get

$$\mathbb{P}\left[\sup_{i \le n} \|\hat{\theta}^{i,n}_{\lfloor t/\eta \rfloor} - \theta_t^{i,\infty}\| > \frac{C}{\epsilon\sqrt{n}}\right] \le \epsilon/3. \tag{21}$$

Denote by E the event

$$E := \left\{\sup_{i \le n} \|\hat{\theta}^{i,n}_{\lfloor t/\eta \rfloor} - \theta_t^{i,\infty}\| \le \frac{C}{\epsilon\sqrt{n}}\right\}.$$

If $\sup_{i \le n} \|\hat{\theta}^{i,n}_{\lfloor t/\eta \rfloor} - \theta_t^{i,\infty}\| \le \frac{C}{\epsilon\sqrt{n}}$ and $\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\| \ge \epsilon\|\hat{\mathbf{w}}_t^n\|$, we obtain

$$\begin{aligned}
\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| &\ge \|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\| - \kappa n \frac{C}{\epsilon\sqrt{n}} \\
&\ge \epsilon\|\hat{\mathbf{w}}_t^n\| - C\sqrt{n}\kappa/\epsilon \\
&\ge \epsilon(\|\mathbf{w}_t^\star\| - C\sqrt{n}/\epsilon) - C\sqrt{n}\kappa/\epsilon \\
&= \epsilon\|\mathbf{w}_t^\star\| - C\sqrt{n}(1 + \kappa/\epsilon)
\end{aligned}$$

Therefore plugging in (21), we get

$$\begin{aligned}
&\mathbb{P}\big[\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\| \ge \epsilon\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\|\big] \\
\le&\mathbb{P}\big[\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\| \ge \epsilon\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\|, E^c\big] + \mathbb{P}\big[\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\| \ge \epsilon\|\sigma_{\lfloor \kappa n \rfloor}(\hat{\mathbf{w}}_t^n)\|, E\big] \\
\le&\mathbb{P}\left[\sup_{i \le n} \|\hat{\theta}^{i,n}_{\lfloor t/\eta \rfloor} - \theta_t^{i,\infty}\| > \frac{C}{\epsilon\sqrt{n}}\right] + \mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \epsilon\|\mathbf{w}_t^\star\| - C\sqrt{n}(1 + \kappa/\epsilon)\right] \\
\le&\epsilon/3 + \mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \epsilon\|\mathbf{w}_t^*\| - C\sqrt{n}(1 + \kappa/\epsilon)\right]
\end{aligned} \tag{22}$$

Moreover, there exists $N' > 0$ such that for all $n \ge N'$,

$$\begin{aligned}
&\mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \epsilon\|\mathbf{w}_t^*\| - C\sqrt{n}(1 + \kappa/\epsilon)\right] \\
\le&\mathbb{P}\left[\|\mathbf{w}_t^\star\| \le 2C\sqrt{n}(1 + \kappa/\epsilon)\right] + \mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \frac{\epsilon}{2}\|\mathbf{w}_t^\star\|\right] \\
=&\mathbb{P}\left[\frac{1}{n}\|\mathbf{w}_t^\star\|^2 \le 4C^2(1 + \kappa/\epsilon)^2\right] + \mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \frac{\epsilon}{2}\|\mathbf{w}_t^\star\|\right] \\
\le&\epsilon/3 + \mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \frac{\epsilon}{2}\|\mathbf{w}_t^\star\|\right],
\end{aligned} \tag{23}$$

where the last inequality follows from Lemma C.3. By the independence of the n coordinates of the vetor $\mathbf{w}_t^\star$, Lemma 4.3 and [GCD12, Proposition 1, Part 2], there exists $N'' > 0$, for all $n \ge N''$,

$$\mathbb{P}\left[\|\sigma_{\lfloor \kappa n \rfloor}(\mathbf{w}_t^\star)\| \ge \frac{\epsilon}{2}\|\mathbf{w}_t^\star\|\right] \le \epsilon/3. \tag{24}$$

Finally, combining (20), (22), (23) and (24) terminates the proof. □

E. Experimental Details and Additional Results

E.1. Software and Hardware Requirements

The experiments have been implemented in Python, using the deep learning framework PyTorch. Experiments were run on the server of an educational institution, using NVIDIA 1080 and 1080 Ti GPUs. The experiments published in the main paper and the Appendix amounted to an estimated GPU time of 1200 hours in total. Pruning and analysis is estimated to have taken an additional 40 GPU hours. We refer the reader to our source code for further implementation details[8].

E.2. Datasets

The ECG5000 dataset (Baim et al., 2000) consists of 5000 20-hour long electrocardiograms interpolated by sequences of length 140 to discriminate between normal and abnormal heart beats of a patient that has severe congestive heart failure. After random shuffling, we use 500 sequences for the training phase and 4500 sequences for the test phase. The MNIST database (LeCun et al., 2010) of black and white handwritten digits consists of a training set of 60,000 examples and a test set of 10,000 examples of dimensions 28 x 28. CIFAR-10 and CIFAR-100 are two other image classification datasets (Krizhevsky, 2009), including 32 x 32 x 3 color images of objects or animals, making up 10 and 100 classes, respectively. We use the default split of 50,000 training and 10,000 test examples.

E.3. Models and Training Hyperparameters

The models used in the experiments are fully connected networks (FCN) and convolutional neural networks (CNN). All models include ReLU activations, and do not include any bias nodes nor any advanced layer structures such as batch normalization or residual connections. Due to the number of parameters being low compared to other layers, last linear layers of the models are not added noise during training, and are not included in pruning or computation of pruning ratios during evaluation. As described in the paper, we use FCNs with 1 or 4 hidden layers in different experiments. The CNN used in the experiments is a modified version of VGG11 (Simonyan & Zisserman, 2015), and has the structure

$$128, M, 256, M, 512, 512, M, 1024, 1024, M, 1024, 1024, M,$$

followed by a final linear layer, where the numbers refer to convolutional layer widths with 3 x 3 filters, followed by ReLU activation functions, and M's refer to 2 x 2 max pooling operations. In Appendix E.5 we experiment with a larger version of this model, abbreviated as CNN-L, this time following the layer ordering of VGG16 (Simonyan & Zisserman, 2015), with the following structure:

$$128, 128, M, 256, 256, M, 512, 512, 512, M, 1024, 1024, 1024, M, 1024, 1024, 1024, M.$$

All models in all experiments are trained until 95% training accuracy, after which the training is concluded. As described in the main paper, no adaptive optimizers has been used in any of the experiments. For each experiment, σ values have been selected to be as large as possible, without incurring divergence during the training or dramatic performance loss to the pruned model (training accuracy > 0.90). In experiments on robustness to pruning, the σ that lead to largest area under the pruning-accuracy curve was selected. Learning rates, batch sizes, and σ values have been provided in the Table 7. We note that the batch sizes for CIFAR-10 + CNN and CIFAR-100 experiments have been selected to be considerably smaller than in MNIST and other CIFAR-10 experiments, due to the former being more memory intensive. Given the very limited additional computational overhead of our method, our approach can easily be combined with standard hyperparameter selection methods, such as using a held-out validation set.

E.4. Extended Results

Tables 8-13 include the extended version of the results presented in the main paper in Tables 1-6, with training set accuracies before and after pruning (a.p.) included.

E.5. Results with a Larger CNN Model

To see whether our results continue to hold as we increase model size, we train a larger convolutional neural network (CNN-L, described in Appendix E.5) model on CIFAR-10, and examine model's robustness against pruning. We train the

[8] https://github.com/mbarsbey/imp_comp

Experiment	LR	B	σ ($\alpha = 1.75$)	σ ($\alpha = 1.8$)	σ ($\alpha = 1.9$)
ECG5000, FCN, $n = 2$K, Type-I	0.0001	500	0.40	0.30	0.25
ECG5000, FCN, $n = 10$K, Type-I	0.0001	500	1.10	1.20	0.80
ECG5000, FCN, $n = 10$K, Type-II	0.0001	500	2.00	1.90	3.60
ECG5000, FCN, $n = 10$K, Type-III	0.0001	500	5.25	3.90	2.75
MNIST, FCN, $n = 5$K, Type-I	0.25	5000	0.001	0.00125	0.0011
CIFAR-10, FCN, Type-I (Table 6)	0.10	5000	0.0003	0.0003	0.0003
CIFAR-10, FCN, Type-I (Fig. 2(a))	0.10	5000	0.0001	0.0001	0.0001
CIFAR-10, CNN, Type-I	0.01	100	0.000075	0.0001	0.000075
CIFAR-100, FCN, Type-I	0.01	100	0.00005	0.000075	0.00009

Table 7: Hyperparameters for the experiments presented in the main paper and the appendices, including learning rate (LR), batch size (B), and chosen noise scales (σ) for various noise tail indices (α).

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
no noise	95.02 ± 0.06	94.15 ± 0.51	11.42 ± 0.04	94.76 ± 0.84	94.12 ± 0.85
1.75	95.02 ± 0.06	94.15 ± 0.51	49.38 ± 22.72	91.46 ± 10.64	90.50 ± 10.64
1.8	95.02 ± 0.06	94.15 ± 0.51	37.22 ± 14.51	93.84 ± 1.92	93.26 ± 1.85
1.9	95.02 ± 0.06	94.15 ± 0.51	25.90 ± 11.02	94.08 ± 2.12	93.56 ± 1.41

Table 8: ECG5000, Type-I noise, $n = 2$K.

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
no noise	95.02 ± 0.06	94.16 ± 0.28	11.44 ± 0.01	94.40 ± 2.12	93.75 ± 1.77
1.75	95.02 ± 0.06	94.16 ± 0.28	54.72 ± 18.46	90.20 ± 5.79	90.12 ± 5.42
1.8	95.02 ± 0.06	94.16 ± 0.28	44.90 ± 11.04	91.52 ± 7.43	91.45 ± 6.36
1.9	95.02 ± 0.06	94.16 ± 0.28	30.86 ± 8.11	93.44 ± 1.43	93.21 ± 1.02

Table 9: ECG5000, Type-I noise, $n = 10$K.

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
1.75	95.32 ± 0.32	94.17 ± 0.51	54.40 ± 18.92	93.88 ± 2.10	93.41 ± 2.24
1.8	95.48 ± 0.70	94.72 ± 1.22	39.85 ± 13.66	92.28 ± 3.28	91.68 ± 3.13
1.9	95.56 ± 0.49	94.62 ± 0.36	22.05 ± 9.87	93.58 ± 3.32	93.31 ± 2.05

Table 10: ECG5000, Type-II noise, $n = 10$K.

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
1.75	95.28 ± 0.36	94.35 ± 0.64	51.25 ± 16.55	92.30 ± 3.47	91.93 ± 3.13
1.8	95.26 ± 0.39	94.10 ± 0.49	38.09 ± 16.67	93.50 ± 1.93	92.74 ± 2.23
1.9	95.56 ± 0.52	94.51 ± 0.47	21.92 ± 9.56	93.58 ± 3.60	93.18 ± 2.83

Table 11: ECG5000, Type-III noise, $n = 10$K.

model with a learning rate of 0.001 and a batch size of 100, fix $\alpha = 1.8$, and select σ to be 0.0001 for Type-I and Type-III noises, and 0.000075 for Type-II noise. The results are presented in Figure 3, and suggest that benefits conferred by our methodology continue to hold as we increase model size and baseline performance. The performance differences between Type-I and the other noises highlight the investigation of the effect of noise type on compressibility and robustness as a fruitful future research direction, as discussed in Section 6.

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
no noise	96.32 ± 0.68	96.00 ± 0.48	10.58 ± 0.01	96.30 ± 0.67	95.95 ± 0.47
1.75	95.48 ± 0.20	95.01 ± 0.15	40.63 ± 8.55	93.14 ± 1.54	92.89 ± 1.70
1.8	95.42 ± 0.25	94.95 ± 0.16	36.05 ± 6.53	93.62 ± 1.32	93.27 ± 1.33
1.9	95.88 ± 0.36	95.44 ± 0.24	23.82 ± 5.89	95.30 ± 0.89	94.94 ± 0.81

Table 12: MNIST, Type-I noise, $n = 5$K.

α	Train Acc.	Test Acc.	Pruning Ratio	Train Acc. a.p.	Test Acc. a.p.
no noise	96.52 ± 0.85	56.71 ± 0.38	11.60 ± 0.09	96.13 ± 0.91	56.31 ± 0.50
1.75	95.56 ± 0.24	51.60 ± 0.22	49.67 ± 2.30	95.28 ± 0.23	51.48 ± 0.27
1.8	95.86 ± 0.36	52.36 ± 0.31	41.01 ± 1.36	95.61 ± 0.51	52.03 ± 0.27
1.9	96.08 ± 0.21	52.60 ± 0.41	30.25 ± 1.95	96.17 ± 0.21	52.65 ± 0.36

Table 13: CIFAR-10, Type-I noise, $n = 5$K.

F. Implications on Federated Learning

The federated learning (FL) setting (McMahan et al., 2017; Ramage & McMahan, 2017) is one in which there are a number of devices or clients, say n; all equipped with the same neural network model and each holding an independent own dataset. Every client learns an individual (or local) model from its own dataset, e.g., via Stochastic Gradient Descent (SGD). The individual models are aggregated by a *parameter server* (PS) into a global model and then sent back to the devices, possibly over multiple rounds of communication between them. The rationale is that the individually learned models are refined progressively by taking into account the data held by other devices; and, at the end the training process, all relevant features of all devices' datasets are captured by the final aggregated model.

The results of this paper are useful towards a better understanding of the *compressibility* of the models learned by the various clients in this FL setting. Specifically, viewing each neuron of the hidden layer of the setup of this paper as if it were a distinct client, the results that we establish *suggest* that if the local models are learned via heavy-tailed SGD this would enable a better compressibility of them. This is particularly useful for resource-constrained applications of FL, such as in telecommunication networks where bandwidth is scarce and latency is important.

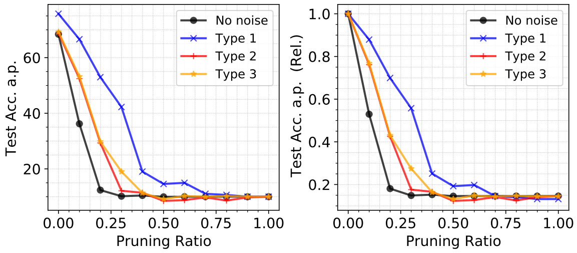


Figure 3: Absolute and relative test accuracies for CNN-L model trained on CIFAR-10, as a function of pruning ratio.