Asymmetric Heavy Tails and Implicit Bias in Gaussian Noise Injections

Alexander Camuto [*1] **Xiaoyu Wang** [*2] **Lingjiong Zhu** [2] **Mert Gürbüzbalaban** [3] **Chris Holmes** [1]
Umut Şimşekli [4]

Abstract

Gaussian noise injections (GNIs) are a family of simple and widely-used regularisation methods for training neural networks where one injects additive or multiplicative Gaussian noise to the network activations at every iteration of the optimisation algorithm, which is typically chosen as stochastic gradient descent (SGD). In this paper we focus on the so-called 'implicit effect' of GNIs, which is the effect of the injected noise on the dynamics of SGD. We show that this effect induces an *asymmetric heavy-tailed noise* on SGD gradient updates. In order to model this modified dynamics, we first develop a Langevin-like stochastic differential equation that is driven by a general family of *asymmetric* heavy-tailed noise. Using this model we then formally prove that GNIs induce an 'implicit bias', which varies depending on the heaviness of the tails and the level of asymmetry. Our empirical results confirm that different types of neural networks trained with GNIs are well-modelled by the proposed dynamics and that the implicit effect of these injections induces a bias that degrades the performance of networks.

1. Introduction

Noise injections are a family of methods that involve adding or multiplying samples from a noise distribution to the weights and activations of a neural network during training. The most commonly used distributions are Bernoulli distributions and Gaussian distributions (Srivastava et al., 2014; Poole et al., 2014) and the noise is most often inserted at the level of network activations.

[*]Equal contribution [1]Alan Turing Institute, University of Oxford, Oxford, UK [2]Department of Mathematics, Florida State University, Tallahassee, USA [3]Department of Management Science and Information Systems, Rutgers Business School, Piscataway, USA [4] INRIA - Département d'Informatique de l'École Normale Supérieure - PSL Research University, Paris, France. Correspondence to: Alexander Camuto <acamuto@turing.ac.uk>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

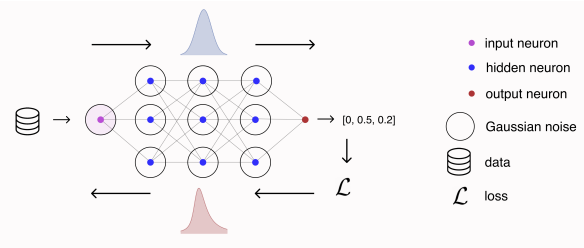

Figure 1. Illustration of the effect of GNIs *added* to a network's activations. Each colored dot represents a neuron's activations. We add GNIs, represented as circles, to each layer's activations bar the output layer. Perhaps counter-intuitively, though the forward pass experiences Gaussian noise, gradient updates in the backward pass experience heavy-tailed asymmetric noise.

Though the regularisation conferred by Gaussian noise injections (GNIs) can be observed empirically, and there have been many studies on the benefits of noising data (Bishop, 1995; Cohen et al., 2019; Webb, 1994), the mechanisms by which these injections operate are not fully understood. Recently, the *explicit* effect of GNIs, which is the added term to the loss function obtained when marginalising out the injected noise, has been characterised analytically (Camuto et al., 2020): it corresponds to a penalisation in the Fourier domain which improves model generalisation.

Here we extend this analysis and focus on the *implicit* effect of GNIs. This is the effect of the remaining noise that has been marginalised out when studying the explicit effect. In particular we focus on the manner in which such noise alters the dynamics of Stochastic Gradient Descent (SGD) (Wei et al., 2020; Zhang et al., 2017). We show that the implicit effect is driven by an *asymmetric heavy-tailed noise* on the SGD gradient updates, as illustrated in Figure 1.

To study the effect of this gradient noise, we model the dynamics of SGD for a network experiencing GNIs by a stochastic differential equation (SDE) driven by an *asymmetric heavy-tailed* α-stable noise. We demonstrate that this model captures the dynamics of networks trained with GNIs and we show that the stationary distribution of this process becomes arbitrarily distant from the so-called *Gibbs measure*, whose modes exactly match the local minima of the loss function, as the gradient becomes increasingly heavy-tailed and asymmetric. Heavy-tailed and asymmetric

gradient noise thus degrades network performance and this suggests that models trained with the full effect of GNIs will underperform networks trained solely with the explicit effect. We confirm this experimentally for a variety of dense and convolutional networks.[1]

2. Background

Stable Distributions. The Generalised Central Limit Theorem (GCLT) (Gnedenko & Kolmogorov, 1954) states that for a sequence of independent and identically distributed (i.i.d.) random variables whose distribution has a power-law tail with index $0 < \alpha < 2$, the normalised sum converges to a heavy-tailed distribution called the α-stable distribution ($\mathcal{S}_\alpha$) as the number of summands grows. An α-stable distributed random variable X is denoted by $X \sim \mathcal{S}_\alpha(\sigma, \theta, \mu)$, where $\alpha \in (0, 2]$ is the *tail-index*, $\theta \in [-1, 1]$ is the *skewness* parameter, $\sigma \geq 0$ is the *scale* parameter, and $\mu \in \mathbb{R}$ is called the *location* parameter. The mean of X coincides with μ if $\alpha > 1$, and otherwise the mean of X is undefined. In this work, we always assume $\mu = 0$. The parameter θ is a measure of asymmetry. We say that X follows a *symmetric* α-stable distribution denoted as $\mathcal{S}\alpha\mathcal{S}(\sigma) = \mathcal{S}_\alpha(\sigma, 0, 0)$ if $\theta = 0$ (and $\mu = 0$). The parameter $\alpha \in (0, 2]$ determines the tail thickness of the distribution, and $\sigma > 0$ measures the spread of X around its mode. Note that when $\alpha < 2$, α-stable distributions have heavy tails such that their moments are finite only up to the order α.

The probability density function (p.d.f.) of an α-stable random variable, $\alpha \in (0, 2]$, does not have a closed-form expression except for a few special cases. When $\alpha = 1$ and $\alpha = 2$, the symmetric α-stable distribution reduces to the Cauchy and the Gaussian distributions, respectively, (cf. Section 1.1. in (Samorodnitsky & Taqqu, 1994)). By their flexibility, such distributions can model many complex stochastic phenomena for which exact analytic forms are intractable (Sarafrazi & Yazdi, 2019; Fiche et al., 2013).

Lévy Processes. A Lévy process (motion) is a stochastic process with independent, stationary increments. Formally, $\mathbf{L}_t$ is Lévy process if

(i) $\mathbf{L}_0 = 0$ almost surely;
(ii) For any $t_0 < t_1 < \cdots < t_N$, the increments $\mathbf{L}_{t_n} - \mathbf{L}_{t_{n-1}}$ are independent, $n = 1, 2, \ldots, N$;
(iii) The difference $\mathbf{L}_t - \mathbf{L}_s$ and $\mathbf{L}_{t-s}$ have the same distribution;
(iv) $\mathbf{L}_t$ is continuous in probability, i.e. for any $\delta > 0$ and $s \geq 0$, $\mathbb{P}(|\mathbf{L}_t - \mathbf{L}_s| > \delta) \to 0$ as $t \to s$.

The α-stable Lévy process is an important class of Lévy processes. In particular, for $\alpha \in (0, 2]$, let $\mathbf{L}_t^{\alpha,\theta}$ denote the d-dimensional α-stable Lévy process with independent components, i.e. each component is an independent scalar α-stable Levy motion (Duan, 2015) such that $\mathbf{L}_{t-s}^{\alpha,\theta}$ has the distribution $\mathcal{S}_\alpha((t-s)^{1/\alpha}, \theta, 0)$ for any $s < t$.

Stochastic Gradient Descent and Differential Equations. Let $\mathcal{D}$ be a training dataset composed of data-label pairs of the form $(\mathbf{x}, \mathbf{y})$, and let $\mathbf{w} \equiv \{\mathbf{W}_1, \ldots, \mathbf{W}_L\} \in \mathbb{R}^d$ be the d parameters of an L layer neural network in vector form. When a neural network operates on input data $\mathbf{x}$, we obtain the activations $\mathbf{h} \equiv \{\mathbf{h}_0, \ldots, \mathbf{h}_{L-1}\}$, where $\mathbf{h}_0 = \mathbf{x}$ and $\mathbf{h} \in \mathbb{R}^{n_0 + \cdots + n_L}$ where n_i is the number of neurons in the i^{th} layer: we consider a non-linearity $\kappa : \mathbb{R} \to \mathbb{R}$, $\mathbf{h}_i(\mathbf{x}) = \kappa(\mathbf{W}_i \mathbf{h}_{i-1}(\mathbf{x}))$, where κ is applied element-wise to each coordinate of its argument. In supervised settings, our objective is to find the optimal parameters $\mathbf{w}_*$ that minimise the negative log-likelihood $-\log p_{\mathbf{w}}(\mathbf{y}|\mathbf{x})$ of the labels $\mathbf{y}$, given the parameters $\mathbf{w}$ and data $\mathbf{x}$:

$$\mathbf{w}_* = \arg\min_{\mathbf{w}} \mathcal{L}(\mathcal{D}; \mathbf{w}),$$
$$\mathcal{L}(\mathcal{D}; \mathbf{w}) := -\mathbb{E}_{\mathbf{x},\mathbf{y}\sim\mathcal{D}} \left[\log p_{\mathbf{w}}(\mathbf{y}|\mathbf{x})\right] . \tag{2.1}$$

SGD and its variants are the most prevalent optimisation routines that underpin the training of very large neural networks. Under SGD, we estimate equation (2.1) by sampling a *random* mini-batch of data-label pairs $\mathcal{B} \subset \mathcal{D}$,

$$\mathcal{L}(\mathcal{B}; \mathbf{w}) = -\mathbb{E}_{\mathbf{x},\mathbf{y}\sim\mathcal{B}} \left[\log p_{\mathbf{w}}(\mathbf{y}|\mathbf{x})\right] \approx \mathcal{L}(\mathcal{D}; \mathbf{w}). \tag{2.2}$$

SGD optimises this equation and approximates $\mathbf{w}_*$ using iterative parameter updates. At training step k

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta \nabla \mathcal{L}(\mathcal{B}_{k+1}; \mathbf{w}_k), \tag{2.3}$$

where η is the step-size for updates (the network's learning rate) (Robbins & Monro, 1951; Ruder, 2016).

Studying the dynamics of SGD allows us to understand the subtle effects that batching may have on neural network training. The similarities between the SGD algorithm and Langevin diffusions (Roberts & Stramer, 2002) have inspired many studies modelling the dynamics of SGD using stochastic differential equations (SDEs) under different noise conditions (Agapiou et al., 2014; Şimşekli et al., 2019; Raginsky et al., 2017; Gao et al., 2018; 2020; Jastrzębski et al., 2017; Li et al., 2017). In this approach, one models the discrete SGD updates (2.3), as the discretisation of a continuous-time stochastic process, making assumptions about the properties of the 'noise' that drives this process (Mandt et al., 2016; Jastrzębski et al., 2017). This noise stems from the stochasticity in approximating the 'true' gradient over the dataset $\nabla\mathcal{L}(\mathcal{D}; \mathbf{w}_k)$ with that of a mini-batch $\mathcal{B}$, $\nabla\mathcal{L}(\mathcal{B}; \mathbf{w}_k)$. We denote this noise as,

$$U_{k+1}(\mathbf{w}) := \nabla\mathcal{L}(\mathcal{D}; \mathbf{w}_k) - \nabla_{\mathbf{w}_k}\mathcal{L}(\mathcal{B}_{k+1}; \mathbf{w}_k). \tag{2.4}$$

The most prevalent assumption is that the gradient noise admits a multi-variate Gaussian noise: $U_k(\mathbf{w}) \sim \mathcal{N}(0, \sigma^2\mathbf{I})$.

[1] See https://github.com/alexander-camuto/asym-heavy-tails-bias-GNI for all code.

This is rationalised by the central limit theorem, as the sum of estimation errors in equation (2.4) is approximately Gaussian for sufficiently large batches. Under this assumption, we can rewrite the SGD parameter update as:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta \nabla \mathcal{L}(\mathcal{D}; \mathbf{w}_k) + \eta U_{k+1}(\mathbf{w}). \tag{2.5}$$

Then, one obtains the following continuous-time SDE to approximate the gradient updates (Welling & Teh, 2011; Mandt et al., 2016; Jastrzębski et al., 2017):

$$d\mathbf{w}_t = -\nabla \mathcal{L}(\mathcal{D}; \mathbf{w}_t)dt + \sqrt{\eta \sigma^2} d\mathbf{B}_t, \tag{2.6}$$

where $\mathbf{B}_t$ is the Brownian motion in $\mathbb{R}^d$ and σ is the assumed noise variance for U. However, recent work suggests that the Gaussian assumption might not be always appropriate (Gürbüzbalaban et al., 2021; Hodgkinson & Mahoney, 2020), and connectedly, the gradient noise is observed to be heavy-tailed in different settings (Şimşekli et al., 2019; Zhou et al., 2020). Under this noise assumption, the corresponding SDE is such that $\mathbf{B}_t$ in (2.6) is replaced with the *symmetric* stable process $\mathbf{L}_t^{\alpha,0}$ where $\theta = 0$:

$$d\mathbf{w}_t = -\nabla \mathcal{L}(\mathcal{D}; \mathbf{w}_t)dt + \eta^{(\alpha-1)/\alpha} \sigma d\mathbf{L}_t^{\alpha,0}. \tag{2.7}$$

Gaussian Noise Injections. GNIs are regularisation methods that consist of injecting Gaussian noise to the network activations. More precisely, let $\boldsymbol{\epsilon}$ be a collection of 'noise vectors' injected to the network activations at each layer *except the final layer*: $\boldsymbol{\epsilon} \equiv \{\boldsymbol{\epsilon}_0, \ldots, \boldsymbol{\epsilon}_{L-1}\}$, where $\boldsymbol{\epsilon}_i \in \mathbb{R}^{n_i}$, $\boldsymbol{\epsilon} \in \mathbb{R}^{n_0 + \cdots + n_{L-1}}$ and n_i is the number of neurons in the i^{th} layer. We have two values for an activation: the soon-to-be noised value $\widehat{\mathbf{h}}_i$, and the subsequently noised value $\widetilde{\mathbf{h}}_i$. For a multi-layer perceptron (MLP),

$$\widehat{\mathbf{h}}_i(\mathbf{x}) = \kappa\left(\mathbf{W}_i \widetilde{\mathbf{h}}_{i-1}(\mathbf{x})\right), \qquad \widetilde{\mathbf{h}}_i = \widehat{\mathbf{h}}_i \circ \boldsymbol{\epsilon}_i, \tag{2.8}$$

where $\circ$ is some element-wise operation (e.g., addition or multiplication). For additive GNIs typically, $\boldsymbol{\epsilon}_i \sim \mathcal{N}\left(0, \sigma_i^2 \mathbf{I}\right)$, and for multiplicative GNIs we often use $\boldsymbol{\epsilon}_i \sim \mathcal{N}\left(1, \sigma_i^2 \mathbf{I}\right)$, where $\mathcal{N}$ is the Gaussian distribution. The 'accumulated' noise at each layer, induced by the noise injected to the layer *and* in previous layers is

$$\mathcal{E}_i(\mathbf{x}; \mathbf{w}, \boldsymbol{\epsilon}) = \widetilde{\mathbf{h}}_i(\mathbf{x}) - \mathbf{h}_i(\mathbf{x}). \tag{2.9}$$

Given that GNIs are commonly used as regularisation methods (Camuto et al., 2020; Dieng et al., 2018; Srivastava et al., 2014; Poole et al., 2014; Kingma et al., 2015; Bishop, 1995), our goal is to understand better the mechanisms by which they affect neural networks.

3. The Implicit Effect of GNIs

Recently, Camuto et al. (2020) showed that the effect of GNIs on the cost function can be expressed as a term $\Delta\mathcal{L}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon})$ that is added to the loss, i.e.,

$$\widetilde{\mathcal{L}}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon}) := \mathcal{L}(\mathcal{B}; \mathbf{w}) + \Delta\mathcal{L}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon}), \tag{3.1}$$

where $\widetilde{\mathcal{L}}$ is the modified loss that SGD ultimately aims to minimise. The term $\Delta\mathcal{L}$ can be further broken down into explicit and implicit effects, as described in Section 1.

We build on the approach of Camuto et al. (2020) and define the explicit effect as the additional term obtained on the loss when we marginalise out the noise we have injected. It offers a consistently positive objective for gradient descent to optimise and we denote it as $\mathbb{E}_{\boldsymbol{\epsilon}}(\Delta\mathcal{L}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon}))$. The implicit effect is then the remainder of the terms marginalised out in the explicit effect:

$$E_{\mathcal{L}}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon}) := \Delta\mathcal{L}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon}) - \mathbb{E}_{\boldsymbol{\epsilon}}(\Delta\mathcal{L}(\mathcal{B}; \mathbf{w}, \boldsymbol{\epsilon})). \tag{3.2}$$

While the explicit effect focuses on the consistent and *non-stochastic* regularisation induced by GNIs, the implicit effect instead studies the effect of the *inherent stochasticity* of GNIs. This term does not offer a consistent objective for SGD to minimise. Rather we show that it affects neural network training by way of the *heavy-tailed and skewed* noise it induces on gradient updates.

Subsequently, we first characterise the tail properties of the noise accumulated during the forward pass. As this accumulated noise defines the implicit effect we can then use this result to show that gradients induced by the implicit effect are heavy-tailed and skewed and apt to be modelled by heavy-tailed and asymmetric α-stable noise.

The Properties of the Accumulated Noise. Before considering the properties of the implicit effect gradients, we first need to study the noise that is accumulated during the *forward pass* of a neural network experiencing GNIs. Here we use asymptotic analysis to bound the moments of this noise, allowing us to characterise the tail properties of gradients using the broad class of *sub-Weibull* distributions that include a range of heavy-tailed distributions (Vladimirova et al., 2019; 2020; Kuchibhotla & Chakrabortty, 2018).

Definition 3.1. *(Asymptotic order) A positive sequence a_m is of the same order as another positive sequence b_m ($a_m \lesssim b_m$) if $\exists C > 0$ such that $\frac{a_m}{b_m} \leq C \; \forall m \in \mathbb{N}$. a_m is of the same order of magnitude as b_m ($a_m \asymp b_m$, i.e. 'asymptotically equivalent') if there exist some $c, C > 0$ such that: $c \leq \frac{a_m}{b_m} \leq C$ for any $m \in \mathbb{N}$.*

Definition 3.2. *(Sub-Weibull distributions), : We say that a random variable X is sub-Weibull (Vladimirova et al., 2020; Kuchibhotla & Chakrabortty, 2018) with tail parameter r if $\|X\|_m \lesssim m^r, r > 0$, where $\|X\|_m := \mathbb{E}\left[|X|^m\right]^{\frac{1}{m}}$. In this case, we write $X \sim \text{subW}(r)$. Such distributions satisfy the tail bound $\mathbb{P}(|X| > x) \leq 2e^{-(\frac{x}{C})^{\frac{1}{r}}}$, where $C > 0$ is some constant. Note that for $r = \frac{1}{2}$ and $r = 1$ we recover the sub-Gaussian and sub-exponential families and that if $X \sim \text{subW}(r')$, then $X \sim \text{subW}(r)$ for $r > r'$.*

As r increases the tail distribution becomes heavier-tailed.

To simplify our analysis we consider activation functions ϕ that obey the extended envelope property.

Definition 3.3 (Extended Envelope Property (Vladimirova et al., 2019):)**.** *A non-linear function $\kappa : \mathbb{R} \to \mathbb{R}$ is said to obey the extended envelope property if $\exists\ c_1, c_2 \geq 0,\ d_1, d_2 \geq 0$ such that:*

- $|\kappa(x)| \geq c_1 + d_1|x|$, *for any* $x \in \mathbb{R}^+$ *or* $x \in \mathbb{R}^-$;
- $|\kappa(x)| \leq c_2 + d_2|x|$, *for any* $x \in \mathbb{R}$.

Activation functions that obey this property, such as ReLU, are broadly moment preserving (Vladimirova et al., 2020). Using this property, we can characterise the moments for the l^{th} noised activation in a layer i, $\widetilde{h}_{i,l}(\mathbf{x})$.

Lemma 3.1. *For feed-forward neural networks with an activation function ϕ that obeys the extended envelope property, the noised activations at each layer $i < L-1$, resulting from additive-GNIs $\boldsymbol{\epsilon}$ obey*

$$\left\|\widetilde{h}_{i,l}(\mathbf{x})\right\|_m \lesssim \sqrt{m}, \quad \text{for any } m \geq 1;\ l = 1, \ldots, n_i\,.$$

For multiplicative-GNIs we have,

$$\left\|\widetilde{h}_{i,l}(\mathbf{x})\right\|_m \lesssim m^{\frac{i+1}{2}}, \quad \text{for any } m \geq 1;\ l = 1, \ldots, n_i\,,$$

where n_i is the dimensionality of the i^{th} layer.

Lemma 3.1 shows that when the injected noise is additive, the noised activations at each layer will have *(sub)-Gaussian tails*. By equation (2.9), the accumulated noise $\mathcal{E}_i(\mathbf{x}; \mathbf{w}, \boldsymbol{\epsilon})$ will also have sub-Gaussian tails as the non-noised activations $\mathbf{h}(\mathbf{x})$ are deterministic and do not affect the asymptotic relationships of moments. See Figure A.1 of the Appendix for a demonstration that the activations experience Gaussian-like noise for additive-GNIs. For multiplicative-GNIs the noise at each layer, except the input layer which experiences Gaussian noise, behaves with a sub-Weibull tail. We demonstrate this behaviour in Figure A.2 of the Appendix.

We can now study the properties of the gradient noise induced by the implicit effect by taking the gradients of this forward pass noise.

Kurtosis of The Gradient Noise. We characterise the gradient noise corresponding to $W_{i,l,j}$, the weight that maps from neuron l in layer $i-1$ to neuron j in layer i.

Theorem 3.1. *Consider a feed-forward neural network with an activation function ϕ that obeys the extended envelope property and a cross-entropy or mean-squared-error cost (see Appendix B). The gradient noise from additive-GNIs $\boldsymbol{\epsilon}$, has zero mean and has moments that obey for a pair $(\mathbf{x}, \mathbf{y})$:*

$$\left\|\frac{\partial E_{\mathcal{L}}((\mathbf{x},\mathbf{y}); \mathbf{w}, \boldsymbol{\epsilon})}{\partial W_{i,l,j}}\right\|_m \lesssim m, \quad \text{for any } m \geq 1,$$

where $E_{\mathcal{L}}((\mathbf{x},\mathbf{y}); \mathbf{w}, \boldsymbol{\epsilon})$ is defined in (3.2). For multiplicative GNIs, we have

$$\left\|\frac{\partial E_{\mathcal{L}}((\mathbf{x},\mathbf{y}); \mathbf{w}, \boldsymbol{\epsilon})}{\partial W_{i,l,j}}\right\|_m \lesssim m^{\frac{L+i}{2}}, \quad \text{for any } m \geq 1,$$
$$i = 1, \ldots, L;\ l = 1, \ldots, n_{i-1};\ j = 1, \ldots, n_i.$$

For the additive noise, these bounds infer that the gradient noise at each layer will have *sub-exponential tails*. For the multiplicative case, gradient noise will be *sub-Weibull*, with a tail parameter that increases with i the layer index. Unlike the forward pass, which experienced noise bounded in its tails by a Gaussian, the backward pass experiences noise that is bounded in its tails by heavy-tailed Weibull distributions with tail parameter $r \geq 1$.

Remark 3.1. *The bounds defined by Lemma 3.1 and Theorem 3.1 become an asymptotic equivalence ($\asymp$) in the case of 1-D data and 1-neuron-wide neural networks, i.e. the bounds are maximally tight in this case.*

Our result applies to the gradient for a single pair $(\mathbf{x}, \mathbf{y})$. During SGD, we take the mean gradient across a batch $\mathcal{B}$ of size B. To study the tail distribution of this mean gradient, we restate in a simplified manner Kuchibhotla & Chakrabortty (2018)'s generalisation of the Bernstein inequality for zero-mean sub-Weibull random variables in Theorem 3.2.

Theorem 3.2 (Theorem 3.1 in Kuchibhotla & Chakrabortty (2018))**.** *Let $X_1, \ldots, X_B$ be independent mean-zero sub-Weibull random variables with a shared tail parameter $p \geq 1$. Then, for every $x \geq 0$, we have*

$$\mathbb{P}\left\{\left|(1/B)\sum_{i=1}^{B} X_i\right| \geq x\right\}$$
$$\leq 2\exp\left[-\min\left(\frac{Bx^2}{C\sum_{i=1}^{B}\|X_i\|_{\psi 2}^2}, \frac{Bx^{\frac{1}{\theta}}}{L\max_i \|X_i\|_{\psi \frac{1}{p}}}\right)\right],$$

where $C, L > 0$ are constants that depend on the tail parameter and where

$$\|X\|_{\psi \frac{1}{p}} = \inf\left\{\nu \geq 0 : \mathbb{E}\left[(|X|/\nu)^{\frac{1}{p}} \leq 1\right]\right\} \tag{3.3}$$

is the 'sub-Weibull norm'.

This theorem states that the the tails of the mean of zero-mean i.i.d. sub-Weibull random variables are produced by a single variable, say X_i, with the maximal sub-Weibull norm $\|X_i\|_{\psi \frac{1}{p}} = \inf\{\nu \geq 0 : \mathbb{E}[(|X_i|/\nu)^{\frac{1}{p}} \leq 1]\}$, i.e., the one with the heaviest tails. Assuming gradients are independent across data points, we can use this inequality to bound the tail probability for the sum of our zero-mean gradients. If a single gradient is sufficiently heavy-tailed, then the mean across the batch will also be heavy-tailed.

In Figure 2 we show that the implicit effect gradient, averaged over *the entire dataset* $\mathcal{D}$ (i.e., the largest batch-size possible), is heavy-tailed, unlike the forward pass. To calculate these gradients, we estimate the explicit regulariser in equation (3.2) using Monte Carlo sampling, $\mathbb{E}_{\boldsymbol{\epsilon}}(\Delta\mathcal{L}(\mathcal{D};\mathbf{w},\boldsymbol{\epsilon})) \approx \frac{1}{M}\sum_{m=0}^{M}\Delta\mathcal{L}(\mathcal{D};\mathbf{w},\boldsymbol{\epsilon}_m)$, similarly to (Wei et al., 2020; Camuto et al., 2020). We show these results for multiplicative-GNIs in Figure A.3 of the Appendix. In this setting as well, the backward pass experiences *heavy-tailed* noise from GNIs.

Skewness of The Gradient Noise. In the proof of Theorem 3.1 we decompose $\partial E_{\mathcal{L}}(\cdot)/\partial W_{i,l,j}$ as $(\partial E_{\mathcal{L}}(\cdot)/\partial \widetilde{h}_{i,j})\cdot(\partial \widetilde{h}_{i,j}/\partial W_{i,l,j})$, where $\widetilde{h}_{i,j}$ is the (noised) activation of the j^{th} neuron in the i^{th} layer. Both these derivatives are likely to be skewed due to the asymmetry of the activation functions and their gradients. Ignoring potential correlations between variables, we observe that product of two zero-mean skewed independent variables X and Y is also skewed $|\text{skew}(XY)| = \left|\mathbb{E}\left[X^3\right]\mathbb{E}\left[Y^3\right]\right| > 0$. The skewness of the derivatives in the backward pass will induce *skewed* gradient noise, as seen in Figure 2. Though correlations between gradients could also cause skewness, we show that this is not the case in Appendix C.

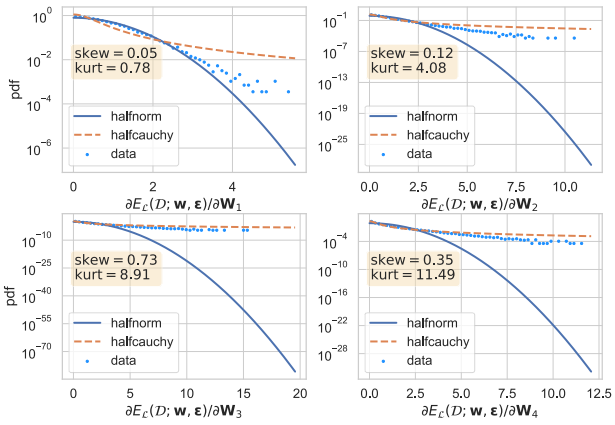

Figure 2. We measure the skewness and kurtosis *at initialisation* of the gradients noise accrued on networks weights during the backward pass for additive-GNIs. The model is a 4-layer-256-unit-wide MLP trained to regress $\lambda(x) = \sum_i \sin(2\pi q_i x + \phi(i))$ with $q_i \in (5, 10, \ldots, 45, 50), x \in \mathbb{R}$. We plot the probability density function (p.d.f.) of positive samples, comparing against half-normal and half-Cauchy distributions. Each blue point represents the mean gradient noise over the entire dataset $\mathcal{D}$ of an individual weight in a layer i. This gradient noise is *skewed and heavy-tailed*, with a p.d.f. that is more Cauchy-like than Gaussian.

4. An SDE Model for SGD with GNIs.

In this section, we will analyse the effects of the skewed and heavy-tailed noise on the SGD dynamics. Recall that the modified loss function by the GNIs is the sum of the explicit regulariser and the original loss over the dataset $\mathcal{D}$,

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left(\widetilde{\mathcal{L}}(\mathcal{D};\mathbf{w},\boldsymbol{\epsilon})\right) = \mathcal{L}(\mathcal{D};\mathbf{w}) + \mathbb{E}_{\boldsymbol{\epsilon}}\left(\Delta\mathcal{L}(\mathcal{D};\mathbf{w},\boldsymbol{\epsilon})\right),$$

and the SGD recursion takes the following form:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta\nabla\,\mathbb{E}_{\boldsymbol{\epsilon}}\left(\widetilde{\mathcal{L}}(\mathcal{D};\mathbf{w}_k,\boldsymbol{\epsilon})\right) + \eta U_k(\mathbf{w}), \quad (4.1)$$

where $U_{k+1}(\mathbf{w})$ is given as follows:

$$\nabla\left[\mathbb{E}_{\boldsymbol{\epsilon}}\left(\widetilde{\mathcal{L}}(\mathcal{D};\mathbf{w}_k,\boldsymbol{\epsilon})\right)\right] - \nabla\left[\mathbb{E}_{\boldsymbol{\epsilon}}\left(\widetilde{\mathcal{L}}(\mathcal{B}_{k+1};\mathbf{w}_k,\boldsymbol{\epsilon})\right) + E_{\mathcal{L}}(\mathcal{B}_{k+1};\mathbf{w}_k,\boldsymbol{\epsilon})\right]. \quad (4.2)$$

To ease the notation, let us denote the modified loss function by $f(\mathbf{w}) := \mathbb{E}_{\boldsymbol{\epsilon}}\left(\widetilde{\mathcal{L}}(\mathcal{D};\mathbf{w},\boldsymbol{\epsilon})\right)$.

Remark 4.1. *Note that when the gradients are computed over the whole dataset $\mathcal{D}$, the gradient noise solely stems from the implicit effect, $U_k(\mathbf{w}) = -\nabla\left[E_{\mathcal{L}}(\mathcal{D};\mathbf{w}_k,\boldsymbol{\epsilon})\right]$.*

Recent studies have proven that heavy-tailed behaviour can already emerge in stochastic optimisation (without GNIs) (Hodgkinson & Mahoney, 2020; Gürbüzbalaban et al., 2021; Gürbüzbalaban & Hu, 2021), which can further result in an overall heavy-tailed behaviour in the gradient noise, as empirically reported in (Şimşekli et al., 2019; Zhang et al., 2020; Zhou et al., 2020). Accordingly, (Şimşekli et al., 2019; Zhou et al., 2020) proposed modelling the gradient noise by using a centred *symmetric* α-stable noise, which then paved the way for modelling the SGD dynamics by using an SDE driven by a symmetric α-stable process (see (2.7)).

We take a similar route for modelling the trajectories of SGD with GNIs; however, due to the skewness arising from the GNIs, the symmetric noise assumption is not appropriate for our purposes. Hence, we propose modelling the skewed gradient noise (4.2) by using an *asymmetric* α-stable noise, which aims at modelling both the heavy-tailed behaviour and the asymmetries at the same time. In particular, we consider an SDE driven by an asymmetric stable process and its Euler discretisation as follows:

$$d\mathbf{w}_t = -\nabla f(\mathbf{w}_t)dt + \varepsilon d\mathbf{L}_t^{\alpha,\theta}, \quad (4.3)$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta_{k+1}\nabla f(\mathbf{w}_k) + \varepsilon\eta_{k+1}^{1/\alpha}\Delta\mathbf{L}_{k+1}^{\alpha,\theta}, \quad (4.4)$$

where $\theta = (\theta_i, 1 \leq i \leq d)$ is the d-dimensional skewness and each coordinate can have its idiosyncratic skewness θ_i. $\mathbf{L}_t^{\alpha,\theta} = (L_t^{\alpha,\theta_1}, \ldots, L_t^{\alpha,\theta_d})$ is a d-dimensional asymmetric α-stable Lévy process with independent components, and ε encapsulates all the scaling parameters. Furthermore, $(\eta_k)_k$ denotes the sequence of step-sizes, which can be taken as constant or decreasing, and finally $(\Delta\mathbf{L}_k^{\alpha,\theta})_k$ is a sequence of i.i.d. random vectors where each component of $\Delta\mathbf{L}_k^{\alpha,\theta}$ is i.i.d. with $\mathcal{S}_\alpha(1,\theta_i,0)$. We then propose the discretised

process (4.4) as a proxy to the original recursion (4.1) and we will directly analyse the theoretical properties of (4.4). Note that our approach strictly extends (Şimşekli et al., 2019), which appears as a special case when $\theta = 0$.

Before deriving our theoretical results, we first verify empirically that the proxy dynamics (4.4) are indeed a good model for representing (4.1). We ascertain that $\Delta\mathbf{L}_k^{\alpha,\theta}$ is sufficiently general in the sense that it can capture the gradient noise induced by the implicit effect even when there is no batching noise (when the batch size approaches the size of the dataset for example, see Remark 4.1). As a first line of evidence, in Figure A.4 of the Appendix we model $\nabla E_{\mathcal{L}}(\cdot)$ as being drawn from a univariate $\mathcal{S}_\alpha$. The equivalent $\mathcal{S}_\alpha$ distributions are skewed ($|\theta| > 0$) and heavy-tailed ($\alpha < 2$), demonstrating that $\mathbf{L}_t^{\alpha,\theta}$ captures the core properties of the implicit effect gradients highlighted in Section 3. To illustrate this more clearly, in Figure 3 we use symmetric sub-Weibull distributions with $r = 0.8, 1, 2$, and fit $\mathcal{S}_\alpha$ and Gaussians ($\mathcal{N}$) using maximum likelihood (MLE) from 10^4 samples. We plot the MLE densities, and clearly MLE $\mathcal{S}_\alpha$ ($\alpha < 2$) better model the tails of the sub-Weibulls than a Gaussian distribution, even for $r = 0.8$ which is close to Gaussian tails ($r = 0.5$). The $\mathcal{S}_\alpha$ modelling of the tails improves as r increases. This further illustrates the appropriateness of our noise model.

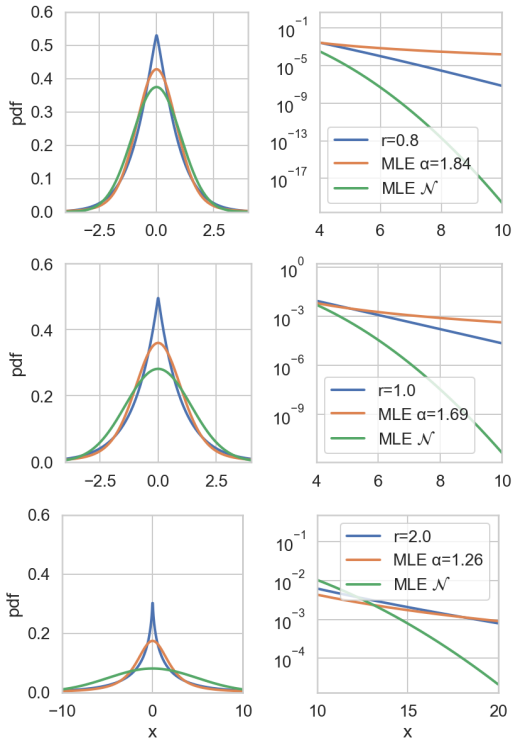

Figure 3. [left col.] sub-Weibull pdfs (r=0.8, 1.0, 2.0), and pdfs of MLE fitted $\mathcal{S}_\alpha$ and $\mathcal{N}$. [right col.] pdf in the tails ($x > 4$ and $x > 10$, note log y-axis).

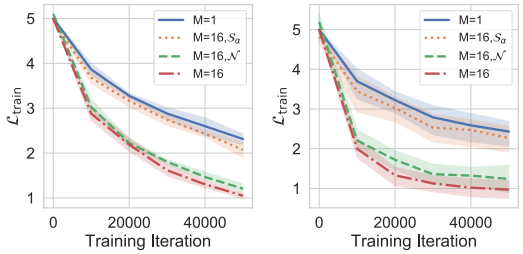

Figure 4. We train the networks in Figure 2 on $\widetilde{\mathcal{L}}_M$ (4.5) for the sinusoidal toy-data with additive GNIs [left] and multiplicative GNIs [right]. We fit univariate $\mathcal{S}_\alpha$ and univariate $\mathcal{N}$ via maximum likelihood (Nolan, 2001) to $\nabla E_{\mathcal{L}}(\cdot)$ for $M = 1$ models *at each training step*. We add draws from these distributions to the gradients of $M = 16$ models and plot the training loss ($\mathcal{L}_{\text{train}}$). Shading is the standard deviation over 5 random seeds.

In Figure 4, we further show that the gradient noise of the implicit effect and gradient noise drawn from an equivalent $\mathcal{S}_\alpha$ distribution will have similar effects on gradient descent. We sample M GNI samples and evaluate:

$$\widetilde{\mathcal{L}}_M(\mathcal{D}; \mathbf{w}, \boldsymbol{\epsilon}) = (1/M) \sum_{m=0}^{M} \widetilde{\mathcal{L}}(\mathcal{D}; \mathbf{w}, \boldsymbol{\epsilon}_m). \quad (4.5)$$

The objective is over the entire dataset such that we eliminate noise from the batching process. M allows us to control the 'degree' to which the implicit effect is marginalised out. $M = 1$ corresponds to the usual training with GNI and larger values of M mimic the effects of marginalising out the implicit effect. We model $\nabla E_{\mathcal{L}}$ for $M = 1$ as being drawn from a univariate $\mathcal{S}_\alpha$ or a univariate normal distribution and estimate distribution parameters using maximum likelihood estimation, as in (Nolan, 2001), at each training iteration. We add draws from the estimated distributions to the gradients of $M = 16$ models to mimic the combined implicit and explicit effects. $M = 16$ models with the added $\mathcal{S}_\alpha$ noise have the same training path as $M = 1$ models, whereas those with Gaussian gradient noise do not. Thus, $\mathcal{S}_\alpha$ distributions are able to faithfully capture the dynamics induced by the implicit effect on gradient descent.

In these same experiments $M = 16$ models outperform $M = 1$ models on training data. We refine this study for a greater range of M values in Figure 5. As M increases, performance of models on training data improves gradually, suggesting that the implicit effect degrades performance. Further, in Figure 4, $M = 16$ models *trained with Gaussian noise* added to gradients outperform $M = 1$ models, suggesting that the heavy-tails and skew of the implicit effect gradients are responsible for this performance degradation. We now study this apparent bias.

Theoretical analysis of implicit bias. Due to their heavy-tailed nature, stable processes have significantly different statistical properties from those of their Brownian counter-

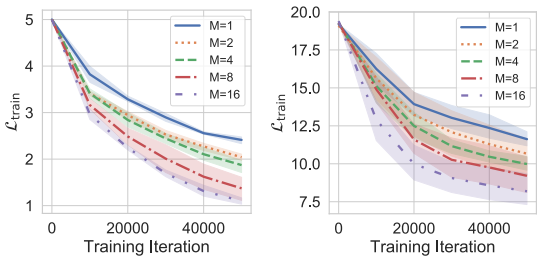

Figure 5. We train the networks in Figure 2 on $\widetilde{\mathcal{L}}_M$ (4.5) for the sinusoidal toy-data with additive GNIs [left] and MNIST with multiplicative GNIs [right]. We plot the training loss ($\mathcal{L}_{\text{train}}$). Shading is the standard deviation over 5 random seeds.

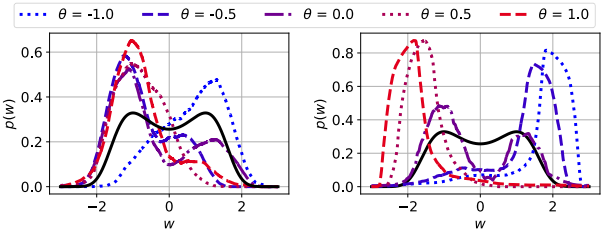

Figure 6. The stationary distributions of (4.4) with $\alpha = 1.9$ (left) and $\alpha = 1.1$ (right). The solid black line represents the density of the Gibbs measure $\exp(-f(w))$ with $f(w) = w^4/4 - w^2/2$.

parts. Their trajectories have a countable number of discontinuities, called *jumps*, whereas Brownian motion is continuous almost everywhere. With these jumps the process can escape from 'narrow' basins and spend more time in 'wider' basins (see Appendix F for the definition of width). Theoretical results demonstrating this have been provided for symmetric stable processes ($\theta = 0$) (Şimşekli et al., 2019). By translating the related metastability results from statistical physics (Imkeller & Pavlyukevich, 2008) to our context, in Appendix F we illustrate that this also holds for SDEs driven by asymmetric stable processes ($\theta \neq 0$). In this sense, the SDE (4.3) is 'biased' towards wider basins.

While driving SGD iterates towards wider minima could be beneficial, we now show that heavy tails can also introduce an undesirable bias and that this bias is magnified by asymmetries. To quantify this bias, we focus on the invariant measure (i.e., the stationary distribution) of the Markov process (4.4) and investigate its modes (i.e., its local maxima), around where the process resides most of the time.

In a statistical physics context, Dybiec et al. (2007) empirically illustrated that the asymmetric stable noise can cause 'shifts', in the sense that the modes of the stationary distribution of (4.4) can shift away from the true local minima of f, which are of our interest as our aim is to minimise f. They further illustrated that such shifts can be surprisingly large when α gets smaller and $|\theta|$ gets larger. We illustrate this outcome by reproducing one of the experiments provided in (Dybiec et al., 2007) in Figure 6. Here, we consider a one-dimensional problem with the quartic potential $f(w) = w^4/4 - w^2/2$, and simulate (4.4) for 10K iterations with constant step-size $\eta_k = 0.001$ and $\varepsilon = 1$. By using the generated iterates, we estimate the density of the invariant measure of (4.4) by using the kernel density estimator provided in scikit-learn (Pedregosa et al., 2011), for different values of α and θ. When α is larger (left), the heavy-tails cause a shift in the modes of the invariant measure, where these shifts become slightly larger with increasing asymmetries ($|\theta| > 0$). When the tails are heavier (right), we observe a much stronger interaction between α and θ, and observe *drastic* shifts as the asymmetry is increased.

From an optimisation perspective, these results are rather unsettling as they imply that SGD might spend most of its time in regions that are arbitrarily far from the local minima of the objective function f, since the high probability regions of its stationary distribution might be shifted away from the local minima of interest. In the symmetric case ($\theta = 0$), this observation has been formally proven in (Sliusarenko et al., 2013) when f is chosen as the one dimensional quartic potential of Figure 6 and when $\alpha = 1$. A direct quantification of such shifts is non-trivial; and in the presence of asymmetries ($\theta \neq 0$), even further difficulties emerge.

In a recent study Şimşekli et al. (2020) focused on eliminating the undesired bias introduced by *symmetric* stable noise in SGD with momentum (Qian, 1999), and proposed an indirect way to ensure that modes of the stationary distribution exactly match the objective function's local minima. They developed a 'modified' SDE whose invariant distribution can be proven to be the *Gibbs measure*, denoted by $\pi(d\mathbf{w})$, which is a probability measure that has a density proportional to $\exp(-Cf(\mathbf{w}))$ for some $C > 0$. Clearly, all the local maxima of this density coincide with the local minima of the function f; hence, their approach eliminates the possibility of a shift in the modes by imposing a stronger condition which controls the entire invariant distribution.

By following a similar approach, we will bound the gap between the invariant measure of (4.4) and the Gibbs measure in terms of the tail index α and the skewness θ, using it as a quantification of the bias induced by the asymmetric heavy-tailed noise. In particular, for any sufficiently regular test function g, we consider its expectation under the Gibbs measure $\nu(g) := \int g(\mathbf{w})\pi(d\mathbf{w})$, and its sample average computed over (4.4), i.e., $\nu_N(g) := \frac{1}{H_N}\sum_{k=1}^{N} \eta_k g(\mathbf{w}_k)$, where $H_N = \sum_{k=1}^{N} \eta_k$. We then bound the *weak error*: $|\nu(g) - \lim_{N\to\infty} \nu_N(g)|$, whose convergence to zero is sufficient for ensuring the modes do not shift.

We derive this bound for our case in three steps: (i) We first link the discrete-time process (4.4) to its continuous-time limit (4.3) by directly using the results of (Panloup, 2008). (ii) We then design a modified SDE that has the unique

invariant measure as the Gibbs measure, with all the modes matching those of the loss function (Theorem 4.1). (iii) Finally, we show that the SDE (4.3) is a poor numerical approximation to the modified SDE, and we develop an upper-bound for the approximation error (Theorem 4.2).

To address (ii), we introduce a modification to (4.3), coined *asymmetric fractional Langevin dynamics*:

$$d\mathbf{w}_t = b(\mathbf{w}_{t-}, \alpha, \theta)dt + \varepsilon d\mathbf{L}_t^{\alpha,\theta}\,. \tag{4.6}$$

where, the drift function $b(\mathbf{w}, \alpha, \theta) := ((b(\mathbf{w}, \alpha, \theta))_i, 1 \leq i \leq d)$ is defined as follows:

$$(b(\mathbf{w}, \alpha, \theta))_i = \frac{\varepsilon^\alpha}{\varphi(\mathbf{w})}\mathcal{D}_{w_i}^{\alpha-2,-\theta_i}(\partial_{w_i}\varphi(\mathbf{w})), \tag{4.7}$$

where $\theta_i \in (-1, 1)$, $1 \leq i \leq d$, $1 < \alpha < 2$, and $\varphi(\mathbf{w}) := e^{-\varepsilon^{-\alpha} f(\mathbf{w})}$. Here, the operator $\mathcal{D}^{\alpha-2,-\theta_i}$ denotes a *Riesz-Feller type fractional derivative* (Gorenflo & Mainardi, 1998; Mainardi et al., 2001) whose exact (and rather complicated) definition is not essential in our problematic, and is given in Appendix E in order to avoid obscuring the main results. The next theorem states that the SDE (4.6) targets the Gibbs measure.

Theorem 4.1. *The Gibbs measure* $\pi(d\mathbf{w}) \propto \exp(-\varepsilon^{-\alpha} f(\mathbf{w}))d\mathbf{w}$ *is an invariant distribution of* (4.6). *If* $b(\mathbf{w}, \alpha, \theta)$ *is Lipschitz continuous in* $\mathbf{w}$, *then* $\pi(d\mathbf{w})$ *is the unique invariant distribution of* (4.6).

This theorem states that the use of the modified drift b in place of $-\nabla f$, prevents any potential shifts in the modes of the invariant measure.

The fractional derivative in (4.7) is a non-local operator that requires the knowledge of the full function, and does not admit a closed-form expression. In the next step, we develop an approximation scheme for the drift b in (4.6), and show that the gradient $-\nabla f$ in (4.4) appears as a special case of this scheme. To simplify notation, we consider the one-dimensional case ($d = 1$); however, our results can be easily extended to multivariate settings by applying the same approach to each coordinate. Hence, in the general case the bounds will scale linearly with d. We define the following approximation for b:

$$b_{h,K}(w, \alpha, \theta) := \frac{\varepsilon^\alpha}{\varphi(w)}\Delta_{h,K}^{\alpha-2,-\theta}(\partial_w\varphi(w)), \tag{4.8}$$

where, for an arbitrary function ψ, we have

$$\Delta_{h,K}^{\gamma,-\theta}\psi(w) := \frac{c_\gamma}{h^\gamma}\sum_{k=-K}^{K}(1 + \theta\,\mathrm{sgn}(k))\tilde{g}_{\gamma,k}\psi(w - kh).$$

Here, sgn denotes the sign function, $h > 0$, $K \in \mathbb{N} \cup \{0\}$, $c_\gamma := 1/(2\cos(\gamma\pi/2))$, and $\tilde{g}_{\gamma,k} := (-1)^k\Gamma(-\gamma + k)/\Gamma(k + 1)\Gamma(-\gamma)$. This approximation is designed in the way that we recover the original drift b as $h \to 0$ and $K \to \infty$ for sufficiently regular φ. It is clear that when we set $K = 0$ and $h = h_0 := [2\varepsilon^{-\alpha}\cos((\alpha - 2)\pi/2)]^{1/(2-\alpha)}$, we have $b_{h,K}(w, \alpha, \theta) = -\partial_w f(w)$. In the multidimensional case, where we apply this approximation to each coordinate, the same choice of K and h gives us the original gradient $-\nabla f$; hence, we fall back to the original recursion (4.4). By considering the recursion with this approximate drift

$$\tilde{\mathbf{w}}_{n+1} = \tilde{\mathbf{w}}_n + \eta_{n+1}b_{h,K}(\tilde{\mathbf{w}}_n, \alpha, \theta) + \varepsilon\eta_{n+1}^{1/\alpha}\Delta\mathbf{L}_{n+1}^{\alpha,\theta},$$

and the corresponding sample averages $\tilde{\nu}_N(g) := \frac{1}{H_N}\sum_{k=1}^{N}\eta_k g(\tilde{\mathbf{w}}_k)$[2], we are ready to state our error bound. We believe this result is interesting on its own, and would be of further interest in statistical physics and applied probability. To avoid obscuring the result, we state the required assumptions in the Appendix, which mainly require decreasing step-size and ergodicity.

Theorem 4.2. *Let* $\gamma := \alpha - 2 \in (-1, 0)$. *Suppose that the assumptions stated in the Appendix hold. Then, the following bound holds almost surely:*

$$\begin{aligned}&|\nu(g) - \lim_{N\to\infty}\tilde{\nu}_N(g)| \\ &\leq \frac{\tilde{C}}{4\pi(|\gamma| + 2)}\left[|\theta||\gamma| + |\tan(\gamma\pi/2)|\,|\gamma|\right]h \\ &\quad + ((1 + \theta)C_0' + (1 - \theta)C_0'')\,\frac{1}{hK} + \mathcal{O}(h^2)\,,\end{aligned} \tag{4.9}$$

where $\tilde{C}$, C_0', $C_0'' > 0$ *are constants.*

We note our result extends the case $\alpha = 2, \theta = 0$ in Durmus & Moulines (2017) and the case $\alpha \neq 2, \theta = 0$ in Şimşekli (2017); whereas we cover the case $\alpha \neq 2$, $\theta \in (-1, 1)$. The right-hand-side of (4.9) contains two main terms. The second term shows that the error increases linearly with decreasing K, indicating that the error can be *arbitrarily large* when $K = 0$, and the gap cannot be controlled without imposing further assumptions on f.

More interestingly, even when K goes to infinity (i.e., the second term vanishes), the first term stays unaffected. Note that for large enough ε, h_0 increases as $\alpha \in (1, 2)$ decreases. In this regime, the first term indicates that the error increases with decreasing α, and an additional error term appears whenever $\theta \neq 0$, which is further amplified with the heaviness of the tails (measured by $|\gamma|$). This outcome provides a theoretical justification to the empirical observations stated in Figures 6 and 8.

4.1. Further Experiments

We have already ascertained that the bias implied by Theorem 4.2 has a visible impact on training performance in

[2]Note that, with the choice of $K = 0$ and $h = h_0$, $\tilde{\nu}_N(g)$ reduces to the original sample average $\nu_N(g)$.

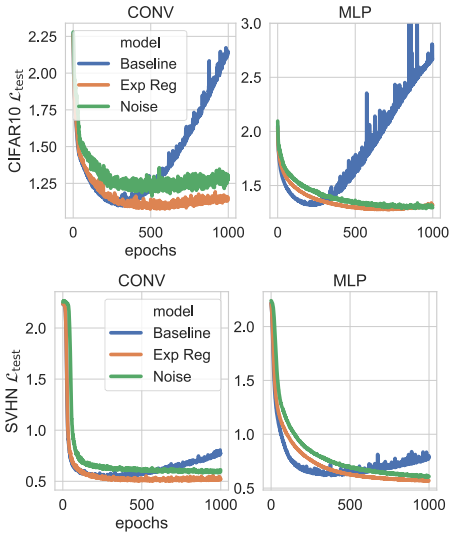

Figure 7. We show the test-set loss for SVHN [bottom] and CIFAR10 [top], for 2-layer convolutional (CONV) and 4-layer MLPs with 512 units per layer trained with the explicit regulariser approximation $R(\cdot)$ of Camuto et al. (2020) (Exp Reg), with additive-GNIs ($\sigma^2 = 0.1$) (Noise), and no regularisation (Baseline).

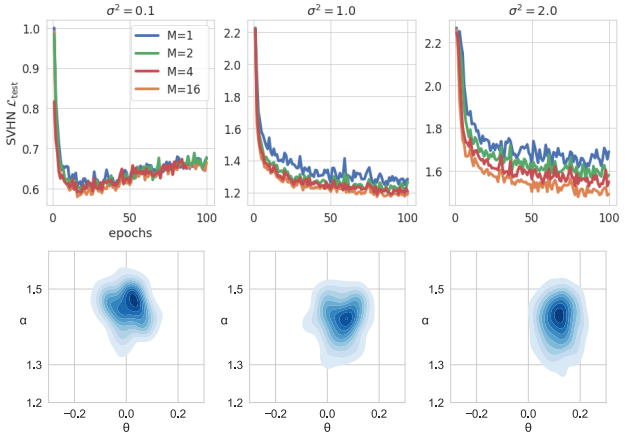

Figure 8. [first row] We train 2-dense-layer-256-unit-per-layer ELU networks on the objective $\frac{1}{M}\sum_{m=0}^{M}\widetilde{\mathcal{L}}(\mathcal{B};\mathbf{w},\boldsymbol{\epsilon}_m)$ with a cross-entropy loss (see Appendix B) for SVHN. We use *multiplicative* noise of variance σ^2 and batch size of 512. We plot the test-set loss ($\mathcal{L}_{\text{test}}$). [second row] We fit univariate $\mathcal{S}_\alpha$ via maximum likelihood (Nolan, 2001) to $\nabla E_{\mathcal{L}}(\cdot)$ and show KDE plots of parameters' estimates.

Figures 4 and 5. We have also already shown that the asymmetry and heavy-tails of the implicit-effect gradient noise are responsible for this performance degradation in Figure 4: models trained with Gaussian noise on gradients outperform models trained with $\mathcal{S}_\alpha$ gradient noise, and those trained with the implicit effect, *on training data without batching*. We corroborate these findings with experiments *with mini-batching and results on test data*. In Figure 7 we use the approximation of the explicit regulariser $R(\mathcal{B};\mathbf{w})$ derived by Camuto et al. (2020) for computational efficiency. Convolutional networks trained with R consistently outperform those trained with GNIs and mini-batching on *held-out data*, supporting that the implicit effect degrades performance. In Figure 8, we sample M multiplicative-GNI samples and marginalise out the implicit effect as before. We model the gradients of the implicit effect, $\nabla E_{\mathcal{L}}(\cdot)$, as an $\mathcal{S}_\alpha$ distribution. Empirically, we found that when increasing the variance (σ^2) of the injected noise, the gradient noise $\nabla E_{\mathcal{L}}(\cdot)$ becomes increasingly heavy-tailed and skewed, i.e. α decreases and $|\theta|$ increases, and in tandem larger M models begin to outperform smaller M models on *held-out data*, when trained with mini-batches. These results support that GNIs induce bias in SGD because of the asymmetric heavy-tailed noise they induce on gradient updates.

5. Conclusion

Our work lays the foundations for the study of regularisation methods from the perspective of SDEs. We have shown that Gaussian Noise Injections (GNIs), though they inject Gaussian noise in the forward pass, induce asymmetric heavy-tailed noise on gradient updates by way of the implicit effect. By modelling the overall induced noise using an asymmetric α-stable noise, we demonstrate that the stationary distribution of this process gets arbitrarily distant from the so-called Gibbs measure, whose modes exactly match the local minima of the loss function, shedding light on why neural networks trained with GNIs underperform networks trained solely with the explicit effect. Given the deleterious effects of asymmetric gradient noise on gradient descent, extensions of this work could focus on methods that symmetrise gradient noise, stemming from batching or noise injections, so as to limit these negative effects.

Acknowledgements

This research was directly funded by the Alan Turing Institute under Engineering and Physical Sciences Research Council (EPSRC) grant EP/N510129/1. Alexander Camuto was supported by an EPSRC Studentship. Xiaoyu Wang and Lingjiong Zhu are partially supported by the grant NSF DMS-2053454 from the National Science Foundation. Lingjiong Zhu is also grateful to the partial support from a Simons Foundation Collaboration Grant. Mert Gürbüzbalaban's research is supported in part by the grants Office of Naval Research Award Number N00014-21-1-2244, National Science Foundation (NSF) CCF-1814888, NSF DMS-1723085, NSF DMS-2053485. Umut Şimşekli's research is partly 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).

6. References

Agapiou, S., Stuart, A. M., and Zhang, Y. X. Bayesian posterior contraction rates for linear severely ill-posed inverse problems. *Journal of Inverse and Ill-Posed Problems*, 22 (3):297–321, 2014.

Bishop, C. M. Training with Noise is Equivalent to Tikhonov Regularization. *Neural Computation*, 7(1):108–116, 1995.

Camuto, A., Willetts, M., Şimşekli, U., Roberts, S., and Holmes, C. Explicit Regularisation in Gaussian Noise Injections. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Cohen, J., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:2323–2356, 2019.

Dieng, A. B., Ranganath, R., Altosaar, J., and Blei, D. M. Noisin: Unbiased regularization for recurrent neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 1252–1261, 2018.

Duan, J. *An Introduction to Stochastic Dynamics*. Cambridge University Press, New York, 2015.

Durmus, A. and Moulines, E. Nonasymptotic convergence analysis for the unadjusted Langevin algorithm. *Annals of Applied Probability*, 27(3):1551–1587, 2017.

Dybiec, B., Gudowska-Nowak, E., and Sokolov, I. Stationary states in Langevin dynamics under asymmetric Lévy noises. *Physical Review E*, 76(4):041122, 2007.

Fiche, A., Cexus, J. C., Martin, A., and Khenchaf, A. Features modeling with an α-stable distribution: Application to pattern recognition based on continuous belief functions. *Information Fusion*, 14(4):504–520, 2013.

Gao, X., Gürbüzbalaban, M., and Zhu, L. Global Convergence of Stochastic Gradient Hamiltonian Monte Carlo for Non-Convex Stochastic Optimization: Non-Asymptotic Performance Bounds and Momentum-Based Acceleration. *arXiv:1809.04618*, 2018.

Gao, X., Gürbüzbalaban, M., and Zhu, L. Breaking reversibility accelerates Langevin dynamics for global non-convex optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Gnedenko, B. V. and Kolmogorov, A. *Limit Distributions for Sums of Independent Random Variables*. Addison-Wesley, Cambridge, MA, 1954. Translated by Kai Lai Chung.

Gorenflo, R. and Mainardi, F. Random walk models for space-fractional diffusion processes. *Fractional Calculus & Applied Analysis*, 1:167–191, 1998.

Gürbüzbalaban, M. and Hu, Y. Fractional moment-preserving initialization schemes for training fully-connected neural networks. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, 2021.

Gürbüzbalaban, M., Şimşekli, U., and Zhu, L. The heavy-tail phenomenon in SGD. In *International Conference on Machine Learning*, 2021.

Hodgkinson, L. and Mahoney, M. W. Multiplicative noise and heavy tails in stochastic optimization. *arXiv preprint arXiv:2006.06293*, 2020.

Imkeller, P. and Pavlyukevich, I. Metastable behavior of small noise Lévy-driven diffusions. *ESAIM: Probability and Statistics*, 12:412–437, 2008.

Jastrzębski, S., Kenton, Z., Arpit, D., Ballas, N., Fischer, A., Bengio, Y., and Storkey, A. Three Factors Influencing Minima in SGD. *arXiv:1711.04623*, 2017.

Kingma, D. P., Salimans, T., and Welling, M. Variational dropout and the local reparameterization trick. In *Advances in Neural Information Processing Systems*, volume 2015-Janua, pp. 2575–2583, 2015.

Kroneburg, M. The binomial coefficient for negative arguments. *arXiv preprint arXiv:1105.3689*, 2011.

Kuchibhotla, A. K. and Chakrabortty, A. Moving beyond sub-Gaussianity in high dimensional statistics: Applications in covariance estimation and linear regression. *arXiv:1804.02605*, 2018.

Li, Q., Tai, C., and E, W. Stochastic modified equations and adaptive stochastic gradient algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 2101–2110, 06–11 Aug 2017.

Mainardi, F., Luchko, Y., and Pagnini, G. The fundamental solution of the space-time fractional diffusion equation. *Fractional Calculus & Applied Analysis*, 4:153–192, 2001.

Mandt, S., Hoffman, M. D., and Blei, D. M. A variational analysis of stochastic gradient algorithms. *33rd International Conference on Machine Learning, ICML 2016*, 1: 555–566, 2016.

Meerschaert, M. and Tadjeran, C. Finite difference approximation for fractional advection-dispersion flow equations. *Journal of Computational and Applied Mathematics*, 172: 65–77, 2004.

Nadarajah, S. and Pogány, T. K. On the distribution of the

product of correlated normal random variables. *Comptes Rendus Mathematique*, 354(2):201–204, 2016.

Nolan, J. P. Maximum likelihood estimation and diagnostics for stable distributions. In Barndorff-Nielsen, O. E., Resnick, S. I., and Mikosch, T. (eds.), *Lévy Processes: Theory and Applications*, pp. 379–400. Birkhäuser Boston, Boston, MA, 2001.

Oliveira, A., Oliveira, T. A., and Seijas-Macias, A. Skewness into the product of two normally distributed variables and the risk consequences. *Revstat Statistical Journal*, 14 (2):119–138, 2016.

Ortigueira, M. D. Riesz potential operators and inverses via fractional centred derivatives. *International Journal of Mathematics and Mathematical Sciences*, 2006(Article ID 48391):1–12, 2006.

Ortiguera, M. D. Fractional central differences and derivatives. *IFAC Proceedings Volumes*, 39(11):58–63, 2006b.

Panloup, F. Recursive computation of the invariant measure of a stochastic differential equation driven by a Lévy process. *Annals of Applied Probability*, 18(2):379–426, 2008.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Poole, B., Sohl-Dickstein, J., and Ganguli, S. Analyzing noise in autoencoders and deep networks. *arXiv:1406.1831*, 2014.

Qian, N. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, 1999.

Raginsky, M., Rakhlin, A., and Telgarsky, M. Non-convex learning via stochastic gradient Langevin dynamics: a nonasymptotic analysis. In *Conference on Learning Theory*, pp. 1674–1703, 2017.

Robbins, H. and Monro, S. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3): 400–407, 1951.

Roberts, G. O. and Stramer, O. Langevin diffusions and Metropolis-Hastings algorithms. *Methodology and Computing in Applied Probability*, 4(4):337–357, 2002.

Ruder, S. An overview of gradient descent optimization algorithms. *arXiv:1609.04747*, 2016.

Samorodnitsky, G. and Taqqu, M. S. *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*. Chapman & Hall, New York, 1994.

Sarafrazi, K. and Yazdi, M. Skewed alpha-stable distribution for natural texture modeling and segmentation in contourlet domain. *Eurasip Journal on Image and Video Processing*, 2019(1):1–12, 2019.

Schertzer, D., Larchevêque, M., Duan, J., Yanovsky, V., and Lovejoy, S. Fractional Fokker-Planck equation for nonlinear stochastic differential equations driven by non-Gaussian Lévy stable noises. *Journal of Mathematical Physics*, 42(1):200–212, 2001.

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, 2017.

Şimşekli, U., Sagun, L., and Gürbüzbalaban, M. A tail-index analysis of stochastic gradient noise in deep neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 5827–5837, 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 *Proceedings of the 37th International Conference on Machine Learning*, pp. 8970–8980, 2020.

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:1912.00018*, 2019.

Sliusarenko, O. Y., Surkov, D., Gonchar, V. Y., and Chechkin, A. V. Stationary states in bistable system driven by Lévy noise. *The European Physical Journal Special Topics*, 216(1):133–138, 2013.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

Tian, W., Zhou, H., and Deng, W. A class of second order difference approximations for solving space fractional diffusion equations. *Mathematics of Computation*, 84 (294):1703–1727, 2015.

Vershynin, R. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

Vladimirova, M., Verbeek, J., Mesejo, P., and Arbel, J. Understanding priors in Bayesian neural networks at the unit level. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:11248–11257, 2019.

Vladimirova, M., Girard, S., Nguyen, H., and Arbel, J. Sub-Weibull distributions: generalizing sub-Gaussian and

sub-Exponential properties to heavier-tailed distributions. *Stat*, 9(1):1–10, 2020.

Webb, A. R. Functional Approximation by Feed-Forward Networks: A Least-Squares Approach to Generalization. *IEEE Transactions on Neural Networks*, 5(3):363–371, 1994.

Wei, C., Kakade, S., and Ma, T. The Implicit and Explicit Regularization Effects of Dropout. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 10181–10192, 2020.

Welling, M. and Teh, Y. W. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In *Proceedings of the 28th International Conference on Machine Learning*, ICML'11, pp. 681–688, Madison, WI, USA, 2011. Omnipress.

Zhang, C., Recht, B., Bengio, S., Hardt, M., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR 2017*, 2017.

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 (NeurIPS)*, volume 33, 2020.

Zhou, P., Feng, J., Ma, C., Xiong, C., Hoi, S., and E, W. Towards theoretically understanding why SGD generalizes better than ADAM in deep learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.