FLATTER, FASTER: SCALING MOMENTUM FOR OPTIMAL SPEEDUP OF SGD

Anonymous authors
Paper under double-blind review

ABSTRACT

Commonly used optimization algorithms often show a trade-off between good generalization and fast training times. For instance, stochastic gradient descent (SGD) tends to have good generalization; however, adaptive gradient methods have faster training times. Momentum can accelerate training with SGD, but so far there has been no principled way to select the momentum hyperparameter. Here we study training dynamics arising from the interplay between SGD with label noise and momentum in the training of overparametrized neural networks. We find that scaling the momentum hyperparameter $1-\beta$ with the learning rate to the power of $2/3$ maximally accelerates training, without sacrificing generalization. To analytically derive this result we develop an architecture-independent framework, where the main assumption is the existence of a degenerate manifold of global minimizers, as is natural in overparametrized models. Training dynamics display the emergence of two characteristic timescales that are well-separated for generic values of the hyperparameters. The maximum acceleration of training is reached when these two timescales meet, which in turn determines the scaling limit we propose. Our experiments in matrix-sensing, a 6-layer MLP on FashionMNIST and ResNet-18 on CIFAR10 validate this scaling for the time to convergence, and additionally for the momentum hyperparameter which maximizes generalization.

1 INTRODUCTION

The modern paradigm for optimization of deep neural networks has engineers working with vastly overparametrized models and training to near perfect accuracy (Zhang et al., 2017). In this setting, a model will typically have not just isolated minima in parameter space, but a continuous set of minimizers, not all of which generalize well. Liu et al. (2020) demonstrate that depending on parameter initialization and hyperparameters, stochastic gradient descent (SGD) is capable of finding minima with wildly different test accuracies. Thus, the power of a particular optimization method lies in its ability to select a minimum that generalizes amongst this vast set. In other words, good generalization relies on the *implicit bias or regularization* of an optimization algorithm.

Training deep neural networks with SGD is known to favor generalization, with a preference for flatter minima that ensure better generalization (Keskar et al., 2017; Wu et al., 2018; Xie et al., 2020; Hochreiter & Schmidhuber, 1997). Recent studies, like Blanc et al. (2020) and Li et al. (2022), showed rigorously that SGD with algorithmic label noise targets flatter minima, but this process is slow compared to the initial convergence.

With the recent explosion in size of both models and datasets, training time has become an important consideration in addition to asymptotic generalization error. In this context, adaptive gradient methods such as Adam (Kingma & Ba, 2015) are unilaterally preferred over variants of SGD, even though they often yield worse generalization errors in practical settings (Keskar & Socher, 2017; Wilson et al., 2017), though extensive hyperparameter tuning (Choi et al., 2019) or scheduling (Xie et al., 2022) can potentially obviate this problem.

These two constraints motivate a careful analysis of how momentum accelerates SGD. Classic work on acceleration methods, which we refer to generally as momentum, have found a provable benefit in the deterministic setting, where gradient updates have no error. However, rigorous guarantees have been harder to find in the stochastic setting, and remain limited by strict conditions on the noise (Polyak, 1987; Kidambi et al., 2018) or model class and dataset structure Lee et al. (2022).

In this work, we show that there exists a scaling limit for SGD with momentum (SGDM) which provably increases the rate of convergence.

Notation. In what follows, we denote by C^n, for $n = 0, 1, \dots$ the set of functions with continuous n^{th} derivatives. For any function f, $\partial f[u]$ and $\partial^2 f[u, v]$ will denote directional first and second derivatives along directions defined by vectors $u, v \in \mathbb{R}^D$. We may occasionally also write $\partial^2 f[\Sigma] = \sum_{i,j=1}^{D} \partial^2 f[e_i, e_j]\Sigma_{ij}$. Given a submanifold $\Gamma \subset \mathbb{R}^D$ and $w \in \Gamma$, we denote by $T_w\Gamma$ the tangent space to Γ in w, and by $P_L(w)$ the projector onto $T_w\Gamma$ (we will often omit the dependence on w and simply write P_L), and $P_T = \text{Id} - P_L$. Given a matrix $H \in \mathbb{R}^D \times \mathbb{R}^D$, we will denote by $H^\top$ the transpose, and $H^\dagger$ the pseudoinverse of H. We will write the expectation of a quantity O over noise as $\langle O \rangle$.

1.1 Qualitative Explanation for Optimal Momentum-Based Speedup

Deep overparametrized neural networks typically posses a manifold of parametrizations with zero training error. Because the gradients of the loss function vanish along this manifold, the dynamics of the weights is completely frozen under gradient descent. However, as appreciated by Wei & Schwab (2019) and Blanc et al. (2020), noise can generate an average drift of the weights along the manifold. In particular, SGD noise can drive the weights to a lower-curvature region which, heuristically, explains the good generalization properties of SGD. Separately, it is well-known that adding momentum typically leads to acceleration in training Sutskever et al. (2013). Below, we will see that there is a nontrivial interplay between the drift induced by noise and momentum, and find that acceleration along the zero-loss manifold is maximized by a particular hyperparameter choice in the limit of small learning rate. In this section, through heuristic arguments, we present the main intuition leading to this prediction.

We use the following formulation to model momentum SGD with label noise:

$$\begin{aligned} \pi_{k+1} &= \beta \pi_k - \nabla L(w_k) + \epsilon \sigma(w_k)\xi_k \\ w_{k+1} &= w_k + \eta \pi_{k+1} \,, \end{aligned} \tag{1}$$

where η is the learning rate, β is the (heavy-ball) momentum hyperparameter (as introduced by Polyak (1964)), $w \in \mathbb{R}^D$ denotes the weights and $\pi \in \mathbb{R}^D$ denotes momentum (also called the auxiliary variable). $L : \mathbb{R}^D \to \mathbb{R}$ is the training loss, and $\sigma : \mathbb{R}^D \to \mathbb{R}^{D \times r}$ is the noise function, whose dependence on the weights allows to model the gradient noise due to SGD and label noise. Specifically this admits modeling phenomena such as automatic variance reduction (Liu & Belkin, 2020), and expected smoothness, satisfying only very general assumptions such as those developed by Khaled & Richtárik (2020). Finally, $\xi_k \in \mathbb{R}^r$ is sampled i.i.d. at every timestep k from a distribution with zero mean and unit variance, and $\epsilon > 0$.

We now give a qualitative explanation for the origin of our proposed scaling limit. Let us assume that the model has been trained for a sufficient number of epochs so that the weights w_k lie in the vicinity of the zero-loss manifold. Even though gradients are very small near this manifold, the noise induces transverse fluctuations that drive a net longitudinal drift along the manifold. This is represented in a diagram in Fig. 2. Importantly the drift leads towards flatter points of the manifold. This attribute is beneficial, as flatter minima are known to typically possess better generalization Keskar et al. (2017); Dziugaite & Roy (2017); Jiang et al. (2019). How efficient is this noise-induced drift? Intuitively, its speed is controlled by the size of fluctuations that are transverse to the zero-loss valley: the smaller the fluctuations, the slower the drift. The size of these fluctuations is generally relatively small as it is controlled by noise and learning rate, resulting in a slow drift and generalization. Our message is that momentum can significantly accelerate this process. Indeed, as $\beta \to 1$, the past gradients are remembered for a longer time in the training dynamics, making the average direction of the gradient more persistent and, in turn, has the effect of increasing the size of transverse fluctuations. Additionally because momentum averages gradients, it will have a stronger effect on the longitudinal component, whose direction is constant in time, compared to the transverse component whose direction changes with time. These mechanisms accelerate drift along the manifold. On the other hand, as we increase β, again due to the long-time memory of the gradients, the decay of the model weights towards the zero-loss manifold is progressively slower. This may slow down the overall convergence for β very near 1 and thus the time it takes to find a generalizing minimum. Putting together these considerations, we expect a non-monotonic dependence on β of the time it takes to generalize. We will show that the optimal choice of β has a simple dependence on the learning rate.

This is particularly useful in the late stage of training, where the learning rate is often taken to decrease but β is kept fixed. Our proposal is thus that β should instead be tied to the learning rate schedule.

1.2 Heuristic Derivation of the Scaling Law

We will now present a heuristic description of the drift dynamics that is induced by the noise along a manifold of minimizers $\Gamma = \{w : L(w) = 0\} \subseteq \mathbb{R}^D$, in the limit $\epsilon \to 0$. In practice, this limit corresponds to choosing small strength of label noise and large minibatch size, so that weights remain close enough to the zero-loss landscape. Let us assume that the weights at initial time w_0 are already close to Γ, and $\pi_0 = 0$ (note that we assume these initial conditions for simplicity: our main result Theorem 3.4 applies upon the more general Assumption 3.2). Because of this condition, the gradients of L at w_0 are very small, and so only fluctuations transverse to the manifold will generate systematic drifts. Denoting by $\delta w_k = w_0 - w_k$ the displacement of the weights after k timesteps, let us Taylor expand the first equation in (1) to get $\pi_{k+1} = \beta\pi_k - \nabla^2 L(w_0)[\delta w_k] + \epsilon\sigma(w_0)\xi_k$. By construction, the Hessian $\nabla^2 L(w_0)$ vanishes along the directions tangent to Γ, while in the transverse direction we have an Ornstein-Uhlenbeck (OU) process. The number of time steps it takes to this process to relax to the stationary state, i.e. the inverse of the smallest nonzero eigenvalue of the damping term, is $\tau_1 = \Theta\left(1/(1-\beta)\right)$ as $\beta \to 1$, which can be anticipated from the first equation in (1) since $\pi_{k+1} - \pi_k \sim -(1-\beta)\pi_k$. After this time, the variance of this linearized OU process becomes time-independent, and can be estimated to be (see Appendix H): $\langle(\delta w_k^T)^\top \delta w_k^T\rangle = \Theta\left(\epsilon^2\eta/(1-\beta)\right)$. Now, the displacements in the longitudinal directions, δw_k^L, are driven by the third order term in the Taylor expansion of $L(w_0 + \delta w_k)$, i.e. $\partial^2(\nabla L)[\delta w_k, \delta w_k]$. Taking the noise average, the tangential gradient of the loss is then $P_L\langle\nabla L(w_j)\rangle = \partial^2(P_L\nabla L)[\langle(\delta w_j)^\top \delta w_j\rangle] = \Theta(\epsilon^2\eta/(1-\beta))$, where P_L is the projector onto the tangent space to Γ. The expectation value of momentum, upon applying the longitudinal projector, is then $P_L\langle\pi_{k+1}\rangle = -P_L\sum_{j=0}^{k}\beta^{k-j}\langle\nabla L(w_j)\rangle$. Assuming the gradient of the loss to be approximately constant, we then find $P_L\langle\pi_{k+1}\rangle = \Theta(\epsilon^2\eta/(1-\beta))\sum_{j=0}^{k}\beta^{k-j} = \Theta(\epsilon^2\eta/(1-\beta)^2)$. The variance of the transverse displacements therefore activates longitudinal motion. Using the second equation in (1), we see that $P_L(\langle\delta w_{k+1}\rangle - \langle\delta w_k\rangle) = \eta P_L\langle\pi_{k+1}\rangle$. Define τ_2 to be the number of time steps it takes so that the displacement of w_k along Γ becomes a finite number as we take $\epsilon \to 0$ first, and $\eta \to 0$ afterward. From the above considerations, $P_L(\langle\delta w_{\tau_2}\rangle - \langle\delta w_0\rangle) = \tau_2\Theta(\epsilon^2\eta^2/(1-\beta)^2) = C$, where C is a finite constant. We then find that $\tau_2 = \Theta\left((1-\beta)^2/\epsilon^2\eta^2\right)$. For a rigorous definition of τ_2, see Appendix B.

The key observation now is that, near the zero-loss valley, the convergence time is controlled by the largest timescale among τ_1 and τ_2. Therefore, the optimal speedup of training is achieved when these two timescales intersect $\tau_1 = \tau_2$, which happens for $1 - \beta = C\eta^{2/3}$, for some constant C that is independent of the learning rate. Motivated by this relationship, we will occasionally parameterize β as

$$\beta = 1 - C\eta^\gamma\,, \tag{2}$$

and study the dependence of τ_1 and τ_2 as a function of γ to derive theoretically and confirm empirically our proposed scaling.

1.3 Limit Drift-Diffusion

We now describe the rationale for obtaining the limiting drift-diffusion on the zero-loss manifold, for a process of the form (1) which foreshadows the rigorous results presented in Sec. 3. As discussed above, the motion along the manifold is slow, as it takes $\Theta(\epsilon^{-2})$ time steps to have a finite amount of longitudinal drift. We want to extract this slow longitudinal sector of the dynamics by projecting out the fast-moving components of the weights. If the noiseless optimization ($\epsilon^2 = 0$) is stable, eq. (1) will map a generic pair (π, w), as $k \to \infty$, to $(0, w_\infty)$, where $w_\infty \in \Gamma$. Define $\Phi : \mathbb{R}^{D\times D} \to \mathbb{R}^D$ to be this mapping, i.e. $\Phi(\pi, w) = w_\infty$. As we now show, when $\epsilon > 0$, Φ can be used precisely to project onto the slow, noise-induced longitudinal dynamics. Let us collectively denote $x_k =$

(π_k, w_k) and write eq. (1) as $x_{k+1} = x_k + F(x_k) + \epsilon\tilde{\sigma}(x_k)\xi_k$. We can perform a Taylor expansion in ϵ to obtain

$$\Phi(x_{k+1}) - \Phi(x_k) = \partial\Phi(x_k)[\epsilon\tilde{\sigma}(x_k)\xi_k] + \frac{1}{2}\partial^2\Phi(x_k)[\epsilon\tilde{\sigma}(x_k)\xi_k, \epsilon\tilde{\sigma}(x_k)\xi_k]\,.$$

Here we neglected the contributions coming from F: $\partial\Phi(x_k)[F(x_k)] + \frac{1}{2}\partial^2\Phi(x_k)[F(x_k), F(x_k)] + \partial^2\Phi(x_k)[F(x_k), \epsilon\tilde{\sigma}(x_k)\xi_k]$. The first two terms can be shown to vanish by exploiting the Taylor expansion of the relation $\Phi(x + F(x)) - \Phi(x) = 0$. The third term averages to zero and is $\Theta(\epsilon^2)$, since gradients are small near the valley, $F = \Theta(\epsilon)$, and thus this term can be neglected. Denoting $Y(t = \epsilon^2 k) = \Phi(x_k)$, the limit dynamics as $\epsilon \to 0$ is then well-approximated by the continuous time equation

$$dY = \partial\Phi(Y)[\tilde{\sigma}(Y)dW] + \frac{1}{2}\partial^2\Phi(Y)[\tilde{\sigma}(Y)\tilde{\sigma}(Y)^\top]dt\,, \tag{3}$$

where we interpreted the time increment $dt = \epsilon^2$, and introduced a rescaled noise satisfying $\langle dW^2\rangle = dt$. In the arguments of the functions on the right-hand side, we replaced x_k with $Y(t)$ as intuitively, for x_k very close to Γ, i.e. $d(x_k, \Gamma) \to 0$ for any $k > 0$ as $\epsilon \to 0$, we have $\Phi(x_k) \approx x_k$. Also, we assumed that $dW^2 = dt$ up to corrections that are subleading in dt. Note that until here we have not taken a small learning rate limit. The learning rate can be finite, as far as the map $\Phi(\pi, x)$ exists. The small noise limit is sufficient to allow a continuous-time description of the limit dynamics, this is because the noise-induced drift-diffusion along the valley requires $\Theta(\epsilon^{-2})$ timesteps to lead to appreciable longitudinal displacements.

A similar approach to what we just described was used in Li et al. (2022), although in our case the limit drift-diffusion is obtained at small noise, rather than at small learning rate. The reason for this choice is that, since we scale $1 - \beta$ as a positive power of η, the deterministic part of eq. (1) becomes degenerate as we take $\eta \to 0$, in which case it would not be possible to apply the mathematical framework of Katzenberger (1991) on which our results below rely. To further simplify our analysis, particularly the statement of Theorem 3.4, we will further take $\eta \to 0$ after taking $\epsilon \to 0$, and retain only leading order contributions in η. While the formalism of Katzenberger (1991) provides an elegant framework to obtain the limit-drift-diffusion at small noise $\epsilon \to 0$, it does not provide a rigorous estimate of the deviation from this scaling law at finite noise. The deviation will be caused by higher-order contributions in ϵ (or $\sqrt{dt}$) in the Taylor expansion in eq. (3). Additionally, in a given training task, the average weights may not exactly lie on the zero-loss manifold, which will also induce a correction to the proposed scaling. Addressing these issues requires developing a framework beyond that of Katzenberger (1991), which is outside the scope of this paper. Nevertheless, in Sec. 4 we observe strong empirical signature of the scaling law in realistic settings, which gives clear evidence that the scaling law we found is robust to these deviations.

The **main contributions** of this paper are:

1. We develop a general formalism to study SGD with (heavy-ball) momentum in Sec. 3, extending the framework of Li et al. (2022) to study convergence rates and generalization with momentum.
2. We find a novel scaling regime of the momentum hyperparameter $1 - \beta \sim \eta^\gamma$, and demonstrate a qualitative change in the noise-induced training dynamics as γ is varied.
3. We identify a special scaling limit, $1 - \beta \sim \eta^{2/3}$, where training achieves a *maximal speedup at fixed learning rate* η.
4. In Sec. 4, we demonstrate the relevance of our theory with experiments on toy models (2-layer neural networks, and matrix sensing) as well as realistic models and datasets (deep MLP on FashionMNIST, and ResNet-18 on CIFAR10).

2 Related Works

Loss Landscape in Overparametrized Networks The geometry of the loss landscape is very hard to understand for real-world models. Choromanska et al. (2015) conjectured, based on empirical observations and on an idealized model, that most local minima have similar loss function values. Subsequent literature has shown in wider generality the existence of a manifold connecting degenerate minima of the loss function, particularly in overparametrized models. This was supported by work on mode connectivity (Freeman & Bruna, 2017; Garipov et al., 2018; Draxler et al., 2018; Kuditipudi et al., 2019), as well as on empirical observations that the loss Hessian possesses a large set of (nearly) vanishing eigenvalues (Sagun et al., 2016; 2017). In particular, Nguyen (2019) showed that for overparametrized networks with piecewise linear activations, all global minima are connected within a unique valley.

The Implicit Regularization of SGD Wei & Schwab (2019), assuming the existence of a zero-loss valley, observed that SGD noise leads to a decrease in the trace of the Hessian. Blanc et al. (2020) demonstrated that SGD with label noise in the overparametrized regime induces a regularized loss that accounts for the decrease in the trace of the Hessian. Damian et al. (2021) extend this analysis to finite learning rate. HaoChen et al. (2021) study the effect of non-isotropic label noise in SGD and find a theoretical advantage in a quadratic overparametrized model. Wu et al. (2022) show that only minima with small enough Hessians (in Frobenius norm) are stable under SGD. The specific regularization induced by SGD was found in quadratic models (Pillaud-Vivien et al., 2022), 2-layer Relu networks (Blanc et al., 2020), linear models (Li et al., 2022), diagonal networks (Pesme et al., 2021). Additionally, Kunin et al. (2021) and Xie et al. (2021) studied the diffusive dynamics induced by SGD both empirically and in a simple theoretical model.

Momentum in SGD and Adaptive Algorithms Popular implementations of momentum include Nesterov (Nesterov, 1983) and Heavy Ball (HB) or Polyak (Polyak, 1964). We focus on the latter in this paper, which we refer to simply as momentum. Momentum provably improves convergence time in the deterministic setting. Less is known rigorously when stochastic gradient updates are used. Indeed, Polyak (1987) suggests the benefits of acceleration with momentum disappear with stochastic optimization unless certain conditions are placed on the properties of the noise. See also (Jain et al., 2018; Kidambi et al., 2018) for more discussion and background on this issue. Nevertheless, in practice it is widely appreciated that momentum is important for convergence and generalization (Sutskever et al., 2013), and widely used in modern adaptive gradient algorithms Kingma & Ba (2015). Some limited results have been obtained showing speedup in the mean-field approximation (Mannelli & Urbani, 2021) and linear regression Jain et al. (2018). Modifications to Nesterov momentum to make it more amenable to stochasticity (Liu & Belkin, 2020; Allen-Zhu, 2017), and near saddle points (Xie et al., 2022) have also been considered.

3 Theoretical Results

3.1 General Setup

Following the outline presented in Sec. 1.3, in this and the following section we will rigorously derive the limiting drift-diffusion equation for the weights on the zero-loss manifold Γ, and extract the timescale τ_2 associated to this noise-induced motion. In Sec. 3.3 we will then compare τ_2 to the timescale τ_1 associated to the noiseless dynamics and evaluate the optimal scaling of β discussed in Sec. 1.1. We will use Eq. (1) to model momentum SGD. As illustrated in Sec. 1.1, the drift is controlled by the second moment of fluctuations, and we thus expect the drift timescale to be $\Theta(\epsilon^2)$. We will then rescale time $k = t/\epsilon^2$, so that the motion in the units of t is $O(1)$ as $\epsilon \to 0$.

Assumption 3.1. *The loss function $L : \mathbb{R}^D \to \mathbb{R}$ is a C^3 function whose first 3 derivatives are locally Lipschitz, σ is continuous, and $\Gamma = \{w \in \mathbb{R}^D : L(w) = 0\}$ is a C^2-submanifold of $\mathbb{R}^D$ of dimension M, with $0 \le M \le D$. Additionally, for $w \in \Gamma$, rank$(\nabla^2 L(w)) = D - M$.*

Assumption 3.2. *There exists an open neighborhood U of $\{0\} \times \Gamma \subseteq \mathbb{R}^D \times \mathbb{R}^D$ such that the gradient descent starting in U converges to a point $x = (\pi, w) \in \{0\} \times \Gamma$. More explicitly, for $x \in U$, let $\psi(x, 0) = x$ and $\psi(x, k+1) = \psi(x, k) + F(\psi(x, k))$, i.e. $\psi(x, k)$ is the k^{th} iteration of $x + F(x)$. Then $\Phi(x) \equiv \lim_{k\to\infty} \psi(x, k)$ exists and is in Γ. As a consequence, $\Phi \in C^2$ on U (Falconer, 1983).*

3.2 Limiting drift-diffusion in momentum SGD

In this section we shall obtain the explicit expression for the limiting drift-diffusion. The general framework is based on Katzenberger (1991) (reviewed in Appendix C). Before stating the result, we will need to introduce a few objects.

Definition 3.3. For a symmetric matrix $H \in \mathbb{R}^D \times \mathbb{R}^D$, and $W_H = \{\Sigma \in \mathbb{R}^D \times \mathbb{R}^D : \Sigma = \Sigma^\top, HH^\dagger\Sigma = H^\dagger H\Sigma = \sigma\}$, we define the operator $\tilde{\mathcal{L}}_H : W_H \to W_H$ with $\tilde{\mathcal{L}}_H S \equiv \{H, S\} + \frac{1}{2}C^{-2}\eta^{1-2\gamma}[[S, H], H]$, with $[S, H] = SH - HS$. It can be shown that the operator $\tilde{\mathcal{L}}_H$ is invertible (see Lemma D.3).

Theorem 3.4 (Informal)**.** *Suppose the loss function L, the noise function σ, the manifold of minimizers Γ and the neighborhood U satisfy assumptions (3.1) and (3.2), and that $X_n(0) = X(0) \in U$. Set $\beta = 1 - C\eta^\gamma$. Then, as $\epsilon_n \to 0$, and subsequently taking $\eta \to 0$, $Y_n(t)$ converges to $Y(t)$, where the latter satisfies the limiting drift-diffusion equation*

$$\begin{aligned} dY =& (\tfrac{1}{C}\eta^{1-\gamma} + \eta)P_L\sigma dW - \tfrac{1}{2C^2}\eta^{2-2\gamma}(\nabla^2 L)^\dagger\partial^2(\nabla L)[\Sigma_{LL}]dt \\ & - \tfrac{1}{C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[(\nabla^2 L)^\dagger\Sigma_{TL}]dt - \tfrac{1}{2C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[\tilde{\mathcal{L}}^{-1}_{\nabla^2 L}\Sigma_{TT}]dt\,, \end{aligned} \tag{4}$$

where $\Sigma \equiv \sigma\sigma^\top$, $\Sigma_{LL} = P_L\Sigma P_L$, $\Sigma_{TL} = P_T\Sigma P_L$, $\Sigma_{TT} = P_T\Sigma P_T$, and $W(t)$ is a Wiener process.

A rigorous version of this theorem is given in Sec. C (and the assumption $X_\epsilon(0) \in \Gamma$ is relaxed to $X_\epsilon(0) \in U$). The first term in Eq. (4) induces diffusion in the longitudinal direction. The second term is of geometrical nature, and is necessary to guarantee that $Y(t)$ remains on Γ. The second line describes the drift induced by the transverse fluctuations.

Eq. (4) resembles in form that found in Li et al. (2022), although there are two crucial differences. First, time has been rescaled using the strength of the noise ϵ, rather than the learning rate. The different rescaling was necessary as the deterministic part of eq. (1) (i.e., the forcing term F in Eq. (20)) depends non-homogeneously on η, and thus the theory of Katzenberger (1991) would not be directly applied had we taken the small learning rate limit. Second, and more crucially, the drift terms in Eq. (4) are proportional to $\eta^{2-2\gamma}$, which is a key ingredient leading to the change in hierarchy of the timescales discussed in Sec. 1.1. One final difference, is that the last term involves the operator $\tilde{\mathcal{L}}_H$ instead of the Lyapunov operator. For $\gamma < \frac{1}{2}$, $\tilde{\mathcal{L}}_H$ reduces to the Lyapunov operator $\mathcal{L}_H$ at leading order in η, with $\mathcal{L}_H S \equiv \{H, S\}$. For $\gamma > \frac{1}{2}$, however, we cannot neglect the η-dependent term in $\tilde{\mathcal{L}}_H$ (see discussion at the end of Appendix D).

Corollary 3.5. *In the case of label noise, i.e. when, for $w \in \Gamma$, $\Sigma = c\nabla^2 L$, for some constant $c > 0$, Eq. (4) reduces to*

$$dY = -\frac{\epsilon^2\eta^{2-2\gamma}}{4C^2}P_L\nabla\, Tr(c\nabla^2 L)dt\,, \tag{5}$$

where we have rescaled time back to $t = k$, i.e. we performed $t \to t\epsilon^2$.

3.3 Separation of timescales and optimal momentum scaling

The above results provide the estimate for the timescale τ_2 of the drift along the zero-loss valley. As discussed in Sec. 1.1, training along the zero-loss manifold Γ is maximally accelerated if this time scale is equal to the timescale τ_1 for relaxation of off-valley perturbations. As we take $\epsilon \to 0$, this relaxation is governed by the deterministic gradient descent dynamics, i.e. τ_1 will be determined by the nonzero eigenvalues of the Hessian as well as by the learning rate η and momentum β. Therefore we expect $\tau_1 = \Theta(\epsilon^0)$, as will be confirmed by the analysis below. Since we are working in the regime of small noise, we will obtain the leading order expression of τ_1 by focusing on the noiseless $\epsilon = 0$ dynamics. Additionally, since we are interested in local relaxation, it will suffice to look at the linearized dynamics around Γ.

Working in the extended phase space $x_k = (\pi_k, w_k)$, and linearizing Eq. (1) around a fixed point $x^* = (0, w^*)$, with $w^* \in \Gamma$, the linearized update rule is $\delta x_{k+1} = J(x^*)\delta x_k$, where $\delta x_k = x_k - x^*$ and $J(x^*)$ is the Jacobian evaluated at the fixed point (with the explicit form given in Eq. (131)).

Denote by q^i the eigenvector and λ_i the corresponding eigenvalue of the Hessian. We show in Appendix G that the Jacobian is diagonalized by the eigenvectors $k^i_\pm = \left(\mu^i_\pm q^i, q^i\right)$ with eigenvalues

$$\kappa^i_\pm = \frac{1}{2}\left(1 + \beta - \eta\lambda_i \pm \sqrt{(1 + \beta - \eta\lambda_i)^2 - 4\beta}\right), \tag{6}$$

and $\mu^i_\pm = \beta\eta\kappa^i_\pm - 1 - \eta\lambda_i$. We proceed to study the decay rate of the different modes of the Jacobian to draw conclusions about the characteristic timescales of fluctuations around the valley.

Longitudinal motion: On the valley, the Hessian will have a number of "zero modes" with $\lambda_i = 1$. These lead to two distinct modes in the present setting with momentum, which we distinguish as pure and mixed. The first pure longitudinal mode is an exact zero mode which has $\kappa^i_+ = 1$ with $k^i_+ = (0, q^i)$, corresponding to translations of the parameters along the manifold, and keeping $\pi = 0$ at its fixed point value. The second mode is a mixed longitudinal mode with $\kappa^i_- = \beta$ with $k^i_- = (-(1-\beta)/(2\beta\eta)q^i, q^i)$. This mode has a component of π along the valley, which must subsequently decay because the equilibrium is a single point $\pi = 0$. Therefore, this mode decays at the characteristic rate β for π, gleaned directly from Eq. (1).

Transverse motion: When the w and π are perturbed along the transverse directions q^i with positive λ_i, the relaxation behavior exhibits a qualitative change depending on β. Using the parameterization of β as in Theorem 3.4, $\beta = 1 - C\eta^\gamma$, for small learning rate, the spectrum is purely real for $\gamma < 1/2$, and comes in complex conjugate pairs for $\gamma > 1/2$. This leads to two distinct scaling behaviors for the set of timescales. Defining a positive $c_1 \leq \min\{\lambda_i | \lambda_i > 0\}$, we find: 1) For $\gamma < 1/2$, transverse modes are purely decaying as $(1 - C\eta^\gamma)^k \leq |\delta x^T_k| \leq (1 - (c_1/C)\eta^{1-\gamma})^k$, with the lower bound set by the mixed longitudinal mode. For $\gamma > 1/2$, the transverse modes are oscillatory but with an envelope that decays like $|\delta x^{T,env}_k| \approx (1 - C\eta^\gamma)^{k/2}$.

We leave the derivation of these results to Appendix (G).

Collecting these results, we can describe the hierarchy of timescales τ_1 in the deterministic regime as a function of γ (excluding the pure longitudinal zero mode and neglecting multiplicites):

τ_1^{-1}	$\gamma < 1/2$	$\gamma > 1/2$
Long.	η^γ	η^γ
Transv.	$\eta^{1-\gamma}, \eta^\gamma$	η^γ, η^γ

These are illustrated schematically in Fig. 1(a), where the finite timescales are shown as a function of γ. We compare these "equilibration" timescales τ_1, i.e. characteristic timescales associated with relaxation back to the zero-loss manifold, with the timescale $\tau_2 \sim \eta^{2(\gamma-1)}$ associated with drift-diffusion of the noise-driven motion along the zero-loss manifold Eq. (4). For small γ, the timescale associated with the drift-diffusion along the valley is much faster than that associated with the relaxation of the dynamics toward steady state. Transverse and mixed longitudinal fluctuations relax much faster than the motion along the valley, and produce an effective drift toward the minimizer of the implicit regularizer. However, the timescales collide at $\gamma = 2/3$, suggesting a transition to a qualitatively different transport above this value, where the transverse and the mixed longitudinal dynamics, having a long timescale, will disrupt the longitudinal drift Eq. (4). This leads us to propose $\gamma = \frac{2}{3}$ as the optimal choice for SGD training. We consistently find evidence for such a qualitative transition in our experiments below. In addition, we see that speedup of SGD with label noise is in fact maximal at this value where the timescales meet.

4 Experimental Validation

4.1 2-layer MLP with Linear and Non-Linear Activations

We first consider 2-layer linear models parametrized as $f(x) = \frac{1}{\sqrt{n}} U \cdot \sigma(Vx)$, where $x \in \mathbb{R}$, and $U, V \in \mathbb{R}^n$ are the trainable parameters, and σ is an activation function. We used linear, tanh and ReLU activation acting on the first layer and the corresponding timescales are shown in Fig. 1(b,c,d). The timescale power α is defined through the relation $T_c = T_0\eta^{-\alpha}$, where T_c is the number of time steps it takes to the weights U, V to reach a small threshold distance from the widest minimum (see App. A.1 for details), and T_0 is an overall scale independent of η. The optimal scaling value, corresponding to the minimum α, happens at $\gamma = \frac{2}{3}$, consistently with our predictions. Additionally, we find that the optimal prefactor C defined in Eq. (2) is approximately 0.2 for all activations. We obtained an analytical estimate of C for the linear activation case in App. E.

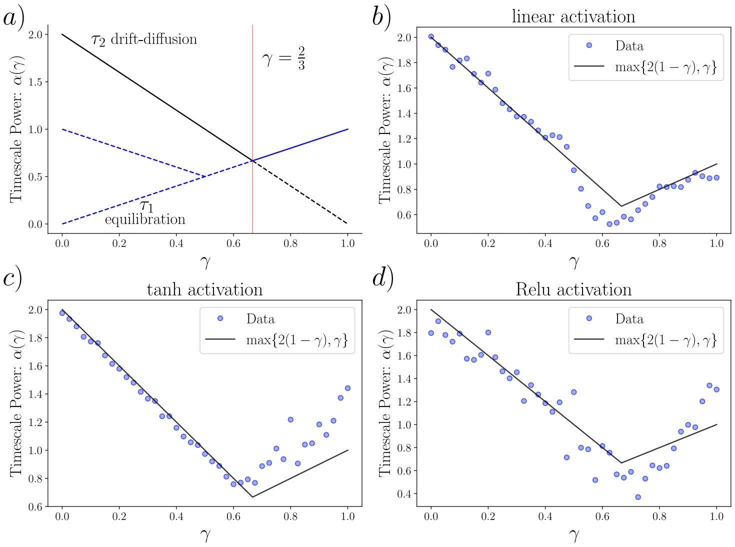

Figure 1: Timescale of training as a function of γ. $a)$: theoretical prediction with blue line representing the timescale for equilibration while the black line shows the timescale of the drift along Γ. The maximum of these two gives the overall timescale. In $b)$, $c)$, and $d)$ we demonstrate this result with the vector UV model with linear, $\tanh$ and Relu activations respectively.

4.2 MLP ON FASHIONMNIST

We also demonstrate agreement with theory on another model-dataset pairing. In this case we train a 6-layer fully connected MLP to classify images in FashionMNIST (full model details given in Appendix A.4). As before, we train from an interpolating solution on the zero-loss manifold with various values of the momentum hyperparameter at each learning rate and extract the optimal $\beta_*(\eta)$. We estimate the exponent $\gamma = 0.64 \pm 0.05$ which again agrees tightly with our theoretical expectation. These experiments show that a range of models support the specific valley structure in which our analysis holds for realistic hyperparameters.

4.3 RESNET-18 ON CIFAR10

We now verify our predictions on larger scale experiments, which will demonstrate the robustness of our analysis. We focus on ResNet-18 (He et al., 2016), specifically implemented by Liu (2021), classifier trained on CIFAR10 Krizhevsky & Hinton (2009). We aim to extrapolate the theory by showing optimal acceleration with our hyperparameter choice once training reaches an interpolating solution. To this purpose, we initialize the network on the valley, obtained starting from a random weight values and training the network using full batch gradient descent without label noise and with a fixed value $\beta = 0.9$ until it reaches perfect training accuracy. With this initialization, we then train with SGD and label noise for a fixed number of epochs multiple times for various values the momentum hyperparameter β. Finally, we project the weights back onto the valley before recording the final test accuracy. This last step can be viewed as noise annealing and allows us to compare the performance of training the drift phase for the different values of β. From this procedure we extract the optimal momentum parameter $\beta_*(\eta)$ that maximizes the best test accuracy during training as a function of the learning rate, which we can then compare with the theoretical prediction. A diagram of this procedure is shown in Fig. 2.

As shown in Fig. 2(b), $1 - \beta_*$ follows the power law we predicted almost exactly with an exponent $\gamma = 0.66 \pm .06$. The optimal choice for speedup does not have to coincide with the optimal choice for generalization. Strikingly, this optimal choice of scaling also leads to the best generalization in a realistic setting! This can be easily interpreted if we assume that as the Hessian decreases more the better the model generalizes better and by applying the fact that our scaling leads to the fastest transport along the manifold towards a smaller Hessian. The second important point is the value

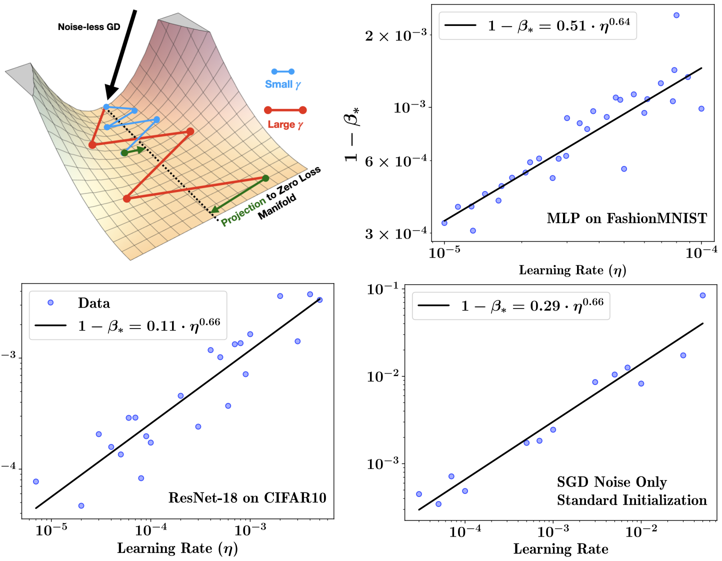

Figure 2: Optimal scaling in various experiments. Top left: illustration of the training protocol for the FashionMNIST and CIFAR10 experiments of Secs. 4.2 and 4.3. We first use noiseless gradient descent (black) to reach the zero-loss manifold, then train with SGD label noise with various (blue, red) values of η and β before finally projecting onto the valley (green) and measuring the test accuracy. Top right and bottom left: scaling of the optimal momentum, β_*, as a function of η on CIFAR10 (FashionMNIST), giving exponents $\gamma_* = 0.64 \pm 0.05$ and $\gamma_* = 0.66 \pm 0.06$ respectively (one standard deviation), which closely match the value predicted by theory $\gamma = \frac{2}{3}$. Bottom right: the CIFAR10 experiment of Sec. 4.3.1 with standard initialization and with only SGD noise also displays scaling behavior with exponent 0.66 ± 0.05. Across all experiments, the constant C is in the range $0.1 - 0.5$.

of the constant $C \approx 0.1$ found as the coefficient of the power-law fit. Curiously, if we set $\eta = 1$ this corresponds to setting $\beta_* = 0.9$ which is the traditionally recommended value. The result here, can therefore be viewed as a modification of this common wisdom when training near a manifold of minimizers. For more experiments, we refer the reader to Appendices A.2 and A.4.

4.3.1 A Standard Training Setup

In experiments using a standard setup with random initialization, away from any valley, and SGD noise instead of label noise, we observed that the optimal value of β still follows a power law, and with the same exponent ($\gamma = \frac{2}{3}$) as in previous experiments. This result, while going beyond the assumptions of our theoretical framework, suggests that scaling relationships can be beneficial even in practical settings outside our theory's strict regime. This robustness might be partly due to the model's weights being near a zero-loss valley during significant portions of training, despite the random initialization.

5 Conclusion

In our study of SGD's implicit regularization with label noise and momentum, we discovered that momentum's speedup and the diffusion from SGD noise interact, generating two characteristic training timescales. Optimal training and generalization occur when these timescales have similar length, which leads to identifying an optimal hyperparameter scaling relationship. It will be interesting to extend our study to adaptive algorithms like Adam and its variants, which we expect to lead to novel scaling relationships between various hyperparameters, as well as to hyperparameter schedule. It will also be interesting to study the interplay of this scaling with modern insights on initialization and normalization (Jacot et al., 2018; Roberts et al., 2021; Yang et al., 2022).

References

Zeyuan Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. The Journal of Machine Learning Research, 18(1):8194–8244, 2017.

Guy Blanc, Neha Gupta, Gregory Valiant, and Paul Valiant. Implicit regularization for deep neural networks driven by an Ornstein-Uhlenbeck like process. In Conference on learning theory, pp. 483–513. PMLR, 2020.

Dami Choi, Christopher J Shallue, Zachary Nado, Jaehoon Lee, Chris J Maddison, and George E Dahl. On empirical comparisons of optimizers for deep learning. arXiv preprint arXiv:1910.05446, 2019.

Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In Artificial intelligence and statistics, pp. 192–204. PMLR, 2015.

Alex Damian, Tengyu Ma, and Jason D Lee. Label noise SGD provably prefers flat global minimizers. Advances in Neural Information Processing Systems, 34:27449–27461, 2021.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In International conference on machine learning, pp. 1309–1318. PMLR, 2018.

Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. arXiv preprint arXiv:1703.11008, 2017.

KJ Falconer. Differentiation of the limit mapping in a dynamical system. Journal of the London Mathematical Society, 2(2):356–372, 1983.

C Daniel Freeman and Joan Bruna. Topology and geometry of half-rectified network optimization. ICLR, 2017.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of DNNs. Advances in neural information processing systems, 31, 2018.

Jeff Z. HaoChen, Colin Wei, Jason Lee, and Tengyu Ma. Shape matters: Understanding the implicit bias of the noise covariance. In Conference on Learning Theory, volume 134 of Proceedings of Machine Learning Research, pp. 2315–2357. PMLR, 15–19 Aug 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. Neural computation, 9(1):1–42, 1997.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. Advances in neural information processing systems, 31, 2018.

Prateek Jain, Sham M Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Accelerating stochastic gradient descent for least squares regression. In Conference On Learning Theory, pp. 545–604. PMLR, 2018.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. arXiv preprint arXiv:1912.02178, 2019.

Gary Shon Katzenberger. Solutions of a stochastic differential equation forced onto a manifold by a large drift. The Annals of Probability, 19(4):1587–1628, 1991.

Nitish Shirish Keskar and Richard Socher. Improving generalization performance by switching from Adam to SGD. arXiv preprint arXiv:1712.07628, 2017.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. ICLR, 2017.

Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world. arXiv preprint arXiv:2002.03329, 2020.

Rahul Kidambi, Praneeth Netrapalli, Prateek Jain, and Sham Kakade. On the insufficiency of existing momentum schemes for stochastic optimization. In Information Theory and Applications Workshop (ITA), pp. 1–9. IEEE, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. ICLR, 2015.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Toronto, ON, Canada, 2009.

Rohith Kuditipudi, Xiang Wang, Holden Lee, Yi Zhang, Zhiyuan Li, Wei Hu, Rong Ge, and Sanjeev Arora. Explaining landscape connectivity of low-cost solutions for multilayer nets. Advances in neural information processing systems, 32, 2019.

Daniel Kunin, Javier Sagastuy-Brena, Lauren Gillespie, Eshed Margalit, Hidenori Tanaka, Surya Ganguli, and Daniel LK Yamins. Limiting dynamics of sgd: Modified loss, phase space oscillations, and anomalous diffusion. arXiv preprint arXiv:2107.09133, 2021.

Kiwon Lee, Andrew N Cheng, Courtney Paquette, and Elliot Paquette. Trajectory of mini-batch momentum: Batch size saturation and convergence in high dimensions. arXiv preprint arXiv:2206.01029, 2022.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. arXiv preprint arXiv:2003.02218, 2020.

Yuanzhi Li, Tengyu Ma, and Hongyang Zhang. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In Conference On Learning Theory, pp. 2–47. PMLR, 2018.

Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss?–a mathematical framework. ICLR, 2022.

Chaoyue Liu and Mikhail Belkin. Accelerating SGD with momentum for over-parameterized learning. ICLR, 2020.

Kuang Liu. Train cifar10 with pytorch. https://github.com/kuangliu/pytorch-cifar/, February 2021. commit/49b7aa97b0c12fe0d4054e670403a16b6b834ddd.

Shengchao Liu, Dimitris Papailiopoulos, and Dimitris Achlioptas. Bad global minima exist and sgd can reach them. Advances in Neural Information Processing Systems, 33:8543–8552, 2020.

Stefano Sarao Mannelli and Pierfrancesco Urbani. Just a momentum: Analytical study of momentum-based acceleration methods in paradigmatic high-dimensional non-convex problems. arXiv preprint arXiv:2102.11755, 2021.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. ICLR, 2020.

Yurii E Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. In Dokl. akad. nauk Sssr, volume 269, pp. 543–547, 1983.

Quynh Nguyen. On connected sublevel sets in deep learning. In International conference on machine learning, pp. 4790–4799. PMLR, 2019.

Scott Pesme, Loucas Pillaud-Vivien, and Nicolas Flammarion. Implicit bias of sgd for diagonal linear networks: a provable benefit of stochasticity. Advances in Neural Information Processing Systems, 34:29218–29230, 2021.

Loucas Pillaud-Vivien, Julien Reygner, and Nicolas Flammarion. Label noise (stochastic) gradient descent implicitly solves the lasso for quadratic parametrisation. arXiv preprint arXiv:2206.09841, 2022.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. Ussr computational mathematics and mathematical physics, 4(5):1–17, 1964.

Boris T Polyak. Introduction to optimization. optimization software. Inc., Publications Division, New York, 1:32, 1987.

Jasson DM Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In Proceedings of the 22nd international conference on Machine learning, pp. 713–719, 2005.

Daniel A Roberts, Sho Yaida, and Boris Hanin. The principles of deep learning theory. arXiv preprint arXiv:2106.10165, 2021.

Levent Sagun, Leon Bottou, and Yann LeCun. Eigenvalues of the hessian in deep learning: Singularity and beyond. arXiv preprint arXiv:1611.07476, 2016.

Levent Sagun, Utku Evci, V Ugur Guney, Yann Dauphin, and Leon Bottou. Empirical analysis of the hessian of over-parametrized neural networks. arXiv preprint arXiv:1706.04454, 2017.

Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. ICLR, 2014.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. arxiv e-prints, art. arXiv preprint arXiv:1710.10345, 2017.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In International conference on machine learning, pp. 1139–1147. PMLR, 2013.

Mingwei Wei and David J Schwab. How noise affects the hessian spectrum in overparameterized neural networks. arXiv preprint arXiv:1910.00195, 2019.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. Advances in neural information processing systems, 30, 2017.

Lei Wu, Chao Ma, et al. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. Advances in Neural Information Processing Systems, 31, 2018.

Lei Wu, Mingze Wang, and Weijie Su. When does sgd favor flat minima? a quantitative characterization via linear stability. arXiv preprint arXiv:2207.02628, 2022.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747, 2017.

Zeke Xie, Issei Sato, and Masashi Sugiyama. A diffusion theory for deep learning dynamics: Stochastic gradient descent exponentially favors flat minima. arXiv preprint arXiv:2002.03495, 2020.

Zeke Xie, Issei Sato, and Masashi Sugiyama. A diffusion theory for deep learning dynamics: Stochastic gradient descent exponentially favors flat minima. In International Conference on Learning Representations, 2021. URL https://openreview.net/forum?id=wXgk_iCiYGo.

Zeke Xie, Xinrui Wang, Huishuai Zhang, Issei Sato, and Masashi Sugiyama. Adaptive inertia: Disentangling the effects of adaptive learning rate and momentum. In International Conference on Machine Learning, pp. 24430–24459. PMLR, 2022.

Greg Yang, Edward J Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tensor programs v: Tuning large neural networks via zero-shot hyperparameter transfer. arXiv preprint arXiv:2203.03466, 2022.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. corr abs/1611.03530. ICLR, 2017.

A Experimental Design

A.1 UV Model

The first experiment we consider is a 2-layer linear model parametrized as $f(x) = \frac{1}{\sqrt{n}} U \cdot V x$, where $x \in \mathbb{R}$, and $U, V \in \mathbb{R}^n$ are the trainable parameters. This model is studied analytically in App. E. Our goal with this experiment is to analyze a simple model, and show quantitative agreement with theoretical expectations. Though simple, this model shows almost all of the important features in our analysis: the $2(1-\gamma)$ exponent below $\gamma = \frac{2}{3}$, the γ exponent above $\frac{2}{3}$, and the constant C theoretically evaluated in App. E.

We train on an artificially generated dataset $\mathcal{D} = \{(x^a, y^a)\}$ with $x \sim \mathcal{N}(0,1)$ and $y = 0$. We initialize with $U_i, V_i \sim \mathcal{N}(0,1)$ and keep this initialization constant over all our experiments to reduce noise associated with the specific initialization. We use the full dataset to train. The loss of this model is

$$L = \frac{1}{2P} \sum_{i=1}^{5} \left(y_i - \frac{1}{\sqrt{n}} U \cdot V x_i \right)^2 \tag{7}$$

From Eq.(122) we know that the norm of the weights follows an approximately exponential trajectory as it approaches the widest minimum ($U = V = 0$). We therefore measure convergence timescale, T_c, by fitting an exponential ae^{-t/T_c} to the squared distance from the origin, $|U|^2 + |V|^2$.

To extract the scaling of T_c with γ we perform SGD label noise with learning rates $\eta \in [10^{-3}, 10^{-1}]$ and corresponding momentum parameters $\beta = 1 - C\eta^{\gamma}$. We fit the timescale to a power-law in the learning rate $T_c(\eta, \gamma) = T_0 \eta^{-\alpha(\gamma)}$ (see Fig. 1(b)). Imposing that T_0 be independent of γ, as predicted by theory, we found the numerical value $C \approx 0.2$, which is consistent with the theoretical estimate of $C = 0.17$ from Sec. E. We find consistency with prediction across all the values of γ we simulated. Note that, for $\gamma > \frac{2}{3}$ the timescale estimate fluctuates more which is a consequence of having a slower timescale for the transverse modes. As discussed at the end of Sec. 3.3, such slowness disrupts the drift motion along the manifold. $\gamma = \frac{2}{3}$ is clearly the optimal scaling. A similar setup was implemented for tanh and ReLU activations.

A.2 Matrix Sensing

We also explore speedup for a well understood problem: matrix sensing. The goal is to find a low-rank matrix X^* given the measurements along random matrices A_i: $y_i = \mathrm{Tr} A_i X^*$. Here $X^* \in \mathbb{R}^{d \times d}$ is a matrix of rank r (Soudry et al., 2017; Li et al., 2018).

Blanc et al. (2020) analyze the problem of matrix sensing using SGD with label noise and show that if X^* is symmetric with the hypothesis $X = UU^\top$ for some matrix U, then gradient descent with label noise corresponds not only to satisfying the constraints $y_i = \mathrm{Tr} A_i X$, but also to minimizing an implicit regularizer, the Frobenius norm of U, which eventually leads to the ground truth.

In the analogous UV matrix model (where X^* is an asymmetric matrix of low rank r), we demonstrate a considerable learning speedup by adding momentum, and show that this speedup is not monotonic with increasing β; there is a value β^* at which the acceleration appears optimal. This non-monotonicity with an optimal β^* is observed for both the Hessian trace and the expected test error. Assuming that in this setting we also have $\gamma = 2/3$, we can extract $C^* = (1 - \beta^*)/\eta^{2/3} \approx 0.24\, P^{-1/3}$, which compares favorably to the upper bound we may extract from Appendix F of $\approx 0.12\, P^{-1/3}$.

In the experiments with matrix sensing we aim to demonstrate the benefit of momentum in a popular setting. Matrix sensing corresponds to the following problem: Given a target matrix $X^* \in \mathbb{R}^{d \times d}$ of low rank $r \ll d$ and measurements $\{y_i = \mathrm{Tr} A_i X^*\}_{i=1}^{P}$ how can we reconstruct X^*? One way to solve this problem is to write our guess $X = UV$ the product of two other matrices, and do stochastic gradient descent on them, hoping that the implicit regularization induced by this parametrization and the learning algorithm will converge to a good low rank X.

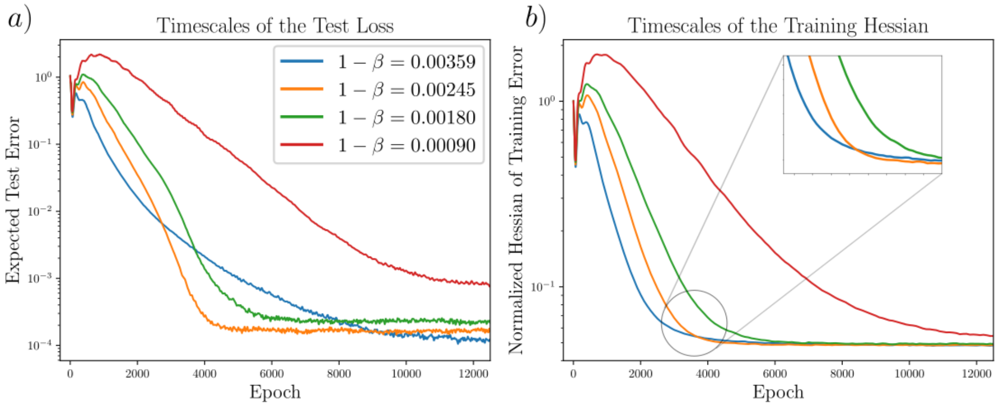

Figure 3: The expected test error, a), and Hessian of the training loss, b), in matrix sensing (with $d = 100, r = 5$, and $5rd = 2500$ samples) as a function of training epoch plotted for different values of β at $\eta = 0.1$. The label noise variance is 0.1. Each curve represents a different value of β. The inset shows that the orange curve crosses below the blue curve before convergence of the Hessian. Therefore, the same value of β is optimal for both the Hessian and the expected test error — increasing or decreasing β from this value slows down generalization.

A.2.1 Experimental Details

In our experiments we study the $d = 100, r = 5, P = 5rd = 2500$ case. We draw $(A_i)_{ij} \sim \mathcal{N}(0, 1)$ as standard Gaussians and choose X^* by drawing first $(X_0)_{ij} \sim \mathcal{N}(0, 1)$ and then performing SVD and projecting onto the top r singular values by zeroing out the smaller singular values in the diagonal matrix. We intitalize $U = V = I_d$. We perform SGD with momentum on the time dependent loss (with label noise depending on time)

$$L(t) = \frac{1}{dP}\sum_{i=1}^{P}\left(\epsilon \cdot \xi_i(t) + y_i - \text{Tr}(A_i UV)\right)^2 \tag{8}$$

where $\epsilon^2 = 0.1$, $\xi_i(t) \sim \mathcal{N}(0, 1)$. We choose $\eta = 0.1$ for all of our experiments.

The hessian of the loss is defined to be the Hessian averaged over the noise. Equivalently we may just set $\xi_i(t) = 0$ when we calculate the Hessian because averaging over the noise decouples the noise. Similarly when we define the expected test loss we define it as an average over all A_i setting $\xi_i(t) = 0$ in order to decouple the noise. Averaging over $\xi_i(t)$ and A_i would simply lead to an additional term $\langle \xi_i(t)^2 \rangle$ which would simply contribute a constant. We remove this constant for clarity. As a result, the expected test that we plot is proportional to the squared Frobenius norm of the difference between the model UV and the target X^*,

$$\langle L \rangle = \frac{1}{d}||UV - X^*||_F^2. \tag{9}$$

It is also interesting to note that we observe epoch-wise double descent Nakkiran et al. (2020) in this problem. In particular, we observe that the peak in the test error can be controlled by the momentum hyperparameter, and becomes especially pronounced for $\beta \to 1$.

A.3 ResNet-18 on CIFAR10

We train our model in three steps: full batch without label noise until 100% train accuracy, SGD with label noise and momentum, and then a final projection onto the interpolating manifold. The model we use is the ResNet-18 and we train on the CIFAR10 training set.

The first step is full batch gradient descent on the full CIFAR10 training set of 50,000 samples. We train with a learning rate $\eta = 0.1$ and momentum $\beta = 0.9$ and a learning rate schedule with linearly increases from 0 to η over 600 epochs, after which it undergoes a cosine learning rate schedule for

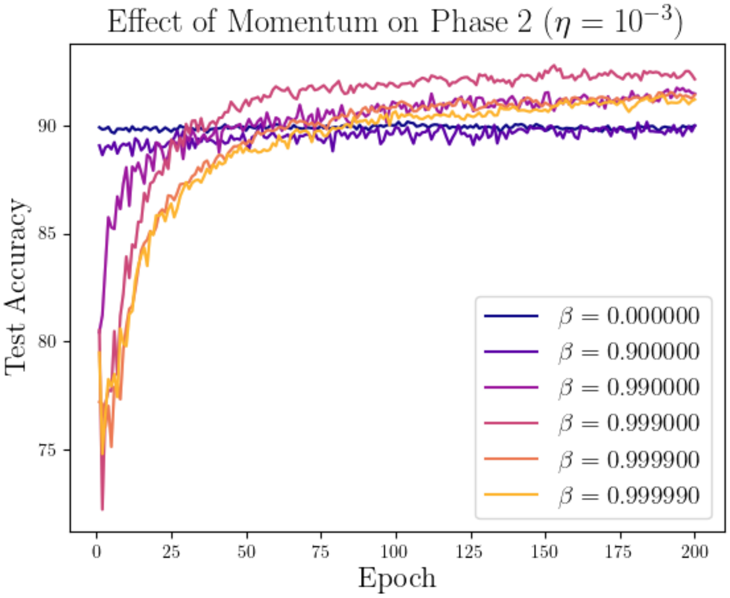

Figure 4: A sample of the curves with different momentum hyperparameters β with $\eta = 0.001$ with Resnet on CIRAR10. The speed of increase in accuracy is non-monotonic in β: the best performance is obtained by an intermediate value of β, consistent with our predictions.

1400 more epochs stopping the first time the network reaches 100% train accuracy which happened on epoch 1119 in our run. This model is saved and the same one is used for all future runs.

The loss function we use is cross cross entropy loss. Because we will choose a label noise level of $p = 0.2$ which corresponds to a uniformly wrong label 20% of the time, during this phase of training we train with the expected loss over this randomness. Notice that this loss is actually linear in the labels so taking the expectation is easy.

The second step involves starting from the initialization in step 1 and training with a different learning rate and momentum parameter. In this step we choose the same level of label noise $p = 0.2$ but take it to be stochastic. Additionally we use SGD instead of gradient descent with a batch size of 512. This necessitates decreasing the learning rate because noise is greatly increased as demonstrated in the main text. In this step we train for a fixed 200 epochs for any learning rate momentum combination. We only compare runs with the same learning rate value. We show an example of the test accuracy as we train in phase 2 for $\eta = 0.001$ in figure 4.

Notice that for both too-small and too-large momentum values that the convergence to a good test accuracy value is slower. The initial transient with the decreased test accuracy happens as we start on the valley and adding noise coupled with momentum causes the weights to approach their equilibrium distribution about the valley. For wider distributions the network is farther from the optimal point on the valley. As training proceeds we see that the test accuracy actually increases over the baseline as the hessian decreases and the generalization capacity of the network increases. This happens most quickly for the momentum which matches our scaling law.

The final step is a projection onto the zero loss manifold. This step is necessary because the total width of the distribution around the zero loss manifold scales with $\frac{1}{1-\beta}$, and this will distort the results systematically at larger momentum, making them look worse than they are. We perform this projection to correct for this effect and put all momenta on an equal footing. This projection is done by training on the full batch with Adam and a learning rate of $\eta = 0.0001$ in order to accelerate training. We do not expect any significant systematic distortion by using Adam for the projection instead of gradient descent.

To determine the optimal value of β we sweep several values of β and observe the test accuracy after the previously described procedure. To get a more precise value of β instead of simply selecting the one with the highest test accuracy we fit the accuracy $A(\beta)$ to

$$A(\beta) = a_{max} + \begin{cases} a_1(\beta - \beta_*) & \text{if } \beta \leq \beta_* \\ a_2(\beta - \beta_*) & \text{if } \beta \geq \beta_* \end{cases} \tag{10}$$

to the parameters a_{max}, a_1, a_2, and β_*, thereby extracting β_* for each η.

We performed a similar analysis in Sec. 4.3.1. The difference is that, instead of dividing training into three phases, we only used one phase where we implemented SGD with constant hyperparameters, where we implemented the same sweep of β as above and eq. (10) to determine the optimal β for a given η. We used Kaiming initialization, and the minibatch size 512. We maintained a label smoothing of 0.2, the same value used in the paragraphs above.

A.4 MLP on FashionMNIST

We perform a similar experiment as in Sec. A.3 but with different model and dataset: a 6-layer MLP trained on FashionMNIST (Xiao et al., 2017) as our dataset. We perform the experiment with a 6 layer MLP with Relu activation after the first 4 layers, $\tanh$ activation after the fifth layer, and a linear mapping to logits. The hidden layers have size 378, 183, 88, 42, and 20 with the input and output dimension being 28^2 and 10 respectively. We use cross entropy loss with label noise $p = 0.2$ as before, and always start training from a reference point initialized on the zero loss manifold. This point was obtained by gradient descent on the expected loss from a random initialized point with learning rate $\eta = 0.002$ and without momentum.

After this we sweep $\eta \in [10^{-5}, 10^{-4}]$ and $\beta \in [0.95, 1 - 2\eta]$ and train for 600 epochs with label noise and 400 without label noise. This allows us to obtain a test accuracy as a function of η and β and therefore we can obtain the best momentum hyperparameter, β_*, as a function of η as in A.3. We extract the scaling exponent by doing a linear fit between $\log(1 - \beta_*)$ and $\log \eta$. The scaling analysis is shown in Fig. 2, and shows that the exponent is consistent with our theory.

B Definition of τ_2

We here provide the definition of τ_2:

Definition B.1. Let w_k denote the weights of a neural network at time step k. Assume that the training algorithm is in particular defined, among other hyperparameters, by noise strength ϵ and a learning rate η. For a given constant M independent of ϵ and η, define T_M as the number of time steps it takes for the longitudinal displacement in the weights along the valley to grow larger than M, i.e. $P_L(\langle \delta w_{k=T_M} \rangle - \langle \delta w_{k=0} \rangle) \geq M$. T_M is a function of η, ϵ, and M. We define τ_2 as the leading order contribution to T_M in the limit as $\epsilon \to 0$ followed by $\eta \to 0$.

We note that M should be sufficiently small so that T_M is finite. The precise value of M will not matter as in our theoretical discussion we are interested in the scaling dependence on η and ϵ, not in overall numerical factors, as far as these is well-defined and finite.

C Review of Relevant Results from Katzenberger (1991)

In this Appendix we summarize the relevant conditions and theorems from Katzenberger (1991) that we use to prove our result on the limiting drift-diffusion. We refer to Katzenberger (1991) for part of the definitions and conditions cited throughout the below. In what follows, $(\Omega^n, \mathcal{F}^n, \{\mathcal{F}^n_t\}_{t\geq 0}, P)$ will denote a filtered probability space, Z_n an $\mathbb{R}^r$-valued cadlag $\{\mathcal{F}^n_t\}$-semimartingale with $Z_n(0) = 0$, A_n a real-valued cadlag $\{\mathcal{F}^n_t\}$-adapted nondecreasing process with $A_n(0) = 0$, and $\tilde{\sigma} : U \to \mathbb{R}^{D\times r}$ a continuous function, where U is a neighborhood of $\{\mathbf{0}\} \times \Gamma$ as defined in the main text. Also, X_n is an $\mathbb{R}^D$ valued cadlag $\{\mathcal{F}^n_t\}$-semimartingale satisfying

$$X_n(t) = X_n(0) + \int_0^t \sigma_n(X_n) dZ_n + \int_0^t F(X_n) dA_n \tag{11}$$

for all $t \leq \lambda_n(K)$ and all compact $K \subset U$, where

$$\lambda_n(K) = \inf\{t \geq 0 | X_n(t-) \neq \mathring{K} or X_n(t) \neq \mathring{K}\} \tag{12}$$

be the stopping time of $X_n(t)$ to leave $\mathring{K}$, the interior of K. For cadlag real-valued seimimartingales X, Y let $[X, Y](t)$ be defined as the limit of sums

$$\sum_{i=0}^{n-1} (X(t_{i+1}) - X(t_i))(Y(t_{i+1} - Y(t_i)) \tag{13}$$

where $0 = t_0 < t_1 < \cdots < t_n = t$ and the limit is in probability as the mesh size goes to zero. If X is an $\mathbb{R}^D$-valued semimartingale, we write

$$[X] = \sum_{i=1}^{D} [X_i, X_i]. \tag{14}$$

Condition C.1. *For every $T > \epsilon > 0$ and compact $K \subset U$*

$$\inf_{0 \leq t \leq T \wedge \lambda_n(K) - \epsilon)} (A_n(t + \epsilon) - A_n(t)) \to \infty \tag{15}$$

as $n \to \infty$ where the infimum of the empty set is taken to be ∞.

Condition C.2. *For every compact $K \subset U$ $\{Z_n^{\lambda_n(K)}\}$ satisfies the following: For $n \geq 1$ let Z_n be a $\{\mathcal{F}_t^n\}$–semimartingale with sample paths in $D_{\mathbb{R}^d}[0, \infty)$. Assume that for some $\delta > 0$ allowing $\delta = \infty$ and every $n \geq 1$, there exist stopping times $\{\tau_n^k | k \geq 1\}$ and a decomposition of $Z_n - J_\delta(Z_n)$ into a local martingale M_n plus a finite variation process F_n such that $P[\tau_n^k \leq k] \leq 1/k$, $\{[M_n](t \wedge \tau_n^k) + T_{t \wedge \tau_n^k}(F_n) | n \geq 1\}$ is uniformly integrable for every $t \geq 0$ and $k \geq 1$ and*

$$\lim_{\gamma \to 0} \limsup_{n \to \infty} P\left[\sup_{0 \leq t \leq T} (T_{t+\gamma}(F_n) - T_t(F_n)) > \epsilon\right] = 0 \tag{16}$$

for every $\epsilon > 0$ and $T > 0$. Also as $n \to \infty$ and for any $T > 0$

$$\sup_{0 < t \leq T \wedge \lambda_n(K)} |\Delta Z_n(t)| \to 0 \tag{17}$$

Condition C.3. *The process*

$$\bar{Z}_n(t) = \sum_{0 < s \leq t} \Delta Z_n(s) \Delta A_n(s) \tag{18}$$

exists, is an $\{\mathcal{F}_t^n\}$–semimartingale, and for every compact $K \subset U$, the sequence $\{\bar{Z}_n^{\lambda_n(K)}\}$ is relatively compact and satisfies Condition 4.1 in Katzenberger (1991).

C.1 An alternative notation for Eq. (1)

Here we show that Eq. (1), i.e.

$$\pi_{k+1} = \beta \pi_k - \nabla L(w_k) + \epsilon \sigma(w_k) \xi_k, \qquad w_{k+1} = w_k + \eta \pi_{k+1}, \tag{19}$$

can be rewritten in the form

$$X_n(t) = X_n(0) + \int_0^t \tilde{\sigma}(X_n) dZ_n + \int_0^t F(X_n) dA_n \,, \tag{20}$$

$$A_n(t) = \left\lfloor \frac{t}{\epsilon_n^2} \right\rfloor, \qquad Z_n(t) = \epsilon_n \sum_{k=1}^{A_n(t)} \xi_k \,. \tag{21}$$

This is how the stochastic process we are interested in is presented in Katzenberger (1991), on which our theory is based. Here $X_n\left(t = \epsilon_n^2 k\right) = (\pi_k, w_k)$, with the correspondence $t = k\epsilon_n^2$, where k denotes the SGD time step. Also, $\tilde{\sigma}(X) = (\sigma, \eta\sigma)$ and $F(X) = ((\beta - 1)\pi - \nabla L(w), \eta(\beta\pi - \nabla L(w)))$. $\lfloor x \rfloor$ denotes the integer part of a real number x.

Before we begin the proof, we will need the following fact (see Sec. 2 of Katzenberger (1991)):

$$\int_s^t f dg = \int_s^t f dg^c + \sum_{s<r\leq t} f(r-)\Delta g(r)\,, \tag{22}$$

where f and g are càdlàg functions, in particular they are right-continuous and have left limits everywhere. The integral above is done with respect to the measure dg, the differential of a function g. The sum is taken over all $r \in (s,t]$ where g is discontinuous and the notation $\Delta g(r) = g(r) - g(r-)$ indicates the discontinuity of g at r, where $g(r-) \equiv \lim_{u\to r^-} g(u)$ indicates the left limit of g at r. Finally g^c denotes the continuous part of g

$$g^c(t) = g(t) - \sum_{0<s\leq t} \Delta g(s), \qquad t \geq 0. \tag{23}$$

We now show by induction that $X_n(t = k\epsilon_n^2)$ solves the first equation above. Note that

$$dA_n(t) = \sum_{k=-\infty}^{\infty} \delta(t - \epsilon_n^2 k), \qquad dZ_n(t) = \epsilon_n \sum_{k=\infty}^{\infty} \xi_k \delta(t - \epsilon_n^2 k)\,. \tag{24}$$

For brevity we will drop the subscript n and let $k\epsilon^2 = t$. Consider

$$\begin{aligned} X(t+\epsilon^2) &= \int_0^{t+\epsilon^2} \tilde\sigma(X(s))dZ(s) + \int_0^{t+\epsilon^2} F(X(s))dA(s) &(25)\\ &= X(t) + \int_t^{t+\epsilon^2} \tilde\sigma(X(s))dZ(s) + \int_t^{t+\epsilon^2} F(X(s))dA(s) &(26)\\ &= X(t) + \sum_{t<s\leq t+\epsilon^2} \tilde\sigma(X(s-))\Delta Z(s) + \sum_{t<s\leq t+\epsilon^2} F(X(s-))\Delta A(s) &(27)\end{aligned}$$

where in the last step we use eq. (22) as $A, Z, \tilde\sigma$ and F are càdlàg, and that $dA^c = dZ^c = 0$. The sums are taken over all $s \in (k\epsilon^2, (k+1)\epsilon^2]$ where $Z(s)$ and $A(s)$ are discontinuous. The only point of discontinuity of A and Z in this interval is at $s = t + \epsilon^2$, so

$$X(t+\epsilon^2) = X(t) + \epsilon\tilde\sigma(X((t+\epsilon^2)-))\xi_{k+1} + F(X((t+\epsilon^2)-)) \tag{28}$$

because the jumps of Z and A at $s = t+\epsilon^2$ are $\epsilon\xi_{k+1}$ and 1, respectively. Now we must determine the left limit of X at $t+\epsilon^2$. Notice that for $0 < \delta < \epsilon$

$$\begin{aligned} X(t+\delta^2) &= X(t) + \sum_{t<s\leq t+\delta^2} \tilde\sigma(X(s-))\Delta Z(s) + \sum_{t<s\leq t+\delta^2} F(X(s-))\Delta A(s) &(29)\\ &= X(t) &(30)\end{aligned}$$

because A and Z are continuous on $(t, t+\delta^2]$. Hence the left limit of $X((t+\epsilon^2)-) = X(t)$. Putting these equations together we find that

$$X((k+1)\epsilon^2) = X(t+\epsilon^2) = X(k\epsilon^2) + \epsilon\tilde\sigma(X(k\epsilon^2))\xi_{k+1} + F(X(k\epsilon^2)) \tag{31}$$

which, using the definition of $X(k\epsilon^2) = (\pi_k, w_k)$, is eq. (19), thus proving equivalence.

C.2 Main theorem from Katzenberger (1991)

Consider the process in Eq. (20). Note that, while at initialization we can have $X_n(0) \notin \Gamma$, the solution $X_n(t) \to \Gamma$ as $n \to \infty$, i.e. it becomes discontinuous. This is an effect of the speed-up of time $t = \epsilon_n^2 k$ introduced below Eq. (20). To overcome this issue, it is convenient to introduce $Y_n(t) \equiv X_n(t) - \psi(X_n(0), A_n(t)) + \Phi(X_n(0))$, so that $Y_n(0) \in \Gamma$ is initialized on the manifold.

Theorem C.4 (Theorem 7.3 in Katzenberger (1991))**.** *Assume that Γ is C^2 and for every $y \in \Gamma$, the matrix $\partial F(y)$ has $D - M$ eigenvalues in $D(1)$. Assume (C.1),(C.2) and (C.3) hold, Φ is C^2 (or F is LC^2) and $X_n(0) \Rightarrow X(0) \in U$. Let*

$$Y_n(t) = X_n(t) - \psi(X(0), A(t)) + \Phi(X(0)) \tag{32}$$

and, for a compact $K \subset U$, let

$$\mu_n(K) = \mathit{inf}\{t \geq 0 | Y_n(t-) \notin \mathring{K} \text{ or } Y_n(t) \notin \mathring{K}\}\,. \tag{33}$$

Then for every compact $K \subset U$, the sequence $\{(Y_n^{\mu_n(K)}, Z_n^{\mu_n(K)}, \mu_n(K))\}$ is relatively compact in $D_{\mathbb{R}^{2D\times r}}[0,\infty) \times [0,\infty]$ (see Katzenberger (1991) for details about the topology). If (Y, Z, μ) is a limit of this sequence then (Y, Z) is a continuous semimartingale, $Y(t) \in \Gamma$ for every t almost surely, $\mu \geq \mathit{inf}\{t \geq 0 | Y(t) \notin \mathring{K}\}$ almost surely, and

$$Y(t) = Y(0) + \int_0^{t\wedge\mu} \partial\Phi(Y)\tilde{\sigma}(Y)dZ + \frac{1}{2}\sum_{ijkl}\int_0^{t\wedge\mu} \partial_{ij}\Phi(Y)\tilde{\sigma}^{ik}(Y)\tilde{\sigma}^{jl}(Y)d[Z^k, Z^l]\,. \tag{34}$$

C.3 Applying Theorem C.4

Recall the equations of motion of stochastic gradient descent

$$\pi_{k+1}^{(n)} = \beta\pi_k^{(n)} - \nabla L(w_k^{(n)}) + \epsilon_n\sigma(w_k^{(n)})\xi_k, \qquad w_{k+1}^{(n)} = w_k^{(n)} + \eta\pi_{k+1}^{(n)}, \tag{35}$$

where $\sigma(w) \in \mathbb{R}^{D\times r}$ is the noise function evaluated at $w \in \mathbb{R}^D$, and $\xi_k \in \mathbb{R}^r$ is a noise vector drawn i.i.d. at every timestep k with zero mean and unit variance. We now show that this equation satisfies all the properties required by Theorem C.4.

The manifold Γ is the fixed point manifold of (non-stochastic) gradient descent. $\{\mathbf{0}\} \times \Gamma$ is a $\mathcal{C}^2$ manifold because Γ is $\mathcal{C}^2$, which follows from assumption 3.1. The flow $F(w, \pi) = (\eta(\beta\pi - \nabla L(w)), \beta\pi - \nabla L(w))$. As shown in Appendix G, dF has exactly M zero eigenvalues on $\Gamma \cap K$. F inherits the differentiable and locally Lipschitz properties from ∇L, and therefore satisfies the conditions of C.4.

Next, notice that the noise function $\tilde{\sigma} : R^{2D} \to \mathbb{R}^{2D\times r}$ is continuous because σ is.

Now we define A_n and Z_n (as in the main text) so that X_n reproduces the dynamics in equation (35), except with a new time parameter $t = k\epsilon_n^2$.

$$A_n(t) = \left\lfloor \frac{t}{\epsilon_n^2} \right\rfloor, \qquad Z_n(t) = \epsilon_n \sum_{k=1}^{A_n(t)} \xi_k \tag{36}$$

So that, with these choices, Eq. (11) precisely corresponds to (35), up to the rescaling $t = k\epsilon_n^2$.

Now we show that A_n, Z_n satisfy the conditions of C.4. Clearly $A_n(0) = Z_n(0)$ by definition. Then

$$A_n(t+\varepsilon) - A_n(t) = \left\lfloor \frac{t+\varepsilon}{\epsilon_n^2} \right\rfloor - \left\lfloor \frac{t}{\epsilon_n^2} \right\rfloor \geq \frac{t+\varepsilon-t}{\epsilon_n^2} - 2 = \frac{\varepsilon}{\epsilon_n^2} - 2 \to \infty \tag{37}$$

when we take $\epsilon_n \to 0$, thus recoverying condition C.1.

By definition Z_n is a martingale. Notice also by the definition of Z_n, because ξ_k is i.i.d. with variance 1, that $Z_n(t)$ has variance $A_n(t)\epsilon_n^2 \leq t$ which is uniformly bounded and hence $Z_n(t)$ is uniformly integrable for stopping times $\tau_n^k = 2k > k$. Also note that $\left|\Delta Z_n(k\epsilon_n^2)\right| = |\epsilon_n\xi_k|$ which goes to zero in probability as ϵ becomes small because ξ_k has bounded variance, and $\Delta Z_n(t)$ is zero otherwise. This shows that Z_n satisfies condition C.2. Because Z_n and A_n are discontinuous at the same time we automatically satisfy condition C.3 as pointed out by Katzenberger.

This shows that we satisfy the conditions of Theorem C.4, therefore we have the following

Lemma C.5. *The SGD equations formulated as in (35) satisfy all the conditions of Theorem C.4.*

D Explicit expression of limiting diffusion in momentum SGD

In this section we provide the proof of Theorem 3.4. Recall that Φ satisfies

$$\Phi(x + F(x)) = \Phi(x)\,, \tag{38}$$

where $x = (\pi, w)$, and F given in (130). To obtain the explicit expression of the limiting drift-diffusion, according to Theorem (C.4), and keeping into account Assumption (3.2), we need to determine Φ up to its second derivatives. To this aim, we shall expand Eq. (38) up to second order in the series expansion in F. In components, this reads:

$$\begin{gathered}\eta\partial_{w_i}\Phi^j(\pi^i - g^i) + \partial_{\pi_i}\Phi^j(-C\eta^\gamma\pi^i - g^i) + \frac{1}{2}\eta^2\partial_{w_i}\partial_{w_k}\Phi^j(\pi^i - g^i)(\pi^k - g^k) \\ +\frac{1}{2}\eta\partial_{w_i}\partial_{\pi_k}\Phi^j(\pi^i - g^i)(-C\eta^\gamma\pi^k - g^k) + \frac{1}{2}\eta\partial_{\pi_i}\partial_{w_k}\Phi^j(-C\eta^\gamma\pi^i - g^i)(\pi^k - g^k) \\ +\frac{1}{2}\partial_{\pi_i}\partial_{\pi_k}\Phi^j(-C\eta^\gamma\pi^i - g^i)(-C\eta^\gamma\pi^k - g^k) = 0\end{gathered} \tag{39}$$

subject to the boundary condition

$$\Phi(\pi, w)|_{w\in\Gamma, \pi=0} = w\,. \tag{40}$$

Here, $g^i = \partial_i L$. In Eq. (39), we already substituted $\beta = 1 - C\eta^\gamma$. In what follows, we shall find Φ to leading order in η as $\eta \to 0$. We will solve the above problem by performing a series expansion in Φ around a point $\bar{w} \in \Gamma$ and $\pi = 0$ up to second order:

$$\Phi(\pi, \bar{w} + \delta w) = \Phi_{00} + \Phi^i_{01}\delta w^i + \Phi^i_{10}\pi^i + \frac{1}{2}\Phi^{ij}_{02}\delta w^i\delta w^j + \Phi^{ij}_{11}\pi^i\delta w^j + \frac{1}{2}\Phi^{ij}_{20}\pi^i\pi^j + \cdots\,. \tag{41}$$

For example, we have

$$\Phi^i_{01} = \frac{\partial\Phi}{\partial w^i}, \qquad \Phi^{ij}_{11} = \frac{\partial^2\Phi}{\partial\pi^i\partial w^j}, \qquad \Phi^{ij}_{20} = \frac{\partial^2\Phi}{\partial\pi^i\partial\pi^j}\,. \tag{42}$$

More precisely, we regard this as an expansion in powers of δw and π: Φ_{00} is zeroth order, Φ_{10}, Φ_{01} are first order, and the remaining terms are second order. We will occasionally write explicitly the index of Φ, e.g. $\Phi^{k,ij}_{02} = \frac{\partial\Phi^k}{\partial w^i\partial w^j}$. It will be useful to introduce the longitudinal projector onto Γ, $P_L(w) : \mathbb{R}^D \to T_w\Gamma$, defined such that for any vector $v \in T_w\Gamma$: $P_L v = v$. The transverse projector is then $P_T = \mathrm{Id} - P_L$. We will also decompose various tensors using these projectors, e.g.

$$\Phi^{ij}_{11} = \Phi^{ij}_{11LL} + \Phi^{ij}_{11LT} + \Phi^{ij}_{11TL} + \Phi^{ij}_{11TT}\,, \tag{43}$$

with $\Phi^{ij}_{11LT} = \Phi^{kl}_{11}P^{ik}_L P^{jl}_T$. Note that the Hessian $H = \nabla g \in \mathbb{R}^{D\times D}$ satisfies $HH^\dagger = P_T$, where $H^\dagger$ denotes the pseudoinverse.

At zeroth order, we obviously have $\Phi_{00}(w) = w$.

Lemma D.1. *The first order terms in the series expansion (41) are given by*

$$\Phi_{01T} = \Phi_{10T} = 0, \qquad \Phi_{01L} = P_L, \qquad \Phi_{10L} = C^{-1}\eta^{1-\gamma}P_L\,. \tag{44}$$

Proof. Suppose $\hat{w}(s) \in \mathbb{R}^D$ is a curve lying on Γ. Then due to the boundary condition (40), $\partial_s\Phi(\pi = 0, \hat{w}) = \Phi^i_{01}\partial_s\hat{w}^i = \partial_s\hat{w}^i$. This means that $\Phi_{01L} = P_L$. Now from (39),

$$\eta\Phi^i_{01}(\pi^i - \partial_j g^i\delta w^j) + \Phi^i_{10}(-C\eta^\gamma\pi^i - \partial_j g^i\delta w^j) = 0\,. \tag{45}$$

This condition should hold for any π and δw, therefore we arrive at

$$\Phi_{10} = C^{-1}\eta^{1-\gamma}\Phi_{01} \tag{46}$$

and

$$(\eta\Phi^i_{01} + \Phi^i_{10})\partial_k g^i = 0\,. \tag{47}$$

Decomposing into longitudinal and transverse components, and noting that the Hessian satisfies $P_L H = 0$, the above equation becomes

$$\eta\Phi^i_{01T} + \Phi^i_{10T} = 0\,, \tag{48}$$

which together with (46) and the above discussion gives

$$\Phi_{01T} = \Phi_{10T} = 0, \qquad \Phi_{01L} = P_L, \qquad \Phi_{10L} = C^{-1}\eta^{1-\gamma}P_L\,, \tag{49}$$

which concludes the first order analysis. □

We now need a Lemma, which requires Definition (3.3) in the main text. We report it here for convenience:

Definition D.2. For a symmetric matrix $H \in \mathbb{R}^D \times \mathbb{R}^D$, and $W_H = \{\Sigma \in \mathbb{R}^D \times \mathbb{R}^D : \Sigma = \Sigma^\top, HH^\dagger \Sigma = H^\dagger H \Sigma = \sigma\}$, we define the operator $\tilde{\mathcal{L}}_H : W_H \to W_H$ with $\tilde{\mathcal{L}}_H S \equiv \{H, S\} + \frac{1}{2} C^{-2} \eta^{1-2\gamma} [[S, H], H]$, with $[S, H] = SH - HS$.

Lemma D.3. *The inverse of the operator $\tilde{\mathcal{L}}_H$ is unique.*

Proof. Let us go to a basis where H is diagonal, with eigenvalues λ_i. In components, the equation $\tilde{\mathcal{L}}_H S = M$ reads

$$(\lambda_i + \lambda_j) S_{ij} + \frac{1}{2} C^{-2} \eta^{1-2\gamma} (\lambda_i - \lambda_j)^2 S_{ij} + C^{-1} \eta^{1-\gamma} M_{ij} = 0 \tag{50}$$

which has a unique solution, with

$$S_{ij} = -C^{-1} \eta^{1-\gamma} \left(\lambda_i + \lambda_j + \frac{1}{2} C^{-2} \eta^{1-2\gamma} (\lambda_i - \lambda_j)^2 \right)^{-1} M_{ij} \,. \tag{51}$$

□

Lemma D.4. *The second order terms in the series expansion (41) are given by*

$$\Phi^{j,ik}_{02LL} = -(H^\dagger)^{jl} \partial^L_i H^{ln} P^{nk}_L , \quad \Phi^{j,ik}_{02LT} = -P^{jl}_L \partial^L_i H^{ln} (H^\dagger)^{nk} , \quad \Phi^{j,ik}_{02TT} = O\left(\eta^{min\{0,1-2\gamma\}} \right) \tag{52}$$

$$\Phi^{j,ik}_{11LL} = -C^{-1} \eta^{1-\gamma} (H^\dagger)^{jl} \partial^L_i H^{ln} P^{nk}_L , \quad \Phi^{j,ik}_{11TL} = -C^{-1} \eta^{1-\gamma} P^{jl}_L \partial^L_k H^{ln} (H^\dagger)^{ni} \tag{53}$$

$$\Phi^{j,ik}_{11TT} = -C^{-1} \eta^{1-\gamma} \tilde{\mathcal{L}}^{-1}_H (M^{(j)})_{ik} - \frac{1}{2} C^{-3} \eta^{2-3\gamma} [H, \tilde{\mathcal{L}}^{-1}_H M^{(j)}]_{ik} \tag{54}$$

$$\Phi^{j,ik}_{20LL} = -\frac{1}{2} C^{-2} \eta^{2-2\gamma} \left((H^\dagger)^{jl} \partial^L_i H^{ln} P^{nk}_L + (H^\dagger)^{jl} \partial^L_k H^{ln} P^{ni}_L \right) \tag{55}$$

$$\Phi^{j,ik}_{20TL} = -\frac{1}{2} C^{-2} \eta^{2-2\gamma} \left(P^{jl}_L \partial^L_k H^{ln} (H^\dagger)^{ni} + P^{jl}_L \partial^L_i H^{ln} (H^\dagger)^{nk} \right) \tag{56}$$

$$\Phi^{j,ik}_{20TT} = -C^{-2} \eta^{2-2\gamma} \tilde{\mathcal{L}}^{-1}_H (P_T \partial^L_j H P_T)_{ik} \tag{57}$$

where

$$(M^{(j)})_{kl} = P^{ki}_T \partial^L_j H_{ni} P^{nl}_T \tag{58}$$

Proof. Consider a path $\hat{w}(s)$ lying on Γ. From $P_T H = H$ we have

$$\frac{dP_T}{ds} H = (\text{Id} - P_T) \frac{dH}{ds} = P_L \frac{dH}{ds} \,, \tag{59}$$

and thus, using $HH^\dagger = P_T$, we find

$$\frac{dP_T}{ds} P_T = P_L \frac{dH}{ds} H^\dagger , \qquad P_T \frac{dP_T}{ds} = H^\dagger \frac{dH}{ds} P_L \,, \tag{60}$$

where the second equation is obtained from the first by taking the transpose. Putting the last two relations together, we find

$$\frac{dP_L}{ds} = -\frac{dP_T}{ds} = -P_T \frac{dP_T}{ds} - \frac{dP_T}{ds} P_T = -H^\dagger \frac{dH}{ds} P_L - P_L \frac{dH}{ds} H^\dagger \,. \tag{61}$$

From (44) we can then write

$$\partial^L_i \Phi^{j,k}_{01} = -(H^\dagger)^{jl} \partial^L_i H^{ln} P^{nk}_L - P^{jl}_L \partial^L_i H^{ln} (H^\dagger)^{nk} \,, \tag{62}$$

where $\partial^L_i = P^{ij}_L \partial_j$. This leads to

$$\Phi^{j,ik}_{02LL} = P^{kl}_L \partial^L_i \Phi^{j,l}_{01} = -(H^\dagger)^{jl} \partial^L_i H^{ln} P^{nk}_L \tag{63}$$

$$\Phi^{j,ik}_{02LT} = P^{kl}_T \partial^L_i \Phi^{j,l}_{01} = -P^{jl}_L \partial^L_i H^{ln} (H^\dagger)^{nk} \,. \tag{64}$$

Also note that $\Phi_{02LT}^{j,ik} = \Phi_{02TL}^{j,ki}$, so the only component still to be determined in Φ_{02} is Φ_{02TT}.

The next step is to expand Eq. (39) to second order:

$$\begin{aligned}
&\pi^i\pi^k\left(\eta(1+C\eta^\gamma)\Phi_{11}^{ki} - C\eta^\gamma(1-\tfrac{1}{2}C\eta^\gamma)\Phi_{20}^{ik} + \tfrac{1}{2}\eta^2\Phi_{02}^{ik}\right)\\
&+\delta w^k\pi^i\left(\eta\Phi_{02}^{ik} - \eta^2 H_{kj}\Phi_{02}^{ji} - \eta\Phi_{11}^{il}H_{kl} - C\eta^\gamma\Phi_{11}^{ik} - \eta(1-C\eta^\gamma)\Phi_{11}^{(ij)}H_{kj} - (1-C\eta^\gamma)\Phi_{20}^{li}H_{kl}\right)\\
&-\delta w^k\delta w^l\Big(\eta\Phi_{02}^{ik}H_{li} + \tfrac{1}{2}\eta^2\Phi_{02}^{ji}H_{lj}H_{ki} + \Phi_{11}^{ik}H_{li} + \eta\Phi_{11}^{ji}H_{kj}H_{li}\\
&\qquad\qquad + \tfrac{1}{2}(\eta\Phi_{01}^{i} + \Phi_{10}^{i})\partial_k H_{li} + \tfrac{1}{2}\Phi_{20}^{ji}H_{lj}H_{ki}\Big) = 0\,,
\end{aligned} \tag{65}$$

where $A^{(ij)} = \frac{1}{2}(A^{ij} + A^{ji})$ denotes the symmetric part. Neglecting various terms that are subleading at small η, gives

$$\begin{aligned}
&\pi^i\pi^k\left(\eta\Phi_{11}^{ki} - C\eta^\gamma\Phi_{20}^{ik} + \tfrac{1}{2}\eta^2\Phi_{02}^{ik}\right)\\
&+\delta w^k\pi^i\left(\eta\Phi_{02}^{ik} - C\eta^\gamma\Phi_{11}^{ik} - \Phi_{20}^{li}H_{kl}\right)\\
&-\delta w^k\delta w^l\left(\eta\Phi_{02}^{ik}H_{li} + \Phi_{11}^{ik}H_{li} + \tfrac{1}{2}(\eta\Phi_{01}^{i} + \Phi_{10}^{i})\partial_k H_{li} + \tfrac{1}{2}\Phi_{20}^{ji}H_{lj}H_{ki}\right) = 0\,,
\end{aligned} \tag{66}$$

The first line immediately gives

$$\Phi_{20}^{ij} = C^{-1}\eta^{1-\gamma}\Phi_{11}^{(ij)} + \frac{1}{2}C^{-1}\eta^{2-\gamma}\Phi_{02}^{ij}\,. \tag{67}$$

Taking the second line of (66) and projecting onto the longitudinal part the index k gives

$$\eta\beta\Phi_{02LL}^{ik} - C\eta^\gamma\Phi_{11LL}^{ik} = 0 \tag{68}$$

$$\eta\beta\Phi_{02TL}^{ik} - C\eta^\gamma\Phi_{11TL}^{ik} = 0 \tag{69}$$

and using (63),(64),

$$\Phi_{11LL}^{j,ik} = C^{-1}\eta^{1-\gamma}\Phi_{02LL}^{j,ik} = -C^{-1}\eta^{1-\gamma}(H^\dagger)^{jl}\partial_i^L H^{ln}P_L^{nk} \tag{70}$$

$$\Phi_{11TL}^{j,ik} = C^{-1}\eta^{1-\gamma}\Phi_{02TL}^{j,ik} = -C^{-1}\eta^{1-\gamma}P_L^{jl}\partial_k^L H^{ln}(H^\dagger)^{ni} \tag{71}$$

Projecting the second line of (66) on the transverse part of the index k we have

$$\eta\Phi_{02LT}^{j,ik} - C\eta^\gamma\Phi_{11LT}^{j,ik} - \Phi_{20TL}^{j,li}H_{kl} = 0 \tag{72}$$

$$\eta\Phi_{02TT}^{j,ik} - C\eta^\gamma\Phi_{11TT}^{j,ik} - \Phi_{20TT}^{j,li}H_{kl} = 0 \tag{73}$$

Using (67) and keeping (64) into account, Eq. (72) becomes, neglecting subleading terms in η,

$$-\eta P_L^{jl}\partial_i^L H^{ln}(H^\dagger)^{nk} - C\eta^\gamma\Phi_{11LT}^{ik} - \frac{1}{2}C^{-1}\eta^{1-\gamma}\Phi_{11TL}^{li}H_{kl} - \frac{1}{2}C^{-1}\eta^{1-\gamma}\Phi_{11LT}^{il}H_{kl} = 0\,. \tag{74}$$

Using (71) in (74),

$$\begin{aligned}
-\eta P_L^{jl}\partial_i^L H^{ln}(H^\dagger)^{nk} - C\eta^\gamma\Phi_{11LT}^{ik}\\
+\frac{1}{2}C^{-2}\eta^{2-2\gamma}P_L^{jp}\partial_i^L H^{pn}(H^\dagger)^{nl}H_{kl} - \frac{1}{2}C^{-1}\eta^{1-\gamma}\Phi_{11LT}^{il}H_{kl} = 0\,,
\end{aligned} \tag{75}$$

and simplifying,

$$\begin{aligned}
-\eta P_L^{jl}\partial_i^L H^{ln}(H^\dagger)^{nk} - C\eta^\gamma\Phi_{11LT}^{ik}\\
+\frac{1}{2}C^{-2}\eta^{2-2\gamma}P_L^{jl}\partial_i^L H^{ln}P_T^{nk} - \frac{1}{2}C^{-1}\eta^{1-\gamma}\Phi_{11LT}^{il}H_{kl} = 0\,.
\end{aligned} \tag{76}$$

which determines Φ_{11LT} in close form. Indeed, the above has the form, in matrix notation

$$-\frac{1}{2}C^{-1}\eta^{1-\gamma}\Phi_{11LT}H - C\eta^\gamma\Phi_{11LT} = M\,, \tag{77}$$

and can be immediately inverted to solve for Φ_{11LT}. The only two undetermined components now are Φ_{11TT} and Φ_{02TT}. One condition is obtained from (73) which gives, keeping (67) into account,

$$\eta\Phi^{j,ik}_{02TT} - C\eta^{\gamma}\Phi^{j,ik}_{11TT} - C^{-1}\eta^{1-\gamma}\Phi^{j,(li)}_{11TT}H_{kl} = 0\,, \tag{78}$$

and thus

$$\Phi^{j,ik}_{02TT} = C\eta^{\gamma-1}\Phi^{j,ik}_{11TT} + C^{-1}\eta^{-\gamma}\Phi^{j,(ni)}_{11TT}H_{kn}\,. \tag{79}$$

Further taking symmetric and antisymmetric part in ik of the above gives

$$\Phi^{j,ik}_{02TT} = C\eta^{\gamma-1}\Phi^{j,(ik)}_{11TT} + \frac{1}{2}C^{-1}\eta^{-\gamma}\Phi^{j,(ni)}_{11TT}H_{kn} + \frac{1}{2}C^{-1}\eta^{-\gamma}\Phi^{j,(nk)}_{11TT}H_{in} \tag{80}$$

$$0 = C\eta^{\gamma-1}\Phi^{j,[ik]}_{11TT} + \frac{1}{2}C^{-1}\eta^{-\gamma}\Phi^{j,(ni)}_{11TT}H_{kn} - \frac{1}{2}C^{-1}\eta^{-\gamma}\Phi^{j,(nk)}_{11TT}H_{in}\,. \tag{81}$$

The other condition comes from the third line of (66) which, using (44) and (67), and neglecting subleading terms in η, gives

$$\eta\Phi^{j,ik}_{02}H_{li} + \eta\Phi^{j,il}_{02}H_{ki} + \Phi^{j,ik}_{11}H_{li} + \Phi^{j,il}_{11}H_{ki} + C^{-1}\eta^{1-\gamma}P^{ji}_{L}\partial_k H_{li} = 0 \tag{82}$$

Projecting the k and l indices on the longitudinal part, gives $P^{ji}_{L}\partial^{L}_{k}H_{li}P^{ln}_{L} = 0$, which is an identity that can be checked from (59). Projecting k on the longitudinal part and l on the transverse part gives

$$\eta\Phi^{j,ik}_{02TL}H_{li} + \Phi^{j,ik}_{11TL}H_{li} + C^{-1}\eta^{1-\gamma}P^{ji}_{L}\partial^{L}_{k}H_{ni}P^{nl}_{T} = 0\,, \tag{83}$$

which is implied by (71), indeed

$$\begin{aligned}(\eta\Phi^{j,ik}_{02TL} + \Phi^{j,ik}_{11TL})H_{li} &= C^{-1}\eta^{1-\gamma}\Phi^{j,ik}_{02TL}H_{li}\\ &= -C^{-1}\eta^{1-\gamma}P^{jl}_{L}\partial^{L}_{k}H^{ln}(H^{\dagger})^{ni}H_{li} = -C^{-1}\eta^{1-\gamma}P^{jl}_{L}\partial^{L}_{k}H^{ln}P^{nl}_{T},\end{aligned} \tag{84}$$

therefore (83) does not give a new condition. The only new condition comes from projecting the k and l indices of (82) on the transverse direction, giving

$$\eta\Phi^{j,ik}_{02TT}H_{li} + \eta\Phi^{j,il}_{02TT}H_{ki} + \Phi^{j,ik}_{11TT}H_{li} + \Phi^{j,il}_{11TT}H_{ki} + C^{-1}\eta^{1-\gamma}P^{ji}_{L}\partial^{T}_{k}H_{ni}P^{nl}_{T} = 0\,. \tag{85}$$

Plugging in (79),

$$\begin{aligned}&C\eta^{\gamma}\Phi^{j,ik}_{11TT}H_{li} + C^{-1}\eta^{1-\gamma}\Phi^{j,(in)}_{11TT}H_{kn}H_{li} + C\eta^{\gamma}\Phi^{j,il}_{11TT}H_{ki} + C^{-1}\eta^{1-\gamma}\Phi^{j,(in)}_{11TT}H_{ln}H_{ki}\\ &\quad + \Phi^{j,ik}_{11TT}H_{li} + \Phi^{j,il}_{11TT}H_{ki} + C^{-1}\eta^{1-\gamma}P^{ji}_{L}\partial^{T}_{k}H_{ni}P^{nl}_{T} = 0\,.\end{aligned} \tag{86}$$

Neglecting subleading terms in η this can be rewritten as

$$2C^{-1}\eta^{1-\gamma}\Phi^{j,(in)}_{11TT}H_{kn}H_{li} + \Phi^{j,ik}_{11TT}H_{li} + \Phi^{j,il}_{11TT}H_{ki} + C^{-1}\eta^{1-\gamma}\partial^{L}_{j}H_{ni}P^{ki}_{T}P^{nl}_{T} = 0\,. \tag{87}$$

where we recall that the last term is symmetric in k and l. In matrix notation this reads

$$C^{-1}\eta^{1-\gamma}H(\Phi^{j}_{11TT} + \Phi^{jt}_{11TT})H + H\Phi^{j}_{11TT} + \Phi^{jt}_{11TT}H + C^{-1}\eta^{1-\gamma}M^{(j)} = 0\,, \tag{88}$$

where

$$(M^{(j)})_{kl} = P^{ki}_{T}\partial^{L}_{j}H_{ni}P^{nl}_{T} \tag{89}$$

is the longitudinal derivative of the Hessian, projected on the transverse directions. Decomposing Φ^{j}_{11TT} into symmetric and antisymmetric parts $\Phi^{j}_{11TT} = S^{(j)} + A^{(j)}$, Eq. (88) reads (suppressing the index j)

$$2C^{-1}\eta^{1-\gamma}HSH + HS + SH + HA - AH + C^{-1}\eta^{1-\gamma}M = 0\,. \tag{90}$$

The first term is subleading in η, therefore we have

$$HS + SH + HA - AH + C^{-1}\eta^{1-\gamma}M = 0\,. \tag{91}$$

This equation together with (81), which in matrix form reads

$$C^{-1}\eta^{-\gamma}(SH - HS) + 2C\eta^{\gamma-1}A = 0\,, \tag{92}$$

determine S and A, and thus Φ_{11TT}. Note that M^j has only a longitudinal component in the index j, therefore the transverse parts of S and A vanish, i.e.

$$P_T^{pj}\,\Phi^{j,ik}_{11TT} = 0\,. \tag{93}$$

Eq. (93) is natural as the slow degrees of freedom are the longitudinal coordinates.

Note that these two equations admit a unique solution. Indeed, solving (92) for A gives

$$A = \frac{1}{2}C^{-2}\eta^{1-2\gamma}[H, S] \tag{94}$$

Plugging in (91), we find

$$\tilde{\mathcal{L}}_H S = -C^{-1}\eta^{1-\gamma}M\,, \tag{95}$$

where $\tilde{\mathcal{L}}_H S \equiv \{H, S\} + \frac{1}{2}C^{-2}\eta^{1-2\gamma}[[S, H], H]$ is introduced in definition D.2. By Lemma D.3, (95) admits a unique solution.

Then

$$\Phi^j_{11TT} = -C^{-1}\eta^{1-\gamma}\tilde{\mathcal{L}}_H^{-1}M^{(j)} - \frac{1}{2}C^{-3}\eta^{2-3\gamma}[H, \tilde{\mathcal{L}}_H^{-1}M^{(j)}]\,. \tag{96}$$

The other components are (see eqs. (63),(64),(70),(71),(79),(67))

$$\Phi^{j,ik}_{02LL} = -(H^\dagger)^{jl}\partial^L_i H^{ln}P_L^{nk} \tag{97}$$

$$\Phi^{j,ik}_{02LT} = -P_L^{jl}\partial^L_i H^{ln}(H^\dagger)^{nk} \tag{98}$$

$$\Phi^{j,ik}_{11LL} = -C^{-1}\eta^{1-\gamma}(H^\dagger)^{jl}\partial^L_i H^{ln}P_L^{nk} \tag{99}$$

$$\Phi^{j,ik}_{11TL} = -C^{-1}\eta^{1-\gamma}P_L^{jl}\partial^L_k H^{ln}(H^\dagger)^{ni} \tag{100}$$

$$\Phi^{j,ik}_{02TT} = C\eta^{\gamma-1}\Phi^{j,ik}_{11TT} + C^{-1}\eta^{-\gamma}\Phi^{j,(ni)}_{11TT}H_{kn} = O\left(\eta^{\min\{0,1-2\gamma\}}\right) \tag{101}$$

$$\Phi^{ij}_{20} = C^{-1}\eta^{1-\gamma}\Phi^{(ij)}_{11} + \frac{1}{2}C^{-1}\eta^{2-\gamma}\Phi^{ij}_{02} = C^{-1}\eta^{1-\gamma}\Phi^{(ij)}_{11}\,. \tag{102}$$

The only contributing term at leading order in η to the limiting diffusion equation is (102). Splitting it into longitudinal and transverse components, we find:

$$\Phi^{j,ik}_{20LL} = C^{-1}\eta^{1-\gamma}\Phi^{j,(ik)}_{11LL} = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}\left((H^\dagger)^{jl}\partial^L_i H^{ln}P_L^{nk} + (H^\dagger)^{jl}\partial^L_k H^{ln}P_L^{ni}\right) \tag{103}$$

$$\Phi^{j,ik}_{20TL} = C^{-1}\eta^{1-\gamma}\Phi^{j,(ik)}_{11TL} = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}\left(P_L^{jl}\partial^L_k H^{ln}(H^\dagger)^{ni} + P_L^{jl}\partial^L_i H^{ln}(H^\dagger)^{nk}\right) \tag{104}$$

and, using (96) and (89) we have, in matrix notation,

$$\Phi^j_{20TT} = \frac{1}{2}C^{-1}\eta^{1-\gamma}(\Phi^j_{11TT} + (\Phi^j_{11TT})^T) = -C^{-2}\eta^{2-2\gamma}\tilde{\mathcal{L}}_H^{-1}(P_T\partial^L_j H P_T)\,. \tag{105}$$

□

To write things more compactly, the following Lemma will be useful:

Lemma D.5. *For any transverse symmetric matrix T:*

$$\Phi^j_{20TT}[T] = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}M^{(j)}[\tilde{\mathcal{L}}_H^{-1}T]\,, \tag{106}$$

Proof. From (57), Φ_{20TT} is proportional to the symmetric part of Φ_{11TT} which was denoted by S below Eq. (89) and satisfies Eq. (95). Therefore Φ_{20TT} also satisfies Eq. (95), up to an overall factor:

$$\tilde{\mathcal{L}}_H\Phi^j_{20TT} = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}M^{(j)}\,. \tag{107}$$

Then, for any T:

$$(\tilde{\mathcal{L}}_H\Phi^j_{20TT})[T] = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}M^{(j)}[T]\,. \tag{108}$$

Moreover,

$$(\tilde{\mathcal{L}}_H\Phi^j_{20TT})[T] = \Phi^j_{20TT}[\tilde{\mathcal{L}}_H T]\,. \tag{109}$$

Since the two above equations hold for any T and $\tilde{\mathcal{L}}_H$ is invertible, this implies

$$\Phi^j_{20TT}[T] = -\frac{1}{2}C^{-2}\eta^{2-2\gamma}M^{(j)}[\tilde{\mathcal{L}}_H^{-1}T]\,, \tag{110}$$

which is the statement of the lemma. □

To leading order in η we then have, for a symmetric matrix V, $\partial^2\Phi[V] = \sum_{i,j=1}^{D}\tilde{\partial}_i\tilde{\partial}_j\Phi V_{ij}$, and thus

$$\begin{aligned}\partial^2\Phi[V] =& -\frac{1}{2C^2}\eta^{2-2\gamma}(\nabla^2 L)^\dagger\partial^2(\nabla L)[V_{LL}]dt - \frac{1}{C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[(H^\dagger)V_{TL}]dt \\ & - \frac{1}{2C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[\tilde{\mathcal{L}}_H^{-1}V_{TT}]\,,\end{aligned} \tag{111}$$

where $V_{LL} = P_L V P_L$, $V_{TL} = P_T V P_L$, and $V_{TT} = P_T V P_T$ are transverse and longitudinal projections of V.

We also have, using (44), and to leading order in η,

$$\partial\Phi\tilde{\sigma}dZ = \Phi_{10}\sigma dZ + \eta\Phi_{01}\sigma dZ = (C^{-1}\eta^{1-\gamma}+\eta)P_L\sigma dW\,, \tag{112}$$

where dW is a Wiener process. Applying Theorem C.4, and keeping into account that, to leading order in dt, $d[Z^i, Z^j] = \delta^{ij}dt$, we find

$$\begin{aligned}dY =&(C^{-1}\eta^{1-\gamma}+\eta)P_L\sigma dZ - \frac{1}{2C^2}\eta^{2-2\gamma}(\nabla^2 L)^\dagger\partial^2(\nabla L)[\Sigma_{LL}]dt \\ & - \frac{1}{C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[(H^\dagger)\Sigma_{TL}]dt - \frac{1}{2C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[\tilde{\mathcal{L}}_H^{-1}\Sigma_{TT}]dt\,,\end{aligned} \tag{113}$$

where $\Sigma = \sigma\sigma^T$.

For $\gamma < \frac{1}{2}$, $\tilde{\mathcal{L}}_H$ reduces to the Lyapunov operator at leading order in η, i.e. $\tilde{\mathcal{L}}_H S = \{H, S\}$. For $\gamma > \frac{1}{2}$, from (51), it is easy to see that the role of the divergent term proportional to $\eta^{1-2\gamma}$, when acting $\tilde{\mathcal{L}}_H^{-1}$ on S, is to set to zero the off-diagonal entries of S_{ij} at $O(\eta^{1-\gamma})$, i.e.

$$S_{ii} = -2C^{-1}\eta^{1-\gamma}\lambda_i M_{ii}, \qquad S_{i\neq j} = 0,. \tag{114}$$

Using Lemma C.5, we finally conclude the following Corollary, which is the formal version of Theorem 3.4 in the main text:

Corollary D.6. *Consider the stochastic process defined in Eq. (35) parametrized by ϵ_n, with initial conditions $(\pi_0, w_0) \in U$, under assumptions 3.1 and 3.2. Fix a compact $K \subset U$, and that $X_n(0) = X(0) \in U$. Then the conclusions of Theorem C.4 apply, and $Y(t)$ satisfies the limiting diffusion equation*

$$\begin{aligned}dY =&(\tfrac{1}{C}\eta^{1-\gamma}+\eta)P_L\sigma dW - \tfrac{1}{2C^2}\eta^{2-2\gamma}(\nabla^2 L)^\dagger\partial^2(\nabla L)[\Sigma_{LL}]dt \\ & - \tfrac{1}{C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[(\nabla^2 L)^\dagger\Sigma_{TL}]dt - \tfrac{1}{2C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[\tilde{\mathcal{L}}_{\nabla^2 L}^{-1}\Sigma_{TT}]dt\,,\end{aligned} \tag{115}$$

where $W(t)$ is a Wiener process.

Let us now see the special case of label noise. In this case $\Sigma = cH$, so that Σ is only transverse. Moreover, using (51),

$$\tilde{\mathcal{L}}_H^{-1}H = \frac{1}{2}P_T\,, \tag{116}$$

and

$$dY = -\frac{1}{4C^2}\eta^{2-2\gamma}P_L\partial^2(\nabla L)[cP_T]dt = -\frac{1}{4C^2}\eta^{2-2\gamma}P_L\nabla\mathrm{Tr}(c\partial^2 L)dt\,. \tag{117}$$

This proves Corollary 3.5 in the main text.

E A SOLVABLE EXAMPLE

In this section we analyse a model that will allow us to determine, on top of the optimal exponent $\gamma = \frac{2}{3}$, also the prefactor C. We will specify to a 2-layer linear MLP, which is sufficient to describe the transition in the hierarchy of timescales described above, and is simple enough to exactly compute C. We will show in Sec. 4.1 that C depends only mildly on the activation function. We apply a simple matching principle to determine C, by asking that the deterministic timescale τ_1 is equal to the drift-diffusion timescale τ_2. In the previous section, we found the critical $\gamma = 2/3$ by requiring these timescales have the same scaling in η. In order to determine C, we need more details of the model architecture.

Definition E.1 (UV model)**.** We define the UV model as a 2-layer linear network parametrized by $f(x) = \frac{1}{\sqrt{n}} UVx \in \mathbb{R}^m$, where $x \in \mathbb{R}^d$, $V \in \mathbb{R}^{n\times d}$, and $U \in \mathbb{R}^{m\times n}$. For $d = m = 1$, we refer to this as the **vector UV model** (Rennie & Srebro, 2005; Saxe et al., 2014; Lewkowycz et al., 2020).

For a training dataset $\mathcal{D} = \{(x^a, y^a)\big| a = 1, ..., P\}$, the dataset covariance matrix is $\Sigma_{ij} = \frac{1}{P}\sum_{a=1}^{P} x_i^a x_j^a$, and the dataset variance is $\mu_2 = \mathrm{tr}\Sigma$. For mean-squared error loss, it is possible to explicitly determine the trace of the Hessian (see Appendix F). SGD with label noise introduces $y^a \to y^a + \epsilon\xi_t$ where $\langle \xi_t^2\rangle = 1$, from which we identify $\sigma_{\mu,ja} = P^{-1}\nabla_\mu f_j(x^a)$, where μ runs over all parameter indices, $j \in [m]$, and $a \in [P]$. With this choice, the SGD noise covariance satisfies $\sigma\sigma^T = P^{-1}\nabla^2 L$. Equipped with this, we may use Corollary 3.5 with $c = P^{-1}$ to determine the effective drift (presented in Appendix F). For the vector UV model, the expression simplifies to $dY = -\tau_2^{-1} Y dt$, with

$$\tau_2^{-1} = \frac{\eta^{2-2\gamma}\epsilon^2\mu_2}{2nPC^2} \,. \tag{118}$$

The timescale of the fast initial phase $\tau_1^{-1} = (C/2)\eta^\gamma$ follows from the previous section. Then requiring $\tau_1 = \tau_2$ implies not only $\gamma = 2/3$, but

$$C = \left(\frac{\epsilon^2\mu_2}{Pn}\right)^{1/3} . \tag{119}$$

One particular feature to note here is that C will be small for overparametrized models and/or training with large datasets.

F EFFECTIVE DRIFT IN UV MODEL

We start with mean-square loss

$$L = \frac{1}{2P}\sum_{a=1}^{P} ||f(x^a) - y^a||^2, \tag{120}$$

with the data covariance matrix Σ as defined in the text. The trace of the Hessian on the zero loss manifold ($L(w^*) = 0$) is given explicitly by

$$\mathrm{tr}H = \frac{1}{n}\left(m\mathrm{Tr}\left(\Sigma V^\top V\right) + \mathrm{Tr}\,\Sigma\,\mathrm{Tr}\left(UU^\top\right)\right). \tag{121}$$

Taking gradients of this and plugging into Corollary 3.5, repeated in Eq. (117) leads to an explicit expression for the drift-diffusion along the manifold

$$dY = -\frac{\epsilon^2\eta^{2-2\gamma}}{4PC^2}\frac{2}{n}\hat{S}_L Y dt, \quad \hat{S} = \begin{pmatrix} \mathrm{Tr}\Sigma\,\mathbb{1}_n & 0 \\ 0 & m\Sigma \end{pmatrix}, \quad \hat{S}_L = P_L\hat{S}P_L, \tag{122}$$

The simplification we cite in App. E is due to the fact that for input and output dimension $d = m = 1$, we have that $\Sigma = \mathrm{Tr}\Sigma = \mu_2$, and $\hat{S}$ is proportional to the identity.

For matrix sensing, in order to compute the trace of the Hessian, we use (8) with $\xi_t = 0$ and with slightly different notation ($V^\top$ instead of V). The loss is then

$$L = \frac{1}{Pd}\sum_{i=1}^{P}\left(y_i - \mathrm{Tr}(A_i U V^\top)\right)^2, \tag{123}$$

and define the data covariance matrices

$$\hat{\Sigma}^1 = \frac{1}{P}\sum_{i=1}^{P} A_i A_i^\top \in \mathbb{R}^{d\times d}, \quad \hat{\Sigma}^2 = \frac{1}{P}\sum_{i=1}^{P} A_i^\top A_i \in \mathbb{R}^{d\times d}. \tag{124}$$

Then the trace of the Hessian is

$$\mathrm{Tr}H = \frac{2}{d}\mathrm{Tr}\left(\hat{\Sigma}^2 U U^\top + \hat{\Sigma}^1 V V^\top\right). \tag{125}$$

We find the noise function is

$$\sigma_{\mu i} = \frac{2}{Pd}\nabla_\mu f(A_i) \in \mathbb{R}^{d^4\times P} \tag{126}$$

where $f(A) = \mathrm{Tr}(UV^\top A)$. Since the Hessian on the zero loss manifold is $H_{\mu\nu} = \frac{2}{Pd}\sum_i \nabla_\mu f(A_i)\nabla_\nu f(A_i)$, we see that $\sigma\sigma^T = (2/Pd)H$. Therefore, we get

$$dY = -\frac{\eta^{2-2\gamma}}{4C^2}\frac{4\epsilon^2}{Pd^2}\hat{S}_L Y dt, \quad \hat{S} = \begin{pmatrix} \hat{\Sigma}^2 & 0 \\ 0 & \hat{\Sigma}^1 \end{pmatrix}, \quad \hat{S}_L = P_L \hat{S} P_L \tag{127}$$

To get a crude estimate of the coefficient C in the main text, we approximate the top eigenvalue of $\hat{S}$ with $\frac{1}{d}\mathrm{Tr}\hat{\Sigma}^1$. With this, we get for

$$\eta^{-2+2\gamma}\tau_2^{-1} = \frac{\epsilon^2}{C^2Pd^2}\frac{1}{dP}\sum_{i=1}^{P}\mathrm{Tr}A_iA_i^\top \approx \frac{\epsilon^2}{C^2Pd^2}d\langle a_{ij}^2\rangle = \frac{\epsilon^2}{C^2dP}\langle a_{ij}^2\rangle. \tag{128}$$

Here we have denoted by a_{ij} an arbitrary element of the data matrices A, with brackets signifying an average over the distribution of these elements. Assuming the fast initial phase remains the same, so that $\tau_1^{-1} \approx (C/2)\eta^\gamma$, we get

$$C^3 = \frac{2\epsilon^2}{dP}\langle a_{ij}^2\rangle \tag{129}$$

For the values used in our experiments, this gives $C \approx 0.12 \times P^{-1/3}$.

G Linearization Analysis of Momentum Gradient Descent in the Scaling Limit

Here we elaborate on the discussion in Sec. 3.3, providing derivations of various results. We take a straightforward linearization of the deterministic (noise-free) gradient descent with momentum. Working in the extended phase space $x = (\pi, w)$, the dynamical updates are of the form

$$x_{t+1} = x_t + F(x_t), \quad F(x_t) = \begin{pmatrix} (\beta-1)\pi_t - \nabla L(w_t) \\ \eta(\beta\pi_t - \nabla L(w_t)) \end{pmatrix}. \tag{130}$$

The fixed point of the evolution $x^* = (0, w^*)$ will have the momentum variable $\pi = 0$, and the coordinate satisfying $\nabla L(w^*) = 0$. Linearizing the update (130) around this point

$$\delta x_{t+1} = J(x^*)\delta x_t, \quad J(x^*) = \left(\begin{array}{cc} \beta & -\nabla^2 L(w^*) \\ \eta\beta & 1 - \eta\nabla^2 L(w^*) \end{array} \right). \tag{131}$$

Note $\nabla^2 L(w^*)$ is the Hessian of the loss function at the fixed point. The spectrum of the Jacobian can be written in terms of the eigenvalues of the Hessian λ_i. This is accomplished by using a straightforward ansatz for the (unnormalized) eigenvectors of the Jacobian $k^i = (\mu^i q^i, q^i)$, where q^i are eigenvectors of the Hessian with eigenvalue λ_i. Solving the resulting coupled eigenvalue equations for eigenvalue κ^i:

$$1 - \eta\lambda_i + \eta\beta\mu^i = \kappa^i, \quad -\lambda_i + \mu^i\beta = \mu^i\kappa^i. \tag{132}$$

For a fixed λ_i, there will be two solutions given by

$$\kappa^i_\pm = \frac{1}{2}\left(1 + \beta - \eta\lambda_i \pm \sqrt{(1 + \beta - \eta\lambda_i)^2 - 4\beta}\right), \quad i = 1, ..., D \tag{133}$$

$$\mu^i_\pm = \frac{1}{2\beta\eta}\left(\beta - 1 + \eta\lambda_i \pm \sqrt{(1 + \beta - \eta\lambda_i)^2 - 4\beta}\right). \tag{134}$$

For the set of zero modes $\lambda_i = 0$, we get the following modes: $\kappa_+ = 1$, corresponding to motion only along w, with eigenvector $k^i = (0, q^i)$. In addition, there is a mixed longitudinal mode which includes a component of π along the zero manifold $k^i = (\mu_- q^i, q^i)$, and has an eigenvalue $\kappa_- = \beta$.

On the zero loss manifold, we can assume the Hessian is positive semi-definite, and that the positive eigenvalues satisfy

$$0 < c_1 \leq \lambda_i \leq c_2. \tag{135}$$

for constants c_1, c_2 independent of η, β.

We now analyze the spectrum of the Jacobian one eigenvalue at a time, and then use these results to informally control the relaxation rate of off-manifold perturbations. It is useful first to consider the conditions for stability, i.e. $|\kappa^i| < 1$, which are stated in (136,137) below:

$$\text{Case 1: If } \eta\lambda_i < (1 - \sqrt{\beta})^2, \text{ then } \kappa^i_\pm \in \mathbb{R} \text{ and } |\kappa^i_\pm| < 1 \text{ iff } 0 < \eta\lambda_i < 2(1 + \beta). \tag{136}$$

$$\text{Case 2: If } \eta\lambda_i > (1 - \sqrt{\beta})^2, \text{ then } \kappa^i_\pm \in \mathbb{C} \text{ and } |\kappa^i_\pm| = \sqrt{\beta} < 1. \tag{137}$$

Proof of Case 1: The condition $\eta\lambda_i < (1 - \sqrt{\beta})^2$ implies $A^2 > 4\beta$, where $A = 1 + \beta - \eta\lambda_i$. The condition for stability then requires $-1 < \kappa_i < +1$. We satisfy both sides of this inequality:

If $1 + \beta - \eta\lambda_i = A > 0$, then $|\kappa^i_-| < 1$,and $\kappa^i_+ > 0$, so we simply require $\kappa^i_+ < 1$, i.e.

$$\kappa_i < +1 \tag{138}$$

$$\frac{1}{2}(A + \sqrt{A^2 - 4\beta}) < 1 \tag{139}$$

$$+\sqrt{A^2 - 4\beta} < 2 - A \tag{140}$$

$$A^2 - 4\beta < 4 - 4A + A^2 \tag{141}$$

$$A = 1 + \beta - \eta\lambda_i < 1 + \beta \quad \Rightarrow \eta\lambda_i > 0 \tag{142}$$

If $1+\beta-\eta\lambda_i<0$, then $|\kappa_+^i|<1$, and $\kappa_-^i<1$, so we only require $\kappa_-^i>-1$

$$-1<\kappa_-^i \tag{143}$$
$$-1<\frac{1}{2}(-|A|-\sqrt{A^2-4\beta}) \tag{144}$$
$$|A|-2<-\sqrt{A^2-4\beta} \tag{145}$$
$$2-|A|>\sqrt{A^2-4\beta} \tag{146}$$
$$A^2-4|A|+4>A^2-4\beta \tag{147}$$
$$|A|=-1-\beta+\eta\lambda_i<1+\beta \quad \Rightarrow \eta\lambda_i<2(1+\beta) \tag{148}$$

Proof of Case 2: the condition $\eta\lambda_i>(1-\sqrt{\beta})^2$ implies $A^2<4\beta$, where $A=(1+\beta-\eta\lambda_i)$. The corresponding eigenvalues of the Jacobian can be written $\kappa_i^\pm=(1/2)(A\pm\sqrt{A^2-4\beta})=(1/2)(A\pm i\sqrt{4\beta-A^2})$. Computing the absolute value then gives $|\kappa_i^\pm|^2=(1/4)(A^2+4\beta-A^2)=\beta$ □

These results show us when the GD+momentum is stable. Next, assuming stability, we want to estimate the rate of convergence to the fixed point. More precisely, we would like to determine the fastest mode as well as the slowest mode. To this end, we define two quantities

$$\rho_1=\max\{|\kappa_\pm^i|\big|\ i=1,...,D,\ |\kappa_\pm^i|<1\}. \tag{149}$$
$$\rho_2=\min\{|\kappa_\pm^i|\big|\ i=1,...,D,\ |\kappa_\pm^i|<1\}. \tag{150}$$

Using the explicit scaling for momentum, and in the limit of small learning rate, we prove the following for ρ_1:

Lemma G.1. *Let $\beta=1-C\eta^\gamma$, and η sufficiently small:*

For $\gamma<1/2$, the condition for Case 1 (136) holds,

$$\rho_1\approx 1-(c_1/C)\eta^{1-\gamma}, \tag{151}$$
$$\rho_2\approx 1-C\eta^\gamma. \tag{152}$$

.

For $\gamma>1/2$, the condition for Case 2 (137), and

$$\rho_1=\sqrt{\beta}\approx 1-(C/2)\eta^\gamma, \tag{153}$$
$$\rho_2=\beta=1-C\eta^\gamma. \tag{154}$$

.

Proof: For small η, we find that

$$\eta^{-1}(1-\sqrt{\beta})^2=\left(1-\sqrt{1-C\eta^\gamma}\right)^2\approx C^2\eta^{2\gamma-1}/4. \tag{155}$$

When $\gamma>1/2$, this expression tends to zero as $\eta\to 0$. Therefore, for sufficiently small η, the condition for Case 2 (137) will be satisfied and $|\kappa^i|=\sqrt{\beta}$ for all $\lambda_i>0$. Since $\beta<\sqrt{\beta}$, this implies $\rho_1=\sqrt{\beta}$ and $\rho_2=\beta$. The scaling behavior in the Lemma follows by substitution.

For $\gamma<1/2$, (155) diverges as $\eta\to 0$, which means Case 1 (136) will obtain for all λ_i. Next, since for small η, $1+\beta-\eta\lambda_i>1+\beta-\eta c_2=2-C\eta^\gamma-\eta c_2>0$, since C and c_2 are order one constants, and $\eta\to 0$. In this case, the largest contribution from the nonzero eigenvalues λ_i will come from κ_+^i. In particular, we find

$$|\kappa_+^i|\leq\frac{1}{2}\left(1+\beta-\eta c_1+\sqrt{(1+\beta-\eta c_1)^2-4\beta}\right), \tag{156}$$
$$=\frac{1}{2}\left(2-C\eta^\gamma-\eta c_1+\sqrt{C^2\eta^{2\gamma}-2(2-C\eta^\gamma)\eta c_1+\eta^2c_1^2}\right). \tag{157}$$

For $\gamma < 1/2$, we have the hierarchy $\eta^\gamma > \eta^{2\gamma} > \eta > \eta^{\gamma+1} > \eta^2$. This allows us to simplify the upper bound

$$\rho_1 = \max|\kappa^i_+| \leq 1 - \frac{c_1}{C}\eta^{1-\gamma} + O(\eta). \tag{158}$$

Next, we find a lower bound for ρ_2. This will be controlled by κ^i_-. We may use then that

$$\rho_2 = \min\{|\kappa^i_-|\} \geq \frac{1}{2}\left(1 + \beta - \eta c_2 - \sqrt{(1+\beta-\eta c_1)^2 - 4\beta}\right), \tag{159}$$

$$= \frac{1}{2}\left(2 - C\eta^\gamma - \eta c_2 - \sqrt{C^2\eta^{2\gamma} - 2(2 - C\eta^\gamma)\eta c_1 + \eta^2 c_1^2}\right) \approx 1 - C\eta^\gamma + O(\eta^{1-\gamma}) \tag{160}$$

Finally, note that since $\eta^{1-\gamma} < \eta^\gamma$, we have that $1 - \frac{c_1}{C}\eta^{1-\gamma} > 1 - C\eta^\gamma = \beta$, so indeed $\rho_2 < \rho_1$.

Equipped with the upper and lower bounds on the spectrum leads naturally to bounds on the relaxation rate. For the purely decaying modes at $\gamma < 1/2$, we use $\rho_2^t \leq |\delta x_t^T| \leq \rho_1^t$, δx_t^T represents the projection of the fluctuations δx_t onto the transverse and mixed longitudinal modes. After applying Eqs. (151) and (152), we arrive at the result quoted in the main text in Sec. 3.3. For $\gamma > 1/2$, the modes are oscillatory. However, the eigenvalues within the unit circle have norm either $\sqrt{\beta}$, β, as reflected in Eqs. (153) and (154). This implies that we can estimate the decay rate of the envelope of the transverse and mixed longitudinal modes in this regime, thereby arriving at second expression quoted in the main text in Sec. 3.3.

H Linearized SGD and Ornstein-Uhlenbeck process on the valley

In this appendix, we provide a derivation of some of the statements quoted in Sec. 1.1. To get there, we start with the basic model for momentum SGD (1) but linearize around a point on the valley $w_0 \in \Gamma$ where $L(w_0) = \nabla L(w_0) = 0$. Let $w_k = w_0 + \delta w_k$, and define the Hessian $H(w_0) = \nabla^2 L(w_0)$. Then

$$\pi_{k+1} = \beta\pi_k - H(w_0)\delta w_k + \epsilon\sigma(w_0)\xi_k, \qquad \delta w_{k+1} = \delta w_k + \eta\pi_{k+1}, \tag{161}$$

Consider the projector along transverse nonzero eigemode λ of $H(w_0)$, P^T_λ, and define $P^T_\lambda x = X$, and $P^T_\lambda \pi = \Pi$. Let $\bar\sigma = P^T_\lambda\sigma$, and $P^T_\lambda H = \lambda P^T_\lambda$. Let $\bar\sigma\bar\sigma^\top = \Lambda$. Then

$$\Pi_{k+1} = \beta\Pi_k - \lambda X_k + \epsilon\bar\sigma(w_0)\xi_k, \tag{162}$$

$$X_{k+1} = X_k + \eta\Pi_{k+1}, \tag{163}$$

$$= X_k + \eta\beta\Pi_k - \eta\lambda X_k + \eta\epsilon\bar\sigma(w_0)\xi_k \tag{164}$$

This is a simple OU process, and we can easily compute the second moments. Define the second moments

$$C_{12}(k) = \langle X_k\Pi_k\rangle, \quad C_{11}(k) = \langle X_k X_k\rangle, \quad C_{22}(k) = \langle \Pi_k\Pi_k\rangle. \tag{165}$$

We find by taking the equations above, squaring them, then averaging over the noise,

$$C_{22}(k+1) = \beta^2 C_{22}(k) - 2\beta\lambda C_{12}(k) + \lambda^2 C_{11}(k) + \epsilon^2\Lambda, \tag{166}$$

$$C_{12}(k+1) = \eta\beta^2 C_{22}(k) + \beta(1-2\eta\lambda)C_{12}(k) - \lambda(1-\eta\lambda)C_{11}(k) + \eta\epsilon^2\Lambda, \tag{167}$$

$$C_{11}(k+1) = \eta^2\beta^2 C_{22} + 2\eta\beta(1-\eta\lambda)C_{12}(k) + .(1-\eta\lambda)^2 C_{11}(k) + \eta^2\epsilon^2\Lambda. \tag{168}$$

Next, assuming a stationary distribution implies $C(k+1) = C(k)$, which then allows us to solve for equilibrium variance, and extract the main quantity of interest, which is the variance of the weights. We find then

$$C_{11}(k) = \frac{\eta^2 \epsilon^2 \Lambda}{(1-\beta)\eta\lambda(2(1+\beta) - \eta\lambda)}. \tag{169}$$

In the limit of small η we extract the scaling behavior quoted in Sec.(1.1).

We can see how the mixing timescale τ_1, discussed in Sec. 1.1, arises from this linearized analysis. By taking the expectation value of the OU process, the noise will vanish and we find that the average values follow the linearized GD dynamics analyzed in G. Thus, from this linearized GD analysis we can extract the characteristic timescale for the OU process to approach its mean value.