
A Perturbative Analysis of Stochastic Descent

Anonymous Author(s)

Affiliation

Address


email

Abstract

We analyze stochastic gradient descent (SGD) at small learning rates. Unlike prior analyses based on stochastic differential equations, our theory models discrete time and hence non-Gaussian noise. We prove that gradient noise systematically pushes SGD toward flatter minima. We characterize when and why flat minima overfit less than other minima. We generalize the Akaike information criterion (AIC) to a smooth estimator of overfitting, hence enabling gradient-based model selection. We show how non-stochastic GD with a modified loss function may emulate SGD. We verify our predictions on convnets for CIFAR-10 and Fashion-MNIST.

1 Introduction

Practitioners benefit from the intuition that stochastic gradient descent (SGD) approximates noiseless gradient descent (GD) [Bottou, 1991]. In this paper, we refine that intuition by showing how gradient noise biases learning toward certain areas of weight space. Departing from prior work, we model discrete time and hence non-Gaussian noise. Indeed, we derive corrections to continuous-time, Gaussian-noise approximations such as ordinary and stochastic differential equations (ODE, SDE). For example, we construct a loss landscape on which SGD eternally cycles counterclockwise, a phenomenon impossible with ODEs. Leaving the rigorous development of the general theory to Appendix §B, our paper body highlights our theory’s intuition and main corollaries.

Our analysis offers a novel interpretation of SGD as a sum of many concurrent interactions between weights and data. Diagrams such as , analogous to those of Feynman [1949] and Penrose [1971], depict these interactions. §B.8 discusses this bridge to physics — and its relation to Hessian methods and natural GD — as topics for future research. We also discuss how this work may lessen the energy footprint required to train machine learning models. More broadly, our work adds to the body of theory on optimization in the face of uncertainty, theory that may one day inform solutions to emerging issues in user privacy and pedestrian safety.

1.1 Example of diagram-based computation of SGD’s test loss

If we run SGD for T gradient steps with learning rate η starting at weight θ_0 , then by Taylor expansion we may express the expected test loss of the final weight θ_T in terms of statistics of the loss landscape evaluated at θ_0 . As is, this Taylor series is unwieldy to write and interpret. Our technical contribution is to organize the computation of this Taylor series via combinatorial objects we call *diagrams*:


Main Idea (Informal). We may enumerate the diagrams, and we may assign to each diagram a number that depends on η, T , such that summing those numbers over all diagrams yields SGD’s expected test loss. Restricting to the finitely many diagrams with $\leq d$ edges leads to $o(\eta^d)$ error.


Deferring details, we illustrate the Main Idea by deriving a new result (Example 1). This shows our formalism’s work flow, but only in later sections will we explain the mathematics.

Definition 1 (Informal). Let $l_x(\theta)$ be weight θ 's loss on datapoint x . We define a dictionary between (a) tensors relating to this loss landscape and (b) diagram fragments that we will soon assemble:


$$\begin{aligned} G &\triangleq \mathbb{E}_x [\nabla l_x(\theta)] \triangleq \text{↘} \\ H &\triangleq \mathbb{E}_x [\nabla \nabla l_x(\theta)] \triangleq \text{↘↘} \quad C \triangleq \mathbb{E}_x [(\nabla l_x(\theta) - G)^2] \triangleq \text{↘↘} \\ J &\triangleq \mathbb{E}_x [\nabla \nabla \nabla l_x(\theta)] \triangleq \text{↘↘↘} \quad S \triangleq \mathbb{E}_x [(\nabla l_x(\theta) - G)^3] \triangleq \text{↘↘↘} \end{aligned}$$

Here, G, H, J denote the loss's derivatives with respect to θ , and G, C, S denote the gradient's cumulants with respect to the randomness in x . There are infinitely many analogues (with more edges), but they will not play a role in our leading order results. Each $\nabla^d l_x$ corresponds to a degree- d node, and fuzzy outlines group nodes that occur within the same expectation.

We obtain **diagrams** by pairing together the loose ends of the above fragments.¹ For instance, we may join $C = \text{↘↘}$ with $H = \text{↘↘}$ to get . As another example, we may join two copies of $G = \text{↘}$

with two copies of $H = \text{↘↘}$ to get . Intuitively, each diagram represents the interaction of its components: of gradients (G), noise (C, S, \dots) and curvature (H, J, \dots). In fact, §A.6 physically interprets edges as carrying information between updates and toward the test measurement. \diamond

Example 1. Does non-Gaussian noise affect SGD? Specifically, let's compute how the *skewness* S affects SGD's test loss. The recipe is to identify the fewest-edged diagrams containing $S = \text{↘↘↘}$. In

this case, there is one fewest-edged diagram — ; it results from joining S with $J = \text{↘↘↘}$. To evaluate a diagram, we multiply its components (here, S, J) with exponentiated ηH 's, one for each edge (here, there are three edges). The result is easiest to write in terms of an eigenbasis of ηH :

$$-\frac{\eta^3}{3!} \sum_{\mu\nu\lambda} S_{\mu\nu\lambda} \frac{1 - \exp(-T\eta(H_{\mu\mu} + H_{\nu\nu} + H_{\lambda\lambda}))}{\eta(H_{\mu\mu} + H_{\nu\nu} + H_{\lambda\lambda})} J_{\mu\nu\lambda}$$

This is leading order contribution of skewed noise (S) to SGD's test loss. \diamond

Remark 1. To understand Example 1's result, we specialize to isotropic curvature ($\eta H = \|\eta H\|_2 I$) and take $T \rightarrow \infty$, obtaining: $-(\eta^3/3!) \sum_{\mu\nu\lambda} S_{\mu\nu\lambda} J_{\mu\nu\lambda}/3\|\eta H\|_2$. Since $J = \nabla H$, $J/\|\eta H\|_2$ measures the relative change in the curvature, H , with respect to θ . So skewed noise affects SGD in proportion to the logarithmic derivative of curvature. Gaussian approximations (e.g. SDE) miss this effect.

1.2 Background, notation, and assumptions

Let $G, H, J; C, S$ be as in §1.1. They are tensors with $1, 2, 3; 2, 3$ indices, respectively. We may implicitly sum repeated Greek indices: if a covector A and a vector B ² have coefficients A_μ, B^μ , then $A_\mu B^\mu \triangleq \sum_\mu A_\mu \cdot B^\mu$. We regard the learning rate as an inverse metric $\eta^{\mu\nu}$ that converts gradient covectors to displacement vectors [Bonnabel, 2013]. We use the learning rate η to raise indices; thus, $H^\mu_\lambda \triangleq \sum_\nu \eta^{\mu\nu} H_{\nu\lambda}$ and $C^\mu_\mu \triangleq \sum_{\mu\nu} \eta^{\mu\nu} \cdot C_{\nu\mu}$. Though η is a tensor, we may still define $o(\eta^d)$: a quantity q vanishes to order η^d when $\lim_{\eta \rightarrow 0} q/p(\eta) = 0$ for some homogeneous degree- d polynomial p .

We fix a loss function $l : \mathcal{M} \rightarrow \mathbb{R}$ on a space \mathcal{M} of weights. We fix a distribution \mathcal{D} from which unbiased estimates of l are drawn. We write l_x for a generic sample from \mathcal{D} and $(l_n : 0 \leq n < N)$ for a training sequence drawn i.i.d. from \mathcal{D} . We refer both to n and to l_n as *training points*. We assume §B.1's hypotheses, e.g. that l, l_x are analytic and that all moments exist. For instance, our theory models tanh networks with cross entropy loss on bounded data — and with weight sharing, skip connections, soft attention, dropout, and weight decay. But it does not model ReLU networks.

Our general theory describes SGD with any number N of training points, T of updates, and B of points per batch. SGD then runs T many updates (i.e. $E = TB/N$ epochs, i.e. $M = T/N$ updates per point) of the form $\theta^\mu := \theta^\mu - \eta^{\mu\nu} \nabla_\nu \sum_{n \in \mathcal{B}_t} l_n(\theta)/B$, where in each epoch, \mathcal{B}_t , the t th batch, is sampled without replacement from the training set. For simplicity, our paper body (but not the appendices) will assume unless otherwise stated that **SGD has $E = B = 1$ and GD has $T = B = N$** .

¹ A diagram's colors and geometric layout lack meaning: we **color** only for convenient reference, e.g. to a diagram's "green nodes". Only the topology of a diagram — not its size or angles — appear in our theory.

² Vectors/covectors, a.k.a. column/row vectors, represent distinct geometric concepts [Kolář et al., 1993].

1.3 Related work


It was Kiefer and Wolfowitz [1952] who, in uniting gradient descent [Cauchy, 1847] with stochastic approximation [Robbins and Monroe, 1951], invented SGD. Since the development of back-propagation for efficient differentiation [Werbos, 1974], SGD has been used to train connectionist models, e.g. neural networks [Bottou, 1991], recently to remarkable success [LeCun et al., 2015].

Several lines of work treat the overfitting of SGD-trained networks [Neyshabur et al., 2017a]. For example, Bartlett et al. [2017] controls the Rademacher complexity of deep hypothesis classes, leading to optimizer-agnostic generalization bounds. Yet SGD-trained networks generalize despite their ability to shatter large sets [Zhang et al., 2017], so generalization must arise from not only architecture but also optimization [Neyshabur et al., 2017b]. Others approximate SGD by SDE to analyze implicit regularization (e.g. Chaudhari and Soatto [2018]), but, per Yaida [2019a], such continuous-time analyses cannot treat SGD noise correctly. We avoid these pitfalls by Taylor expanding around $\eta = 0$ as in Roberts [2018]; unlike that work, we generalize beyond order η^1 and $T = 2$.

Our predictions are vacuous for large η . Other analyses treat large- η learning phenomenologically, whether by finding empirical correlates of gen. gap [Liao et al., 2018], by showing that *flat* minima generalize (Hoffer et al. [2017], Keskar et al. [2017], Wang et al. [2018]), or by showing that *sharp* minima generalize (Stein [1956], Dinh et al. [2017], Wu et al. [2018]). At least for small η , our theory reconciles these clashing claims.

Prior work analyzes SGD perturbatively: Dyer and Gur-Ari [2019] perturb in inverse network width, using 't Hooft diagrams to correct the Gaussian Process approximation for specific deep nets. Perturbing to order η^2 , Chaudhari and Soatto [2018] and Li et al. [2017] are forced to assume uncorrelated Gaussian noise. By contrast, we use Penrose diagrams to compute test losses to arbitrary order in η . We allow correlated, non-Gaussian noise and thus *any* smooth architecture. For instance, we do not assume information-geometric relationships between C and H ,¹ so we may model VAEs.


2 Theory, specialized to $E = B = 1$ SGD's test loss

A *diagram* is a finite rooted tree equipped with a partition of its nodes that obeys the *path condition*: no path from leaf to root may encounter any part more than once. We specify the root by drawing it rightmost. We draw the parts of the partition by grouping each part's nodes inside fuzzy outlines. A diagram is *irreducible* when each of its degree-2 nodes is in a part of size one. An *embedding* f of a diagram D is an injection from D 's parts to (integer) times $0 \leq t \leq T$ that sends the root to T and s.t., for each path from leaf to root, the corresponding sequence of times increases. So f might send 's red part to $t = 3$ and its green part to $t = 4$, but — because the green node has a red child — not vice versa. Let $|\text{Aut}_f(D)|$ count automorphisms of D that preserve f . Up to unbiasing terms,² we construct the *re-summed value* $\text{rvalue}_f(D)$ as follows:

Node rule: insert a factor a $\nabla^d l_x$ for each degree d node.

Outline rule: group each part's nodes within brackets $\mathbb{E}_x[\]$.


Edge rule: if f sends an edge's endpoints to times t, t' , insert a factor of $K^{|t'-t|-1}\eta$, where $K \triangleq (I - \eta H)$.











So if f maps 's red part to time $t = T - \Delta t$, then (the red part gives S ; the green part, J):

$$\text{rvalue}_f\left(\text{diagram}\right) = S_{\mu\lambda\rho}(K^{\Delta t-1}\eta)^{\mu\nu}(K^{\Delta t-1}\eta)^{\lambda\sigma}(K^{\Delta t-1}\eta)^{\rho\pi}J_{\nu\sigma\pi}$$

In fact, we may integrate this expression per Remark 2 to recover Example 1.

¹ Disagreement of C and H is typical in modern learning [Roux et al., 2012, Kunstner et al., 2019].

² For example, we actually define  to be the cumulant $C = \mathbb{E}[(\nabla l_x(\theta) - G)^2]$, not the moment $\mathbb{E}[(\nabla l_x(\theta))^2]$. This centering is routine (see §B.4), tedious to notate, and un-germane, so we ignore it in the paper body.

Examples: The diagrams , , each have 2 parts; ,  have 3. Corollaries 2, 4, 3 have $E \neq 1 \neq B$, so they feature  and , generalized diagrams that violate the path condition. Diagrams ,  are irreducible; due to their green nodes, ,  are not. For all f , $|\text{Aut}_f(\text{diagram})| = 1$ and $|\text{Aut}_f(\text{diagram})| = 2$.

2.1 Main result

Theorem 1 expresses SGD’s test loss as a sum over diagrams. A diagram with d edges scales as $O(\eta^d)$, so the following is a series in η . We later truncate the series to small d , thus focusing on few-edged diagrams and simplifying the combinatorics of embeddings.

Theorem 1 (Special case of $E = B = 1$). *For any T : for η small enough, SGD has expected test loss*


$$\sum_{\substack{D \text{ an irreduc-} \\ \text{-ible diagram}}} \sum_{\substack{f \text{ an embed-} \\ \text{-ding of } D}} \frac{(-1)^{|\text{edges}(D)|}}{|\text{Aut}_f(D)|} \text{rvalue}_f(D)$$

Remark 2. In practice, we approximate sums over embeddings by integrals over times and $(I - \eta H)^t$ by $\exp(-\eta H t)$, reducing to a routine integration of exponentials at the cost of an error factor $1 + o(\eta)$.


Theorem 2. *If θ_\star is a non-degenerate local minimum of l (i.e. $G(\theta_\star) = 0$ and $H(\theta_\star) > 0$), then for SGD initialized sufficiently close to θ_\star , the d th-order truncation of Theorem 1 converges as $T \rightarrow \infty$.*

Caution: the $T \rightarrow \infty$ limit in Theorem 2 might not measure any well-defined limit of SGD, since the limit might not commute with the infinite sum. We have not seen such pathologies in practice, so we will freely speak of “SGD in the large- T limit” as informal shorthand when referencing this Theorem.


2.2 SGD descends on a C -smoothed landscape and prefers minima flat w.r.t. C .



Corollary 1 (Computed from ). *Run SGD for $T \gg 1/\eta H$ from a non-degenerate test minimum. Written in an eigenbasis of ηH , θ has an expected displacement of*

$$-\frac{\eta^3}{2} \sum_{\mu\nu} C_{\mu\nu} \frac{1}{\eta(H_{\mu\mu} + H_{\nu\nu})} J_{\mu\nu\lambda} \frac{1}{H_{\lambda\lambda}} + o(\eta^2)$$

Intuitively, $D = \text{diagram of a node with a self-loop and an incoming edge}$ connects the subdiagram $\propto CH$, via an extra edge on the green node (an extra ∇ on H), to D ’s degree-1 root, G . By l’Hôpital,¹ the displacement is $\propto -C\nabla H$. That is, SGD moves toward minima that are flat *with respect to C* (Figure 1 ). Taking limits to drop the non-degeneracy hypothesis, we expect *sustained* motion toward flat regions in a valley of minima. By avoiding Wei and Schwab [2019]’s assumptions of constant C , we find that SGD’s velocity field is typically non-conservative, i.e. has curl (§3.2). Indeed, $\nabla(CH)$ is a total derivative but $C\nabla H$ is not. Since, by low-pass filter theory, $CH/2 + o(C)$ is the loss increase upon convolving l with a C -shaped Gaussian, we say that SGD descends on a C -smoothed landscape that changes as C does. Our $T \gg 1$ result is $\Theta(\eta^2)$, while Yaida [2019b]’s similar $T = 2$ result is $\Theta(\eta^3)$. Indeed, our analysis integrates the noise over many updates, hence amplifying C ’s effect. Experiments verify our law.

2.3 Both flat and sharp minima overfit less

Intuitively, sharp minima are robust to slight changes in the average *gradient* and flat minima are robust to slight *displacements* in weight space (Figure 1 ). However, as SGD by definition equates displacements with gradients, it may be unclear how to reason about overfitting in the presence of curvature. Our theory, by (automatically) accounting for the implicit regularization of fixed- T descent, shows that both effects play a role. In fact, by routine calculus on the left hand side of Corollary 2, overfitting is maximized for medium minima with curvature $H \sim (\eta T)^{-1}$.

Corollary 2 (from , ). *Initialize GD at a non-degenerate test minimum θ_\star . The overfitting (test loss minus $l(\theta_\star)$) and gen. gap (test minus train loss) due to training are:*

$$\left(\frac{C/N}{2H}\right)_{\mu\nu}^{\rho\lambda} \left((I - \exp(-\eta TH))^{\otimes 2}\right)_{\rho\lambda}^{\mu\nu} + o(\eta^2) \quad ; \quad \left(\frac{C/N}{H}\right)_{\mu\nu}^{\mu\lambda} (I - \exp(-\eta TH))_{\lambda}^{\nu} + o(\eta)$$

The gen. gap tends to $C_{\mu\nu}(H^{-1})^{\mu\nu}/N$ as $T \rightarrow \infty$. For maximum likelihood (ML) estimation in well-specified models near the “true” minimum, $C = H$ is the Fisher metric, so we recover AIC: (model dimension)/ N . Unlike AIC, our more general expression is descendably smooth, may be used with MAP or ELBO tasks instead of just ML, and does not assume a well-specified model.

¹ Roughly: if a displacement $\Delta\theta$ grows loss by $GC\nabla H$ nats, and by G nats per foot, then $\Delta\theta$ is $C\nabla H$ feet.

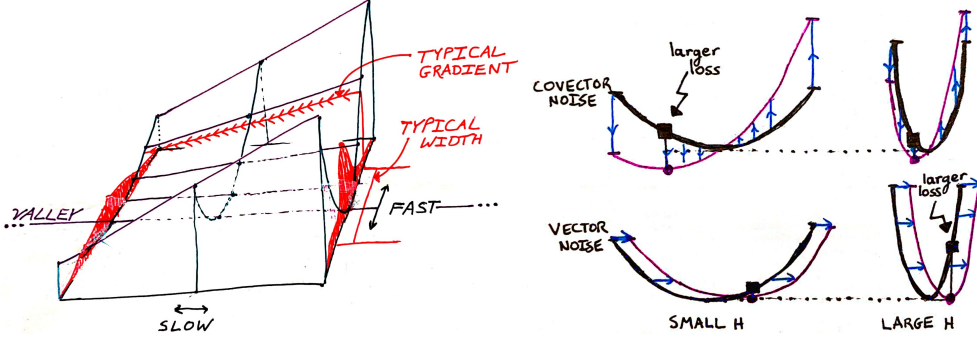


Figure 1: **Geometric intuition for curvature-noise interactions.** **Left:** Gradient noise pushes SGD toward flat minima (Corollary 1). The red densities show the typical θ s, perturbed from the minimum due to noise C , in two cross sections of the loss valley. $J = \nabla H$ measures how curvature changes across the valley. Our theory does not assume separation between “fast” and “slow” modes, but we label them in the picture to ease comparison with Wei and Schwab [2019]. Compare with Figure 4. **Right:** Both curvature and the structure of noise affect overfitting. In each of the four subplots, the \leftrightarrow axis represents weight space and the \uparrow axis represents loss. $\square \dots \square$: covector-perturbed landscapes favor large H s. $\square \dots \square$: vector-perturbed landscapes favor small H s. SGD’s implicit regularization interpolates between these rows (Corollary 2).

2.4 High- C regions repel small- (E, B) SGD more than large- (E, B) SGD

Physical intuition (§A.6) suggests that noise repels SGD. In particular, if two neighboring regions of weight space have high and low levels of gradient noise, respectively, then we expect the rate at which θ jumps from the former to the latter to exceed the opposite rate. There is thus a net movement toward regions of small C ! This mechanism parallels the Chladni effect [Chladni, 1787] (Figure 2).¹ Our theory makes this intuition precise; the drift is in the direction of $-\nabla C$, and the effect is strongest when gradient noise is not averaged out by large batch sizes.

Corollary 3 (). SGD avoids high- C regions more than GD: $l_C \triangleq \frac{N-1}{4N} \nabla^\mu C_\nu^\mu = \mathbb{E} [\theta_{GD} - \theta_{SGD}]^\mu - o(\eta^2)$. If \hat{l}_c is a smooth unbiased estimator of l_c , then GD on $l + \hat{l}_c$ has an expected test loss that agrees with SGD’s to order η^2 . We call this method GDC.

An analogous form of averaging occurs over multiple epochs. For a tight comparison, we scale the learning rates appropriately so that, to leading order, few-epoch and many-epoch SGD agree. Then few and many-epoch SGD differ, to leading order, in their sensitivity to ∇C :

Corollary 4 (). SGD with $M = 1$ and $\eta = \eta_0$ avoids high- C regions more than SGD with $M = M_0$ and $\eta = \eta_0/M_0$. Precisely: $\mathbb{E} [\theta_{M=M_0} - \theta_{M=1}]^\mu = \left(\frac{M_0-1}{4M_0} \right) N (\nabla^\mu C_\nu^\nu) + o(\eta^2)$.

2.5 Non-Gaussian noise affects SGD but not SDE

Stochastic differential equations (SDE: see Liao et al. [2018]) are a popular theoretical approximation of SGD, but SDE and SGD differ in several ways. For instance, the inter-epoch noise correlations in multi-epoch SGD measurably affect SGD’s final test loss (Corollary 4), but SDE assumes uncorrelated gradient updates. Even if we restrict to single-epoch SDE, differences arise due to time discretization and non-Gaussian noise. Intuitively, SGD and SDE respond differently to changes in curvature:

Corollary 5 (,). SGD’s test loss is $\frac{T}{2} C_{\mu\nu} H^{\mu\nu} + o(\eta^2)$ more than ODE’s and SDE’s. The deviation from SDE due to skewed noise is $-\frac{T}{6} S_{\mu\nu\lambda} J^{\mu\nu\lambda} + o(\eta^3)$.²

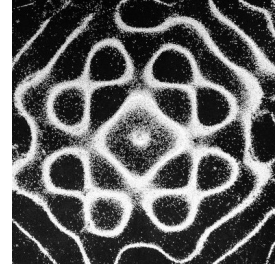


Figure 2: **Chladni plate.** Grains of sand on a vibrating plate tend toward stationary regions.

¹ From Pierre Dragicevic and Yvonne Jansen’s data physicalization project, Creative Commons BY-SA 3.0.
² This approximation of Example 1’s more exact expression agrees with the latter to leading order in η .

3 Experiments

Despite the convergence results in Theorems 1 and 2, we have no theoretical bounds for the domain and *rate* of convergence. Instead, we test our predictions by experiment. We perceive support for our theory in drastic rejections of the null hypothesis. For instance, in Figure 3 $\square\square\square$, [Chaudhari and Soatto, 2018] predicts a velocity of 0 while we predict a velocity of $\eta^2/6$. Here, I bars, + signs, and shaded regions all mark 95% confidence intervals based on the standard error of the mean. §C describes neural architectures, the definitions of artificial landscapes, sample sizes, and further plots.

3.1 Training time, epochs, and batch size; C repels SGD more than GD

We test Theorem 1's order η^3 truncation on smooth convnets for CIFAR-10 and Fashion-MNIST. Theory agrees with experiment through timescales long enough for accuracy to increase by 0.5% (Figure 3 $\square\square\square$, $\square\square\square$). §C.7 supports Corollary 4's predictions about epoch number. Figure 3 $\square\square\square$ tests Corollary 3's claim that, relative to GD, high- C regions *repel* SGD. This is significant because C controls the rate at which the gen. gap (test minus train loss) grows (Corollary 2, Figure 3 $\square\square\square$).

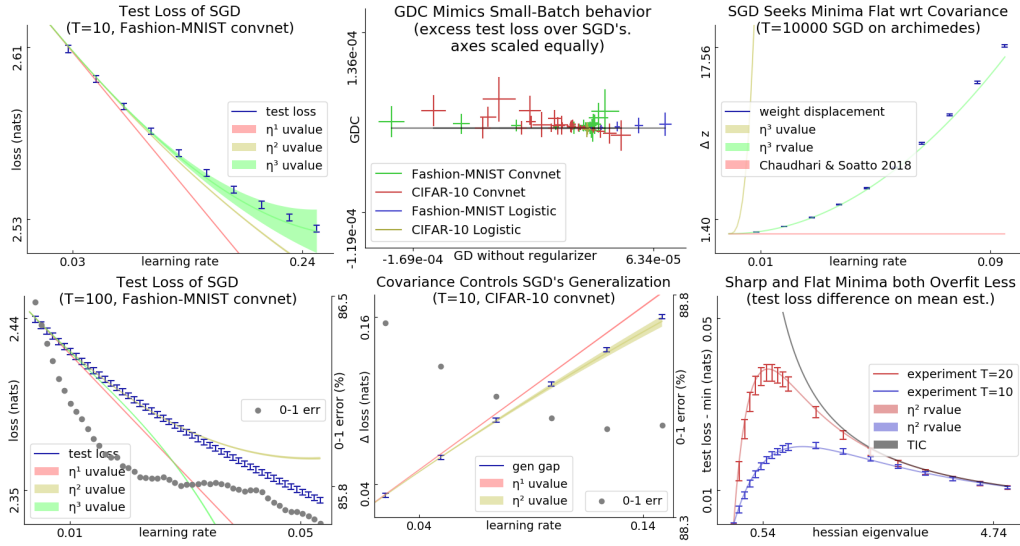



Figure 3: Experiments on natural and artificial losses. The label *rvalue* refers to Theorem 1's predictions, approximated as in Remark 2. Curves marked *uvalue* are polynomial approximations to Theorem 1's result (see §A.5). *uvalues* are simpler to work with but (see $\square\square\square$) may be less accurate. **Left: Perturbation models SGD for small ηT .** Fashion-MNIST convnet's test loss vs learning rate. In this small T setting, we choose to use our theory's simpler un-resummed values (A.4) instead of the more precise *rvalues*. $\square\square\square$: For all init.s tested (1 shown, 11 unshown), the order 3 prediction agrees with experiment through $\eta T \approx 10^0$, corresponding to a decrease in 0-1 error of $\approx 10^{-3}$. $\square\square\square$: For large ηT , our predictions break down. Here, the order-3 prediction holds until the 0-1 error improves by $5 \cdot 10^{-3}$. Beyond this, 2nd order agreement with experiment is coincidental. **Center: C controls gen. gap and distinguishes GD from SGD.** With equal-scaled axes, $\square\square\square$ shows that GDC matches SGD (small vertical variance) better than GD matches SGD (large horizontal variance) in test loss for a range of η ($\approx 10^{-3} - 10^{-1}$) and init.s (zero and several Xavier-Glorot trials) for logistic regression and convnets. Here, $T = 10$. $\square\square\square$: CIFAR-10 generalization gaps. For all init.s tested (1 shown, 11 unshown), the degree-2 prediction agrees with experiment through $\eta T \approx 5 \cdot 10^{-1}$. **Right: Predictions near minima excel for large ηT .** $\square\square\square$: SGD travels ARCHIMEDES' valley of global minima in the positive z direction. Note: H and C are bounded across the valley, we see drift for all small η , and we see displacement exceeding the landscape's period of 2π . So: the drift is not a pathology of well-chosen η , of divergent noise, or of ephemeral initial conditions. $\square\square\square$: For MEAN ESTIMATION with fixed C and a range of H s, initialized at the truth, the test losses after fixed- T GD are smallest for very sharp and very flat H . Near $H = 0$, our predictions improve on TIC [Dixon and Ward, 2018] and thus on AIC.

3.2 Minima that are flat with respect to C attract SGD

To test the claimed dependence on C , §C.1 constructs a landscape, ARCHIMEDES, with non-constant C throughout its valley of global minima. Figure 4 depicts ARCHIMEDES’ chiral shape.¹ As in Archimedes’ screw or Rock-Paper-Scissors, each point θ has a neighbor that, from $C(\theta)$ ’s perspective but not absolutely, is flatter. This permits eternal motion despite the landscape’s symmetry. Indeed, Corollary 1 predicts a z -velocity of $+\eta^2/6$ per timestep, while Chaudhari and Soatto [2018]’s SDE-based analysis predicts a constant velocity of 0.² Our prediction agrees with experiment (Figure 3 ). Because SGD’s motion depends smoothly on the landscape, the special case of ARCHIMEDES implies that non-conservativity is typical. One may have sought an “effective loss” \tilde{l} such that, up to \sqrt{T} diffusion terms, SGD on l matches ODE on \tilde{l} . The non-conservativity of SGD’s velocity shows that no such \tilde{l} exists.

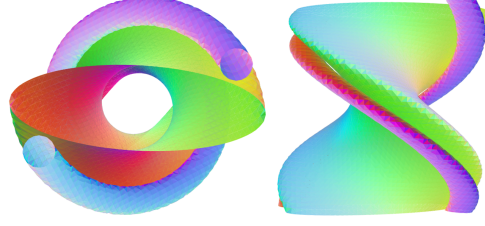






Figure 4: **ARCHIMEDES**. A **green** level surface of l twists around a valley of minima (z axis) at its center; l is large outside this surface. Due to anisotropic noise, θ scatters away from the z axis toward the **purple** tubes. SGD pushes the scattered θ s toward lower loss, i.e. toward the level surface, and so toward larger z . The z axis points into the page (**left**) or upward (**right**).

3.3 Sharp and flat minima both overfit less than medium minima

Prior work (§1.3) finds both that *sharp* minima overfit less (for, l^2 regularization sharpens minima) or that *flat* minima overfit less (for, flat minima are robust to small displacements). In fact, both phenomena occur, and noise structure determines which dominates (Corollary 2). This effect appears even in MEAN ESTIMATION (§C.1): Figure 3 . To combat overfitting, we may add Corollary 2’s expression for gen. gap to l . By descending on this regularized loss, we may tune smooth hyperparameters such as l_2 regularization coefficients for small datasets ($H \ll C/N$) (§C.7). Since matrix exponentiation takes time cubic in dimension, this regularizer is most useful for small models.

4 Conclusion: implications for practice

We presented a diagram-based method for studying stochastic optimization on short timescales or near minima. Corollaries 1 and 2 together offer insight into SGD’s success in training deep networks: SGD avoids curvature and noise, and curvature and noise control generalization.

Analyzing , we proved that **flat and sharp minima both overfit less** than medium minima. Intuitively, flat minima are robust to vector noise, sharp minima are robust to covector noise, and medium minima robust to neither. We thus proposed a regularizer enabling gradient-based hyperparameter tuning. Inspecting , we extended Wei and Schwab [2019] to nonconstant, nonisotropic covariance to reveal that **SGD descends on a landscape smoothed by the current covariance C** . As C evolves, the smoothed landscape evolves, resulting in non-conservative dynamics. Examining , we showed that **GD may emulate SGD**, as conjectured by Roberts [2018]. This is significant because, while small batch sizes can lead to better generalization [Bottou, 1991], modern infrastructure increasingly rewards large batch sizes [Goyal et al., 2018].

Since our predictions depend only on loss data near initialization, they break down after the weight moves far from initialization. Our theory thus best applies to small-movement contexts, whether for long times (large ηT) near an isolated minimum or for short times (small ηT) in general. Thus, the theory might help to analyze meta-learners based on fine-tuning (e.g. Finn et al. [2017]’s MAML).

Much as meteorologists understand how warm and cold fronts interact despite long-term forecasting’s intractability, we quantify how curvature and noise contribute to counter-intuitive dynamics governing each short-term interval of SGD’s trajectory. Equipped with our theory, practitioners may now refine intuitions — e.g. that SGD descends on the training loss — to account for noise.

¹ We made these plots with the help of Paul Seeburger’s online applet, CalcPlot3D.

² Indeed, ARCHIMEDES’ velocity is η -perpendicular to the image of $(\eta C)_\theta^i$ in tangent space.

240 Broader impacts

241 Though machine learning has the long-term potential for vast improvements in world-wide quality of
242 life, it is today a source of enormous carbon emissions [Strubell et al., 2019]. Our analysis of SGD
243 may lead to a reduced carbon footprint in three ways.

244 **First**, §2.4 shows how to modify the loss landscape so that large-batch GD enjoys the stochastic
245 regularizing properties of small-batch SGD, or (symmetrically) so that small-batch SGD enjoys the
246 stability of large-batch GD. By unchaining the effective batch size from the actual batch size, we
247 raise the possibility of training neural networks on a wider range of hardware than currently practical.
248 For example, asynchronous concurrent SGD (e.g. Niu et al. [2011]) might require less inter-device
249 communication and therefore less power. **Second**, §4 discusses an application to meta-learning,
250 which has the potential to decrease the per-task sample complexity and hence carbon footprint of
251 modern machine learning. **Third**, the modification of AIC developed in §2.3 and §3.3 permits certain
252 forms of model selection by gradient descent rather than brute force search. This might drastically
253 reduce the energy consumed during model selection.

254 More broadly, this paper analyzes optimization in the face of uncertainty. As machine learning systems
255 deployed today must increasingly address user privacy, pedestrian safety, and dataset diversity, it
256 becomes important to recognize that test sets and training sets differ. Toward this end, theoretical
257 work relating to non-Gaussian noise may assist practitioners in building provably non-discriminatory,
258 safe, or private models (e.g. Dwork et al. [2006]). By quantifying how correlated, non-Gaussian
259 gradient noise affects descent-based learning, this paper contributes to such broader theory.

260 That said, insofar as our theory furthers practice, it may instead help to popularize GPU-intensive
261 learning, thus negating the aforementioned benefits and accelerating climate change. Perhaps, then, it
262 makes sense to examine the research goals that so often lead to massive computational costs. For
263 instance, only recently have these authors examined their routine assumption that smaller test losses
264 are worth seeking; *was this assumption obvious because it is true or merely because it is familiar?*
265 In fact, even “pure” theory hides a humbling host of assumptions ingrained and un-examined. For
266 example, we began the work reported here by asking, *which minima does SGD prefer?* Only after
267 careful analysis did we realize that the question is ill-founded, for there is no absolute metric that
268 SGD minimizes (§3.2)! Zooming out: the authors have long prized sample efficiency — and yet,
269 should efficiency always be our goal? Reflecting on optimization in general, Ardila–Mantilla [2019]
270 suggests: perhaps it is not efficiency that we should seek, but rather

271 delight,

272 surprise, and

273 beauty.

274 Acknowledgements

275 Redacted during double-blind peer review.

References

- P.-A. Absil, R. Mahony, and R. Sepulchre. Optimization algorithms on matrix manifolds, chapter 4. *Princeton University Press*, 2007.
- S.-I. Amari. Natural gradient works efficiently. *Neural Computation*, 1998.
- F. Ardila–Mantilla. Cat(0) geometry, robots, and society. *Arxiv Preprint*, 2019.
- P.L. Bartlett, D.J. Foster, and M.J. Telgarsky. Spectrally-normalized margin bounds for neural networks. *NeurIPS*, 2017.
- S. Bonnabel. Sgd on riemannian manifolds. *IEEE Transactions on Automatic Control*, 2013.
- L. Bottou. Stochastic gradient learning in neural networks. *Neuro-Nîmes*, 1991.
- A.-L. Cauchy. Méthode générale pour la résolution des systèmes d’équations simultanées. *Comptes rendus de l’Académie des Sciences*, 1847.
- P. Chaudhari and S. Soatto. Sgd performs variational inference, converges to limit cycles for deep networks. *ICLR*, 2018.
- E.F.F. Chladni. Entdeckungen über die theorie des klages. *Leipzig*, 1787.
- Laurent Dinh, R. Pascanu, S. Bengio, and Y. Bengio. Sharp minima can generalize for deep nets. *ICLR*, 2017.
- M.F. Dixon and T. Ward. Takeuchi information as a form of regularization. *Arxiv Preprint*, 2018.
- C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. *Theory of Cryptography*, 2006.
- E. Dyer and G. Gur-Ari. Asymptotics of wide networks from feynman diagrams. *ICML Workshop*, 2019.
- F. Dyson. The radiation theories of tomonaga, schwinger, and feynman. *Physical Review*, 1949.
- R.P. Feynman. A space-time appxoach to quantum electrodynamics. *Physical Review*, 1949.
- C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *ICML*, 2017.
- C.F. Gauss. Theoria combinationis obsevationum erroribus minimis obnoxiae, section 39. *Proceedings of the Royal Society of Gottingen*, 1823.
- P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd. *Data @ Scale*, 2018.
- E. Hoffer, I. Hubara, and D. Soudry. Train longer, generalize better. *NeurIPS*, 2017.
- N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P.T.P. Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *ICLR*, 2017.
- J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *Annals of Mathematical Statistics*, 1952.
- I. Kolář, P.W. Michor, and J. Slovák. Natural operations in differential geometry. *Springer*, 1993.
- A. Krizhevsky. Learning multiple layers of features from tiny images. *UToronto Thesis*, 2009.
- F. Kunstner, P. Hennig, and L. Balles. Limitations of the empirical fisher approximation for natural gradient descent. *NeurIPS*, 2019.
- L.D. Landau and E.M. Lifshitz. The classical theory of fields. *Addison-Wesley*, 1951.
- L.D. Landau and E.M. Lifshitz. Mechanics. *Pergamon Press*, 1960.

316 Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 2015.

317 Qianxiao Li, Cheng Tai, and Weinan E. Stochastic modified equations and adaptive stochastic
318 gradient algorithms i. *PMLR*, 2017.

319 Qianli Liao, B. Miranda, A. Banburski, J. Hidary, and T. Poggio. A surprising linear relationship
320 predicts test performance in deep networks. *Center for Brains, Minds, and Machines Memo 91*,
321 2018.

322 B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. Exploring generalization in deep
323 learning. *NeurIPS*, 2017a.

324 B. Neyshabur, R. Tomioka, R. Salakhutdinov, and N. Srebro. Geometry of optimization and implicit
325 regularization in deep learning. *Chapter 4 from Intel CRI-CI: Why and When Deep Learning*
326 *Works Compendium*, 2017b.

327 M. Nickel and D. Kiela. Poincaré embeddings for learning hierarchical representations. *ICML*, 2017.

328 Feng Niu, B. Recht, C. Ré, and S.J. Wright. Hogwild!: A lock-free approach to parallelizing sgd.
329 *NeurIPS*, 2011.

330 A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, T. Killeen, Zeming Lin, N. Gimselshein, L. Antiga,
331 A. Desmaison, A. Kopf, Edward Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner,
332 Lu Fang, Junjie Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning
333 library. *NeurIPS*, 2019.

334 R. Penrose. Applications of negative dimensional tensors. *Combinatorial Mathematics and its*
335 *Applications*, 1971.

336 H. Robbins and S. Monro. A stochastic approximation method. *Pages 400-407 of The Annals of*
337 *Mathematical Statistics.*, 1951.

338 D.A. Roberts. Sgd implicitly regularizes generalization error. *NeurIPS: Integration of Deep Learning*
339 *Theories Workshop*, 2018.

340 G.-C. Rota. Theory of möbius functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte*
341 *Gebiete*, 1964.

342 N.L. Roux, Y. Bengio, and A. Fitzgibbon. Improving first and second-order methods by modeling
343 uncertainty. *Book Chapter: Optimization for Machine Learning, Chapter 15*, 2012.

344 C. Stein. Inadmissibility of the usual estimator for the mean of a multivariate normal distribution.
345 *Berkeley Symposium on Mathematical Probability*, 1956.

346 E. Strubell, A. Ganesh, and A. McCallum. Energy and policy considerations for deep learning in nlp.
347 *ACL*, 2019.

348 M.P. Vitruvius. De architectura (book 10, chapter 6). *Self-published*, circa 10^{1/2} b.c.e.

349 Huan Wang, N.S. Keskar, Caiming Xiong, and R. Socher. Identifying generalization properties in
350 neural networks. *Arxiv Preprint*, 2018.

351 Mingwei Wei and D.J. Schwab. How noise affects the hessian spectrum in overparameterized neural
352 networks. *Arxiv Preprint*, 2019.

353 P. Werbos. Beyond regression: New tools for prediction and analysis. *Harvard Thesis*, 1974.

354 Lei Wu, Chao Ma, and Weinan E. How sgd selects the global minima in over-parameterized learning.
355 *NeurIPS*, 2018.

356 Han Xiao, L. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine
357 learning algorithms. *Arxiv Preprint*, 2017.

358 Sho Yaida. Fluctuation-dissipation relations for sgd. *ICLR*, 2019a.

359 Sho Yaida. A first law of thermodynamics for sgd. *Personal Communication*, 2019b.

- 360 Chiyuan Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires
361 rethinking generalization. *ICLR*, 2017.
- 362 Hongyi Zhang, S.J. Reddi, and S. Sra. Fast stochastic optimization on riemannian manifolds. *NeurIPS*,
363 2016.

364 **Organization of the appendices**

365 The following three appendices serve three respective functions:

- 366 • to explain how to calculate using diagrams;
- 367 • to prove our results (and pose a conjecture);
- 368 • to specify our experimental methods and results.

369 In more detail, we organize the appendices as follows.

371	A How to calculate test losses	page 13
372	A.1 An example calculation: the effect of epochs	13
373	A.2 How to identify the relevant space-time	15
374	A.3 How to identify the relevant diagram embeddings	16
375	A.4 How to evaluate each embedding	17
376	A.5 How to sum the embeddings' values	18
377	A.6 Interpreting diagrams intuitively	19
378	A.7 How to solve variant problems	20
379	A.8 Do diagrams streamline computation?	21
380	B Mathematics of the theory	page 22
381	B.1 Assumptions	22
382	B.2 A key lemma à la Dyson	22
383	B.3 From Dyson to diagrams	23
384	B.4 Interlude: a review of Möbius inversion	25
385	B.5 Theorems 1 and 2	25
386	B.6 How to modify proofs to handle variants	26
387	B.7 Proofs of corollaries	26
388	B.8 Future topics	28
389	C Experimental methods	page 29
390	C.1 What artificial landscapes did we use?	29
391	C.2 What image-classification landscapes did we use?	30
392	C.3 Measurement process	30
393	C.4 Implementing optimizers	30
394	C.5 Software frameworks and hardware	31
395	C.6 Unbiased estimators of landscape statistics	31
396	C.7 Additional figures	32

397 A How to calculate test losses

398 Our work introduces a novel technique for calculating the expected learning curves of SGD in terms of
 399 statistics of the loss landscape near initialization. Here, we explain this technique. New combinatorial
 400 objects — *space-time grids* — arise as we relax the paper body’s assumption that $E = B = 1$. This,
 401 too, we will explain. We note for now that there are **four steps** to computing the expected test
 402 loss, or other quantities of interest, after a specific number of gradient updates:

- 403 • **Specify, as a space-time grid**, the batch size, training set size, and number of epochs.
- 404 • **Draw embeddings**, of diagrams into the space-time grid, as needed for the desired precision.
- 405 • **Evaluate each diagram embedding**, whether exactly (via what we will call *rvalues*) or
 406 roughly (via what we will call *uvalues*).
- 407 • **Sum the embeddings’ values** to obtain the quantity of interest as a function of η .

408 After, presenting an example calculation that follows these four steps, we detail each step individually.
 409 Though we focus on the computation of expected test losses, we describe how the four steps also
 410 give us variances, train losses, of weight displacements.

411 A.1 An example calculation: the effect of epochs

412 **Question 1.** *How does multi-epoch SGD differ from single-epoch SGD? Specifically, what is the*
 413 *difference between the expected test losses of the following two versions of SGD?*

- 414 • SGD over $T = M_0 \times N$ time steps, learning rate η_0/M , and batch size $B = 1$
- 415 • SGD over $T = N$ time steps, learning rate η_0 , and batch size $B = 1$

416 We seek an answer expressed in terms of the landscape statistics at initialization: G, H, C, \dots .

417 To make our discussion concrete, we will set $M_0 = 2$; our analysis generalizes directly to larger M_0 .

418 We scaled the above two versions of SGD deliberately, to create an interesting comparison. Specif-
 419 ically, on a noiseless linear landscape $l_x = l \in (\mathbb{R}^n)^*$, the versions attain equal test losses, namely
 420 $l(\theta_0) - T l_\mu \eta^{\mu\nu}$. So Question 1’s answer will be second-order (or higher-order) in η .

421 A.1.1 Space-time grids

422 We take an $N \times T$ grid and shade its cells, shading the (n, t) th cell when the t th update involves the n th
 423 data point. Thus, each column contains B (batch size) many shaded cells and that each row contains
 424 E (epoch number) many shaded cells. The shaded grid is SGD’s *space-time*. Two space-times are
 relevant to Question 1: one for multi-epoch SGD and another for single-epoch SGD — see Figure 5.

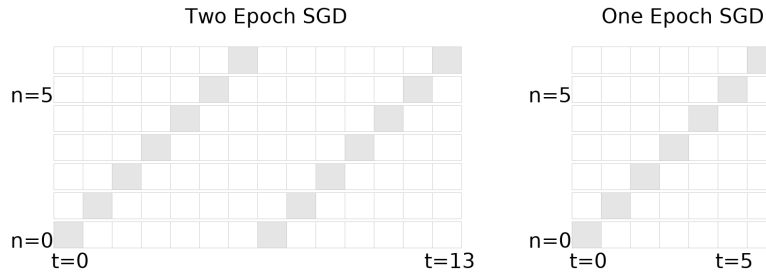







Figure 5: **The space-time grids of single-epoch and of multi-epoch SGD.** A cell at row n and column t is shaded provided that the n th training sample inhabits the t th batch. Both grids depict $N = 7$ training points and batch size $B = 1$; neither depicts training-set permutation between epochs.
Left: SGD with $M = 2$ update per training sample for a total of $T = MN = 2N$ many updates.
Right: SGD with $M = 1$ update per training sample for a total of $T = MN = N$ many updates.

425

426 A.1.2 Embeddings of diagrams into space-time

427 There are four two-edged diagrams: , , , and . We permit the diagram , which violates the path condition mentioned in §2, because we are no longer restricting to the special case $E = B = 1$. An *embedding* of a diagram D into a space-time grid is an assignment of D 's non-root nodes to shaded cells (n, t) that obeys the following two criteria:

- 431 • **time-ordering condition:** the times t strictly increase along each path from leaf to root; and
- 432 • **correlation condition:** if two nodes are in the same part of D 's partition, then they are
- 433 assigned to the same datapoint n .

434 We may conveniently draw embeddings by placing nodes in the shaded cells to which they are

435 assigned. Figure 6 shows some embeddings of order-1 and order-2 diagrams (i.e. one-edged and two-edged diagrams) into the space-times relevant to Question 1.

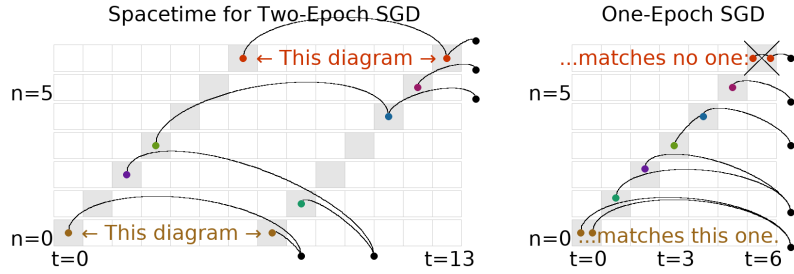





















Figure 6: **The diagram  embeds into multi-epoch but not single-epoch space-time.** Drawn on each of the two grids are examples of embeddings. The black nodes external to the grids are positioned arbitrarily. From top to bottom in each grid, the five diagrams embedded are , , , , and  (or , , , and ). The diagram  may be embedded wherever the diagram  may be embedded, but not vice versa. Likewise for  and . **Left:**  embeds into multi-epoch space-time. **Right:**  cannot embed into single-epoch space-time. Indeed, the correlation condition forces both red nodes into the same row and thus the same cell, while the time-ordering condition forces the red nodes into distinct columns and thus distinct cells.


436

437 A.1.3 Values of the embeddings

438 We choose to compute uvalues instead of rvalues. The former are an approximation of the latter, appropriate when T is fixed instead of taken to infinity. uvalues have the same asymptotic error as rvalues with respect to η . Moreover, uvalues are simpler to calculate, since their numeric values depend only on diagrams, not on embeddings. So to compute a test loss, we will multiply each diagram's uvalue by the number of ways that diagram embeds.



443 Figure 6 shows us that the diagram  embeds similarly into multi-epoch and single-epoch spacetimes. More precisely, its multi-epoch embeddings correspond by a $M_0^2 : 1$ map to its single-epoch embeddings. Since we scaled the learning rate of the two SGD versions by a factor of M_0 , and since 2-edged diagrams such as  scale as η^2 , the total uvalue of the diagram's multi-epoch embeddings will match the total uvalue of the diagram's single-epoch embeddings. In fact, Figure 6 shows that this cancellation happens for all of the order-2 diagrams *except* for . Therefore, to

449 second order, the answer to Question 1 will be (some multiple of) $\text{uvalue}(\text{diagram 5})$.

To compute 's value, we follow the rules in Section 2; the edge rule for uvalues is that each edge becomes an η . So

$$\text{uvalue}(\text{diagram 5}) = \mathbb{E} \left[\nabla_{\mu} l_x \nabla_{\nu} \nabla_{\lambda} l_x \right] \mathbb{E} \left[\nabla_{\rho} l_x \right] \eta^{\mu\lambda} \eta^{\nu\rho} = (\nabla_{\nu} C_{\mu\lambda} / 2) G^{\rho} \eta^{\mu\lambda} \eta^{\nu\rho}$$

450 A.1.4 Sum of the values

451 Referring again to Figure 6, we see that  has $\binom{M_0}{2}N$ many embeddings into the multi-epoch
 452 space-time (one embedding per pair of distinct epochs, per row) — and no embeddings into the
 453 single-epoch space-time. Moreover, each embedding of  has $|\text{Aut}_f(D)| = 1$. Now we plug into
 454 the overall formula for test loss:

$$\sum_{\substack{D \text{ a} \\ \text{diagram}}} \sum_{\substack{f \text{ an embed-} \\ \text{-ding of } D}} \frac{(-B)^{-|\text{edges}(D)|}}{|\text{Aut}_f(D)|} \text{uvalue}(D)$$

We conclude that the test loss of $M = M_0$ SGD exceeds the test loss of $M = 1$ SGD by this much:

$$\binom{M_0}{2}N \cdot \frac{(-1)^2}{1} \cdot (\nabla_\nu C_{\mu\lambda}/2) G^\rho \eta^{\mu\lambda} \eta^{\nu\rho} + o(\eta^2)$$

Since Question 1 defines $\eta^2 = \eta_0^2/M_0^2$, we can rewrite our answer as:

$$l(\theta_{M=M_0, \eta=\eta_0/M_0}) - l(\theta_{M=1, \eta=\eta_0}) = \frac{M_0 - 1}{4M_0} N \cdot G^\nu (\nabla_\nu C_\mu^\mu) + o(\eta_0^2)$$

455 where we use η_0 to raise indices. This completes the example problem.

456 **Remark.** An essentially similar argument proves Corollary 4. ◇

457 A.2 How to identify the relevant space-time

458 Diagrams tell us about the loss landscape but not about SGD’s batch size, number of epochs, and
 459 training set size. We encode this SGD data as a set of pairs (n, t) , where we have one pair for each
 460 participation of the n th datapoint in the t th update. For instance, full-batch GD has NT many pairs,
 461 and singleton-batch SGD has T many pairs. We will draw these (n, t) pairs as shaded cells in an
 462 $N \times T$ grid; we will call the shaded grid the SGD’s *space-time grid* or *space-time*. See Figure 7.

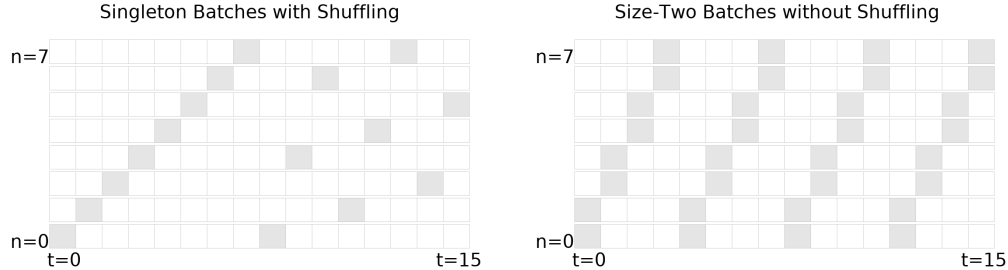


Figure 7: **The space-time grids of two SGD variants.** Shaded cells show (n, t) pairs (see text).

Left: Two epoch SGD with batch size one. The training set is permuted between epochs.





Right: Four epoch SGD with batch size two. The training set is not permuted between epochs.








463 In sum, when using the diagram method to solve a problem relating to SGD with batch size B and E
 464 many epochs (over T many time steps and on N many training samples), one shades the cells of an
 465 $N \times T$ grid with B shaded cells per column and E shaded cells per row.

466 **Remark.** A space-time grid also depicts the shuffling of training sets between epochs. Since each grid
 467 commits to a concrete sequence of training set permutations, we may analyze SGD with randomized
 468 permutations by taking expectations over multiple space-time grids. However, all of the corollaries in
 469 this text are invariant to inter-epoch training set permutations, so we will not focus on this point. ¹ ◇

¹ A routine check shows that for fixed T , inter-epoch shuffling yields only an $o(\eta^3)$ effect on test losses.

470 A.3 How to identify the relevant diagram embeddings

471 A *diagram* is a finite rooted tree equipped with a partition of its nodes, such that the root node
 472 occupies a part of size 1. Note that this definition generalizes the special case reported in the paper
 473 body; in particular, we no longer require the paper body’s “path condition” to hold. For example,
 474 there are four diagrams with two edges: , , , and . As always, we specify a
 475 diagram’s root by drawing it rightmost.

476 A diagram is *irreducible* when each of its degree-2 nodes is in a part of size one. Intuitively,
 477 this rules out multi-edge chains unadorned by fuzzy ties. Thus, only the first diagram in the
 478 list , , , \dots is irreducible. Only the first diagram in the list , , \dots is
 479 irreducible. Only the first diagram in the list , , \dots is irreducible.

480 An *embedding* of a diagram D into a space-time grid is an assignment of D ’s non-root nodes to
 481 shaded cells (n, t) that obeys the following two criteria:

- 482 • **time-ordering condition:** the times t strictly increase along each path from leaf to root; and
- 483 • **correlation condition:** if two nodes are in the same part of D ’s partition, then they are
 484 assigned to the same datapoint n .

485 We may conveniently draw embeddings by placing nodes in the shaded cells to which they are
 486 assigned. Then, the time-ordering condition forbids (among other things) intra-cell edges, and the
 487 correlation condition demands that fuzzily tied nodes are in the same row. See Figure 8.

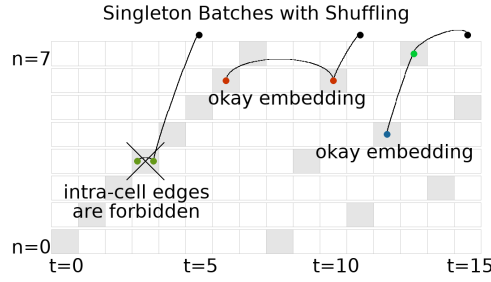











Figure 8: Embeddings, legal and illegal. **Left:** illegal embedding of , since the time-ordering condition is not obeyed. For the same reason, not a legal embedding of . **Middle:** an embedding of . Also an embedding of , since the correlation condition is obeyed. **Right:** a legal embedding of . Not an embedding of , since the correlation condition is not obeyed.

488 In principle, the relevant diagrams for a calculation with error $o(\eta^d)$ are the diagrams with at most d
 489 edges. For d greater than 2, there will be many such diagrams. However, in practice we gain insight
 490 even from considering one diagram at a time:

491 **Remark.** In this paper’s corollaries, we seek to extract the specific effect of a specific landscape or
 492 optimization feature such as skewed noise (Example 1) or multiple epochs (§A.1). In these cases, it
 493 is usually the case that most diagrams are irrelevant. For example, because a diagram evaluates to a
 494 product of its components, the only way the skewness of gradient noise can appear in our calculations
 495 is through diagrams such as  that have a part of size 3. Thus, the analysis in Example 1
 496 was able to ignore diagrams such as . Likewise, in §A.1 we argued by considering which
 497 embeddings that the only diagram relevant to Question 1 is . \diamond

498 In sum, when using the diagram method to analyze how a quantity affects SGD to order $o(\eta^d)$, we
 499 must consider all diagrams with d or fewer edges that include that quantity as a component and that
 500 have a non-zero number of embeddings into the relevant space-time. If we are using rvalues (see next
 501 section for discussion of rvalues and uvalues), then we consider only the irreducible diagrams. For

each diagram, we must enumerate the embeddings, i.e. the assignments of the diagram's nodes to space-time cells that obey both the time-ordering condition and correlation condition.

Here are some further examples. Table 1 shows the 6 diagrams that may embed into the space-time grid of $E = B = 1$. It shows each diagram in multiple ways to underscore that diagrams are purely topological and to suggest the ways in which these diagrams may embed into space-time.

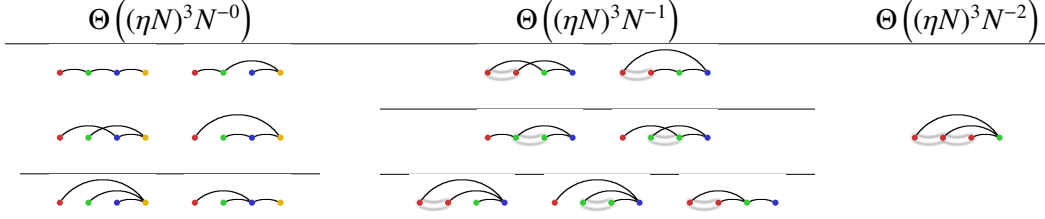


Table 1: **Multiple ways to draw the 6 distinct degree-3 diagrams for $B = E = 1$ SGD's test loss.** Because the space-time of $B = E = 1$ SGD has only one cell per row and one cell per column, the only diagrams that have a non-zero number of embeddings are the diagrams that obey §2's path condition. We show $(4 + 2) + (2 + 2 + 3) + (1)$ ways to draw the 6 diagrams. In fact, these drawings show all of the time-orderings of the diagrams' nodes that are consistent with the time-ordering condition. **Organization:** We organize the diagrams into columns by the number of parts in their partitions. Because partitions (fuzzy outlines) indicate correlations between nodes (i.e. noise), diagrams with fuzzy outlines show deviations of SGD away from deterministic ODE. The big- Θ notation that heads the columns gives the asymptotics of the sum-over-embeddings of each diagram's uvalues (for N large and η small even relative to $1/N$). **Left:** Diagrams for ODE behavior. **Center:** 1st order deviation of SGD away from ODE. **Right:** 2nd order deviation of SGD from ODE with appearance of non-Gaussian statistics.

506

507 A.4 How to evaluate each embedding

508 We will discuss how to compute both rvalues and uvalues. Both are ways of turning a diagram
 509 embedding into a number. The paper body mainly mentions rvalues, uvalues are simpler to calculate,
 510 since they depend only on a diagram's topology, not on the way it is embedded. rvalues are more
 511 accurate; in particular, when we initialize near a local minimum, rvalues do not diverge to $\pm\infty$ as
 512 $T \rightarrow \infty$.

513 A.4.1 Un-resummed values: uvalue(D)

514 Each part in a diagram's partition looks like one of the following fragments (or one of the infinitely
 515 many analogous fragments):

$G \triangleq \mathbb{E}_x [\nabla l_x(\theta)] \triangleq \text{↯}$	$C \triangleq \mathbb{E}_x [(\nabla l_x(\theta) - G)^2] \triangleq \text{↯↯}$
$H \triangleq \mathbb{E}_x [\nabla \nabla l_x(\theta)] \triangleq \text{↯↯}$	$S \triangleq \mathbb{E}_x [(\nabla l_x(\theta) - G)^3] \triangleq \text{↯↯↯}$
$J \triangleq \mathbb{E}_x [\nabla \nabla \nabla l_x(\theta)] \triangleq \text{↯↯↯}$	$\mathbb{E}_x [(\nabla l_x(\theta) - G)(\nabla \nabla l_x(\theta) - H)] \triangleq \text{↯↯↯}$
$\mathbb{E}_x [(\nabla l_x(\theta) - G)(\nabla \nabla l_x(\theta) - H)] \triangleq \text{↯↯↯}$	$\mathbb{E}_x [(\nabla l_x(\theta) - G)^4] - 3C^2 \triangleq \text{↯↯↯↯}$
$\mathbb{E}_x [(\nabla l_x(\theta) - G)(\nabla \nabla \nabla l_x(\theta) - J)] \triangleq \text{↯↯↯↯}$	$\mathbb{E}_x [(\nabla l_x(\theta) - G)^5] - 10CS \triangleq \text{↯↯↯↯↯}$

517 The above examples illustrate the **Node rule**: each degree d node evaluates to $\nabla^d l_x$.

518 Fuzzy outlines dictate how to collect the $\nabla^d l_x$ s into expectation brackets. For example, we could
 519 collect the nodes within each part (of the partition) into a pair of expectation brackets $\mathbb{E}_x[\cdot]$ — call the
 520 result the **moment value**. However, this would yield (un-centered) moments such as $\mathbb{E}_x[(\nabla l_x(\theta))^2]$
 521 instead of cumulants such as $C = \mathbb{E}_x[(\nabla l_x(\theta) - G)^2]$. For technical reasons explained in §B.4 and
 522 §B.5, cumulants will be easier to work with than moments, so we will choose to define the values of
 523 diagrams slightly differently as follows. **Outline rule:** a partition on nodes evaluates to the difference
 524 $X - Y$, where X is the moment-value of the partition and Y is the sum of all strictly finer partitions.
 525 This is just the standard Möbius recursion for defining cumulants (see Rota [1964]).
 526 **Example 2.** For example, if we denote moment values by solid gray fuzzy ties (instead of fuzzy
 527 outlines), then:

$$\begin{aligned} & \text{Diagram with two nodes and two edges (one solid gray, one dashed)} \triangleq \text{Diagram with two nodes and one edge (solid gray)} - \text{Diagram with two nodes and one edge (dashed)} \\ & \triangleq \text{Diagram with two nodes and one edge (solid gray)} - \text{Diagram with two nodes and one edge (dashed)} - \text{Diagram with two nodes and one edge (dashed)} + 2 \cdot \text{Diagram with two nodes and one edge (dashed)} \end{aligned}$$

528 We will not use the phrase “moment value” after this example. ◇

529 Finally, we come to edges. **Edge rule:** insert a factor of $\eta^{\mu\nu}$ for each edge. The indices μ, ν should
 530 match the corresponding indices of the two nodes incident to the edge.

Example 3 (Un-resummed value). Remember that $\text{Diagram with two nodes and one edge (solid gray)} = C_{\mu\nu}$ and $\text{Diagram with two nodes and one edge (dashed)} = H_{\lambda\rho}$, so that $\text{Diagram with two nodes and two edges} = C_{\mu\nu}H_{\lambda\rho}$. Then

$$\text{uvalue}(\text{Diagram with two nodes and two edges}) = C_{\mu\nu}H_{\lambda\rho}\eta^{\mu\lambda}\eta^{\nu\rho}$$

531 Here, $\text{Diagram with two nodes and two edges}$ has two edges, which correspond in this example to the tensor contractions via $\eta^{\mu\lambda}$ and
 532 via $\eta^{\nu\rho}$, respectively. ◇

533 A.4.2 Resummed values: $\text{rvalue}_f(D)$

534 The only difference between rvalues and uvalues is in their rule for evaluating edges.

535 **Edge rule:** if an edge’s endpoints are embedded to times t, t' , insert a factor of $K^{|t'-t|-1}\eta$, where
 536 $K \triangleq (I - \eta H)$. Here, we consider the root node as embedded to the time T .

Example 4 (Re-summed value). Recall as in Example 3 that $\text{Diagram with two nodes and one edge (solid gray)} = C_{\mu\nu}$ and $\text{Diagram with two nodes and one edge (dashed)} = H_{\lambda\rho}$, so that
 $\text{Diagram with two nodes and two edges} = C_{\mu\nu}H_{\lambda\rho}$. Then if f is an embedding of $\text{Diagram with two nodes and two edges}$ that sends the diagram’s red part to a time
 t (and its green root to T), we have:

$$\text{rvalue}_f(\text{Diagram with two nodes and two edges}) = C_{\mu\nu}H_{\lambda\rho} \left(K^{T-t-1}\eta \right)^{\mu\lambda} \left(K^{T-t-1}\eta \right)^{\nu\rho}$$

537 Here, $\text{Diagram with two nodes and two edges}$ has two edges, which correspond in this example to the tensor contractions via $(K^{\cdots}\eta)^{\mu\lambda}$
 538 and via $(K^{\cdots}\eta)^{\nu\rho}$, respectively. ◇

539 A.4.3 Overall

540 In sum, we evaluate an embedding of a diagram by using the **node**, **outline**, and **edge** rules to build
 541 an expression of $\nabla^d l_x$ s, \mathbb{E}_x s and η s. The difference between uvalues and rvalues lies only in their
 542 edge rule.

543 A.5 How to sum the embeddings’ values

544 Theorem 1 in the paper body generalizes to

545 **Theorem.** For any T : for η small enough, SGD has expected test loss

$$\sum_{D \text{ an irreducible diagram}} \sum_{f \text{ an embedding of } D} \frac{(-B)^{-|\text{edges}(D)|}}{|\text{Aut}_f(D)|} \text{rvalue}_f(D)$$

546 which is the same as

$$\sum_{D \text{ a diagram}} \sum_{f \text{ an embedding of } D} \frac{(-B)^{-|\text{edges}(D)|}}{|\text{Aut}_f(D)|} \text{uvalue}(D)$$

547 Here, B is the batch size.

548 How do we evaluate the above sum? Summing uvalues reduces to counting embeddings, which in all
 549 the applications reported in this text is a routine combinatorial exercise. However, when summing
 550 rvalues, it is often convenient to replace a sum over embeddings by an integral over times, and the
 551 power $(I - \eta H)^{\Delta t - 1}$ by the exponential $\exp -\Delta t \eta H$. This incurs a term-by-term $1 + o(\eta)$ error factor,
 552 meaning that it preserves leading order results.

Example 5. Let us return to $D = \text{red arc with green dot}$, embedded, say, in the space-time of one-epoch one-
 sample-per-batch SGD. From Example 4, we know that we want to sum the following value over
 all embeddings f , i.e. over all $0 \leq t < T$ to which the red part of the diagram's partition may be
 assigned:

$$\text{rvalue}_f(\text{red arc with green dot}) = C_{\mu\nu} (K^{T-t-1} \eta)^{\mu\lambda} (K^{T-t-1} \eta)^{\nu\rho} H_{\lambda\rho}$$

553 Each embedding has a factor $(-B)^{-|\text{edges}(D)|} / |\text{Aut}_f(D)| = (-B)^{-2}/2$; we will multiply in this factor at
 554 the end so we now we focus on the \sum_f . So, using the aforementioned approximation, we seek to
 555 evaluate

$$\int_{0 \leq t < T} dt C_{\mu\nu} (\exp(-(T-t)\eta H) \eta)^{\mu\lambda} (\exp(-(T-t)\eta H) \eta)^{\nu\rho} H_{\lambda\rho} =$$

$$C_{\mu\nu} \left(\int_{0 \leq t < T} dt \exp(-(T-t)((\eta H) \otimes I + I \otimes (\eta H)))^{\mu\nu}_{\pi\sigma} \right) \eta^{\pi\lambda} \eta^{\sigma\rho} H_{\lambda\rho}$$

556 We know from linear algebra and calculus that $\int_{0 \leq u < T} du \exp(-uA) = (I - \exp(-TA))/A$ (when A is
 557 a non-singular linear endomorphism). Applying this rule for $u = T - t$ and $A = (\eta H) \otimes I + I \otimes (\eta H)$,
 558 we evaluate the integral as:

$$\dots = C_{\mu\nu} \left(\frac{I - \exp(-T((\eta H) \otimes I + I \otimes (\eta H)))}{(\eta H) \otimes I + I \otimes (\eta H)} \right)^{\mu\nu}_{\pi\sigma} \eta^{\pi\lambda} \eta^{\sigma\rho} H_{\lambda\rho}$$

559 This is perhaps easier to write in an eigenbasis of ηH :

$$\dots = \sum_{\mu\nu} C_{\mu\nu} \frac{1 - \exp(-T((\eta H)_\mu^\mu + (\eta H)_\nu^\nu))}{(\eta H)_\mu^\mu + (\eta H)_\nu^\nu} (\eta H \eta)^{\mu\nu}$$

560 Multiplying this expression by the aforementioned $(-B)^{-2}/2$ gives the contribution of $\text{red arc with green dot}$ to
 561 SGD's test loss. \diamond

562 In short, we sum embeddings of uvalues directly. We sum embeddings of rvalues using an integral-
 563 of-exponentials approximation along with the rule $\int_{0 \leq u < T} du \exp(-uA) = (I - \exp(-TA))/A$. When
 564 written in an eigenbasis of ηH , this A 's coefficients are sums of one or more eigenvalues of ηH (one
 565 eigenvalue for each edge involved in the relevant degrees of freedom over which we integrate). As
 566 another example, see Example 1.

567 A.6 Interpreting diagrams intuitively

568 We may intuitively interpret edges as carrying influence from the training set toward the test measure-
 569 ment. See Figure 10.

570 From this perspective, we may intuitively interpret edges in an rvalue calculation as carrying influence
 571 from the training set toward the test measurement. See Figure 10.

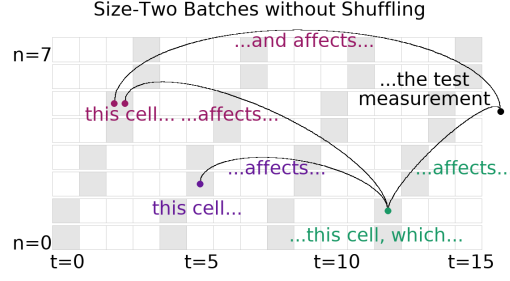


Figure 9: **Edges carry information.** Embedding of a 4-edged diagram.

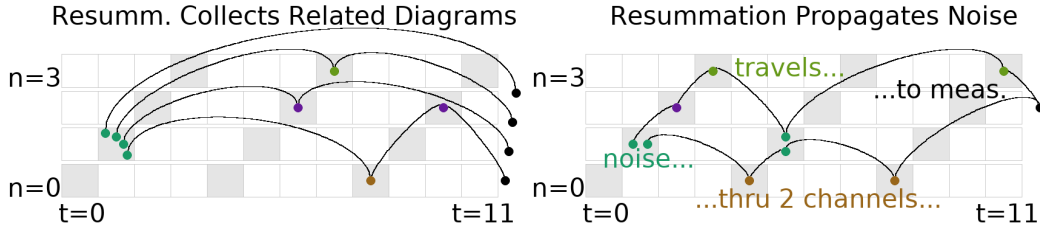




Figure 10: **Resummation propagates information, damped by curvature.** Each resummed value represents many un-resummed values, each modulated by the Hessian (∇^2) in a different way. **Left:** Here is one of many un-resummed terms captured by a single resummed embedding for ∇^2 . **Left:** each resummed value represents many un-resummed values. Here is one of many un-resummed terms captured by a single resummed embedding for ∇^2 .

572 A.7 How to solve variant problems

573 In §B.8, we briefly discuss second-order methods and natural gradient descent. Here, we briefly
 574 discuss modifications. We omit proofs, which would closely follow §B's proof of the expectation-of-
 575 test-loss case.

576 Variance (instead of expectation)

577 To compute variances instead of expectations (with respect to the noise in the training set), one
 578 considers generalized diagrams that have “two roots” instead of one. More precisely, to compute, say,
 579 the un-centered second moment of test loss, one uses diagrams whose edge structures are not rooted
 580 trees but instead forests consisting of two rooted trees. As in the case of test loss expectations, we
 581 require that the set of roots (now a set of size two instead of size one) is a part of the diagram's partition.



582 We draw the two roots rightmost. For example, the generalized diagrams  or  may
 583 appear in this computation.

584 Measuring on the training (instead of test) set

585 To compute the training loss, we compute with all the same diagrams as the test loss, and we also
 586 allow all the additional generalized diagrams that violate the constraint that a diagram's root should
 587 be in a part of size one. Therefore, to compute the generalization gap (i.e. test loss minus training
 588 loss), we sum over all the diagrams that expressly violate this constraint (and then, since gen. gp is
 589 test minus train instead of train minus test, we multiply the whole answer by -1). For example, the

590 generalized diagrams  or  may appear in this computation.

591 **Weight displacement (instead of loss)**

592 To compute displacements instead of losses, one considers generalized diagrams that have a “loose
593 end” instead of a root. For example, the generalized diagrams  or  may appear in this
594 computation.

595 **A.8 Do diagrams streamline computation?**

B Mathematics of the theory

B.1 Assumptions

We assume throughout this work the following regularity properties of the loss landscape.

Existence of Taylor Moments — we assume that each finite collection of polynomials of the 0th and higher derivatives of the l_x , all evaluated at any point θ , may be considered together as a random variable insofar as they are equipped with a probability measure upon of the standard Borel algebra.

Analyticity Uniform in Randomness — we assume that the functions $\theta \mapsto l_x(\theta)$ — and the expectations of polynomials of their 0th and higher derivatives — exist and are analytic with radii of convergence bounded from 0 (by a potentially θ -dependent function). So expectations and derivatives commute.

Boundedness of Gradients — we also assume that the gradients $\nabla l_x(\theta)$, considered as random covectors, are bounded by some continuous function of θ .¹ A metric-independent way of expressing this boundedness constraint is that the gradients all lie in some subset $\mathcal{S} \subseteq TM$ of the tangent bundle of weight space, where, for any compact $C \subseteq M$, we have that the topological pullback — of $S \hookrightarrow TM \rightarrow M$ and $C \hookrightarrow M$ — is compact.

B.2 A key lemma à la Dyson

Suppose s is an analytic function defined on the space of weights. The following Lemma, reminiscent of Dyson [1949], helps us track $s(\theta)$ as SGD updates θ :

Key Lemma. *For all T : for η sufficiently small, $s(\theta_T)$ is a sum over tuples of natural numbers:*

$$\sum_{(d_t: 0 \leq t < T) \in \mathbb{N}^T} (-\eta)^{\sum_t d_t} \left(\prod_{0 \leq t < T} \left(\frac{(g \nabla)^{d_t}}{d_t!} \Big|_{g = \sum_{n \in \mathcal{B}_t} \nabla l_n(\theta)/B} \right) \right) (s)(\theta_0) \quad (1)$$

Moreover, the expectation symbol (over training sets) commutes with the sum over ds .

Here, we consider each $(g \nabla)^{d_t}$ as a higher order function that takes in a function f defined on weight space and outputs a function equal to the d_t th derivative of f , times g^{d_t} . The above product then indicates composition of $(g \nabla)^{d_t}$'s across the different t 's. In total, that product takes the function s as input and outputs a function equal to some polynomial of s 's derivatives.

Proof of the Key Lemma. We work in a neighborhood of the initialization so that the tangent space of weight space is a trivial bundle. For convenience, we fix a coordinate system, and with it the induced flat, non-degenerate inverse metric $\tilde{\eta}$; the benefit is that we may compare our varying η against one fixed $\tilde{\eta}$. Henceforth, a “ball” unless otherwise specified will mean a ball with respect to $\tilde{\eta}$ around the initialization θ_0 . Since s is analytic, its Taylor series converges to s within some positive radius ρ ball. By assumption, every l_t is also analytic with radius of convergence around θ_0 at least some $\rho > 0$. Since gradients are x -uniformly bounded by a continuous function of θ , and since in finite dimensions the closed ρ -ball is compact, we have a strict gradient bound b uniform in both x and θ on gradient norms within that closed ball. When

$$2\eta T b < \rho \tilde{\eta} \quad (2)$$

as norms, SGD after T steps on any train set will necessarily stay within the ρ -ball.² We note that the above condition on η is weak enough to permit all η within some open neighborhood of $\eta = 0$.

Condition 2 together with analyticity of s then implies that $(\exp(-\eta g \nabla) s)(\theta) = s(\theta - \eta g)$ when θ lies in the $\tilde{\eta}$ ball (of radius ρ) and its η -distance from that $\tilde{\eta}$ ball's boundary exceeds b , and that both sides are analytic in η, θ on the same domain — and *a fortiori* when θ lies in the ball of radius $\rho(1 - 1/(2T))$.

¹ Some of our experiments involve Gaussian noise, which is not bounded and so violates the hypothesis. In practice, Gaussians are effectively bounded, on the one hand in that with high probability no standard normal sample encountered on Gigahertz hardware within the age of the universe will much exceed $\sqrt{2 \log(10^{30})} \approx 12$, and on the other hand in that our predictions vary smoothly with the first few moments of this distribution, so that a ± 12 -clipped Gaussian will yield almost the same predictions.

² The 2 ensures that SGD initialized at any point within a $\rho/2$ ball will necessarily stay within the ρ -ball.

Likewise, a routine induction through T gives the value of s (after doing T gradient steps from an initialization θ) as

$$\left(\prod_{0 \leq t < T} \exp(-\eta g \nabla) \Big|_{g=\nabla l_t(\theta)} \right) (s)(\theta)$$

for any θ in the $\rho(1 - T/(2T))$ -ball (that is, the $\rho/2$ -ball), and that both sides are analytic in η, θ on that same domain. Note that in each exponential, the ∇_v does not act on the $\nabla_\mu l(\theta)$ with which it pairs.

Now we use the standard expansion of exp. Because (by analyticity) the order d coefficients of l_t, s are bounded by some exponential decay in d that has by assumption an x -uniform rate, we have absolute convergence and may rearrange sums. We choose to group by total degree:

$$\dots = \sum_{0 \leq d < \infty} (-\eta)^d \sum_{\substack{(d_t; 0 \leq t < T) \\ \sum_t d_t = d}} \left(\prod_{0 \leq t < T} \frac{(g \nabla)^{d_t}}{d_t!} \Big|_{g=\nabla l_t(\theta)} \right) s(\theta) \quad (3)$$

The first part of the Key Lemma is proved. It remains to show that expectations over train sets commute with the above summation.

We will apply Fubini's Theorem. To do so, it suffices to show that

$$|c_d((l_t : 0 \leq t < T))| \triangleq \left| \sum_{\substack{(d_t; 0 \leq t < T) \\ \sum_t d_t = d}} \left(\prod_{0 \leq t < T} \frac{(g \nabla)^{d_t}}{d_t!} \Big|_{g=\nabla l_t(\theta)} \right) s(\theta) \right|$$

has an expectation that decays exponentially with d . The symbol c_d we introduce purely for convenience; that its value depends on the train set we emphasize using function application notation. Crucially, no matter the train set, we have shown that the expansion 3 (that features c_d appear as coefficients) converges to an analytic function for all η bounded as in condition 2. The uniformity of this demanded bound on η implies by the standard relation between radii of convergence and decay of coefficients that $|c_d|$ decays exponentially in d at a rate uniform over train sets. If the expectation of $|c_d|$ exists at all, then, it will likewise decay at that same shared rate.

Finally, $|c_d|$ indeed has a well-defined expected value, for $|c_d|$ is a bounded continuous function of a (finite-dimensional) space of T -tuples (each of whose entries can specify the first d derivatives of an l_t) and because the latter space enjoys a joint distribution. So Fubini's Theorem applies. The Key Lemma follows. \square

B.3 From Dyson to diagrams

We now describe the terms that appear in the Key Lemma. The following result looks like Theorem 1, except it has $\text{uvalue}(D)$ instead of $\text{uvalue}_f(D)$, and the sum is over all diagrams, not just irreducible ones. In fact, we will use Theorem 3 to prove Theorem 1.

Theorem 3 (Test Loss as a Path Integral). *For all T : for η sufficiently small, SGD's expected test loss is*

$$\sum_D \sum_{\text{embeddings } f} \frac{1}{|\text{Aut}_f(D)|} \frac{\text{uvalue}(D)}{(-B)^{|\text{edges}(D)|}}$$

Here, D is a diagram whose root r does not participate in any fuzzy edge, f is an embedding of D into spacetime, and $|\text{Aut}_f(D)|$ counts the graph-automorphisms of D that preserve f 's assignment of nodes to cells. If we replace D by $(-\sum_{p \in \text{parts}(D)} (D_{rp} - D)/N)$, where r is D 's root, we obtain the expected generalization gap (test minus train loss).

Theorem 3 describe the terms that appear in the Key Lemma by matching each term to an embedding of a diagram in spacetime, so that the infinite sum becomes a sum over all diagram spacetime configurations. The main idea is that the combinatorics of diagrams parallels the combinatorics of repeated applications of the product rule for derivatives applied to the expression in the Key Lemma. Balancing against this combinatorial explosion are factorial-style denominators, again from the Key Lemma, that we summarize in terms of the sizes of automorphism groups.

665 *Proof of Theorem 3.* We first prove the statement about test losses. Due to the analyticity property
 666 established in our proof of the Key Lemma, it suffices to show agreement at each degree d and train
 667 set individually. That is, it suffices to show — for each train set $(l_n : 0 \leq n < N)$, spacetime S ,
 668 function $\pi : S \rightarrow [N]$ that induces \sim , and natural d — that

$$\begin{aligned} & (-\eta)^d \sum_{\substack{(d_i: 0 \leq i < T) \\ \sum_i d_i = d}} \left(\prod_{0 \leq i < T} \frac{(g \nabla)^{d_i}}{d_i!} \Big|_{g = \nabla l_i(\theta)} \right) l(\theta) = \\ & \sum_{\substack{D \in \text{im}(\mathcal{F}) \\ \text{with } d \text{ edges}}} \left(\sum_{f: D \rightarrow \mathcal{F}(S)} \frac{1}{|\text{Aut}_f(D)|} \right) \frac{\text{uvalue}_\pi(D, f)}{B^d} \end{aligned} \quad (4)$$

669 Here, uvalue_π is the value of a diagram embedding before taking expectations over train sets. We
 670 have for all f that $\mathbb{E}[\text{uvalue}_\pi(D, f)] = \text{uvalue}(D)$. Observe that both sides of 4 are finitary sums.

Remark 3 (Differentiating Products). The product rule of Leibniz easily generalizes to higher derivatives of finitary products:

$$\nabla^{|M|} \prod_{k \in K} p_k = \sum_{v: M \rightarrow K} \prod_{k \in K} (\nabla^{|v^{-1}(k)|} p_k)$$

671 The above has $|K|^{|M|}$ many term indexed by functions to K from M .

672 We proceed by joint induction on d and S . The base cases wherein S is empty or $d = 0$ both follow
 673 immediately from the Key Lemma, for then the only embedding is the unique embedding of the
 674 one-node diagram \bullet . For the induction step, suppose S is a sequence of $\mathcal{M} = \min S \subseteq S$ followed by
 675 a strictly smaller S and that the result is proven for (\tilde{d}, \tilde{S}) for every $\tilde{d} \leq d$. Let us group by d_0 the
 676 terms on the left hand side of desideratum 4. Applying the induction hypothesis with $\tilde{d} = d - d_0$, we
 677 find that that left hand side is:

$$\begin{aligned} & \sum_{0 \leq d_0 \leq d} \sum_{\substack{\tilde{D} \in \text{im}(\mathcal{F}) \\ \text{with } d - d_0 \text{ edges}}} \frac{1}{d_0!} \sum_{\tilde{f}: \tilde{D} \rightarrow \mathcal{F}(\tilde{S})} \left(\frac{1}{|\text{Aut}_{\tilde{f}}(\tilde{D})|} \right) \cdot \\ & (-\eta)^{d_0} (g \nabla)^{d_0} \Big|_{g = \nabla l_0(\theta)} \frac{\text{uvalue}_\pi(\tilde{D}, \tilde{f})}{B^{d-d_0}} \end{aligned}$$

678 Since $\text{uvalue}_\pi(\tilde{D}, \tilde{f})$ is a multilinear product of $d - d_0 + 1$ many tensors, the product rule for derivatives
 679 tells us that $(g \nabla)^{d_0}$ acts on $\text{uvalue}_\pi(\tilde{D}, \tilde{f})$ to produce $(d - d_0 + 1)^{d_0}$ terms. In fact, $g = \sum_{m \in \mathcal{M}} \nabla l_m(\theta) / B$
 680 expands to $B^{d_0} (d - d_0 + 1)^{d_0}$ terms, each conveniently indexed by a pair of functions $\beta : [d_0] \rightarrow \mathcal{M}$
 681 and $\nu : [d_0] \rightarrow \tilde{D}$. The (β, ν) -term corresponds to an embedding f of a larger diagram D in the sense
 682 that it contributes $\text{uvalue}_\pi(D, f) / B^{d_0}$ to the sum. Here, (f, D) is (\tilde{f}, \tilde{D}) with $|(\beta \times \nu)^{-1}(n, v)|$ many
 683 additional edges from the cell of datapoint n at time 0 to the v th node of \tilde{D} as embedded by \tilde{f} .

684 By the Leibniz rule of Remark , this (β, ν) -indexed sum by corresponds to a sum over embeddings
 685 f that restrict to \tilde{f} , whose terms are multiples of the value of the corresponding embedding of D .
 686 Together with the sum over \tilde{f} , this gives a sum over all embeddings f . So we now only need to check
 687 that the coefficients for each $f : D \rightarrow S$ are as claimed.

688 We note that the (β, ν) diagram (and its value) agrees with the $(\beta \circ \sigma, \nu \circ \sigma)$ diagram (and its value)
 689 for any permutation σ of $[d_0]$. The corresponding orbit has size

$$\frac{d_0!}{\prod_{(m,i) \in \mathcal{M} \times \tilde{D}} |(\beta \times \nu)^{-1}(m, i)|!}$$

690 by the Orbit Stabilizer Theorem of elementary group theory.

691 It is thus enough to show that

$$|\text{Aut}_f(D)| = |\text{Aut}_{\tilde{f}}(\tilde{D})| \prod_{(m,i) \in \mathcal{M} \times \tilde{D}} |(\beta \times \nu)^{-1}(m, i)|!$$

We will show this by a direct bijection. First, observe that $f = \beta \sqcup \tilde{f} : [d_0] \sqcup \tilde{D} \rightarrow \mathcal{M} \sqcup \tilde{\mathcal{S}}$. So each automorphism $\phi : D \rightarrow D$ that commutes with f induces both an automorphism $\mathcal{A} = \phi|_{\tilde{D}} : \tilde{D} \rightarrow \tilde{D}$ that commutes with \tilde{f} together with the data of a map $\mathcal{B} = \phi|_{[d_0]} : [d_0] \rightarrow [d_0]$ that both commutes with β . However, not every such pair of maps arises from a ϕ . For, in order for $\mathcal{A} \sqcup \mathcal{B} : D \rightarrow D$ to be an automorphism, it must respect the order structure of D . In particular, if $x \leq_D y$ with $x \in [d_0]$ and $y \in \tilde{D}$, then we need

$$\mathcal{B}(x) \leq_D \mathcal{A}(y)$$

as well. The pairs $(\mathcal{A}, \mathcal{B})$ that thusly preserve order are in bijection with the $\phi \in \text{Aut}_f(D)$. There are $|\text{Aut}_{\tilde{f}}(\tilde{D})|$ many \mathcal{A} . For each \mathcal{A} , there are as many \mathcal{B} as there are sequences $(\sigma_i : i \in \tilde{D})$ of permutations on $\{j \in [d_0] : j \leq_D i\} \subseteq [d_0]$ that commute with \mathcal{B} . These permutations may be chosen independently; there are $\prod_{m \in \mathcal{M}} |(\beta \times \nu)^{-1}(m, i)|!$ many choices for σ_i . Claim ?? follows, and with it the correctness of coefficients.

The argument for generalization gaps parallels the above when we use $l - \sum_n l_n/N$ instead of l as the value for s . Theorem 3 is proved. \square

Remark 4 (The Case of $E = B = 1$ SGD). The spacetime of $E = B = 1$ SGD permits all and only those embeddings that assign to each part of a diagram's partition a distinct cell. Such embeddings factor through a diagram ordering and are thus easily counted using factorials per Proposition 1. That proposition immediately follows from the now-proven Theorem 3.

Proposition 1. *The order η^d contribution to the expected test loss of one-epoch SGD with singleton batches is:*

$$\frac{(-1)^d}{d!} \sum_D |\text{ords}(D)| \binom{N}{P-1} \binom{d}{d_0, \dots, d_{P-1}} \text{uvalue}(D)$$

where D ranges over d -edged diagrams. Here, D 's parts have sizes $d_p : 0 \leq p \leq P$, and $|\text{ords}(D)|$ counts the total orderings of D s.t. children precede parents and parts are contiguous.

B.4 Interlude: a review of Möbius inversion

B.5 Theorems 1 and 2

The diagrams summed in Theorem 1 and 2 may be grouped by their geometric realizations. Each nonempty class of diagrams with a given geometric realization has a unique element with minimally many edges, and in this way all and only irreducible diagrams arise.

We encounter two complications: on one hand, that the sizes of automorphism groups might not be uniform among the class of diagrams with a given geometric realization. On the other hand, that the embeddings of a specific member of that class might be hard to count. The first we handle using Orbit-Stabilizer. The second we address as described by via Möbius sums.

Proof of Theorem 1. We apply Möbius inversion (§B.4) to Theorem 3 (§B.3). The result is that chains of embeddings **FILL IN**

The difference in loss from the noiseless case is given by all the diagram embeddings with at least one fuzzy tie, where the fuzzy tie pattern is actually replaced by a difference between noisy and noiseless cases as prescribed by the preceding discussion on Möbius Sums. Beware that even relatively noiseless embeddings may have illegal collisions of non-fuzzily-tied nodes within a single spacetime (data) row. Throughout the rest of this proof, we permit such illegal embeddings of the fuzz-less diagrams that arise from the aforementioned decomposition.

Because the Taylor series for analytic functions converge absolutely in the interior of the disk of convergence, the rearrangement of terms corresponding to a grouping by geometric realizations preserves the convergence result of Theorem 3.

Let us then focus on those diagrams σ with a given geometric realization represented by an irreducible diagram ρ . By Theorem 3, it suffices to show that

$$\sum_{f: \rho \rightarrow \mathcal{S}} \sum_{\substack{\tilde{f}: \sigma \rightarrow \mathcal{S} \\ \exists i_*: f = \tilde{f} \circ i_*}} \frac{1}{|\text{Aut}_{\tilde{f}}(\sigma)|} = \sum_{f: \rho \rightarrow \mathcal{S}} \sum_{\substack{\tilde{f}: \sigma \rightarrow \mathcal{S} \\ \exists i_*: f = \tilde{f} \circ i_*}} \sum_{f = \tilde{f} \circ i} \frac{1}{|\text{Aut}_{\tilde{f}}(\rho)|} \quad (5)$$

Here, f is considered up to an equivalence defined by precomposition with an automorphism of ρ . We likewise consider \tilde{f} up to automorphisms of σ . And above, i ranges through maps that induce isomorphisms of geometric realizations, where i is considered equivalent to \hat{i} when for some automorphism $\phi \in \text{Aut}_{\tilde{f}}(\sigma)$, we have $\hat{i} = i \circ \phi$. Name as X the set of all such i s under this equivalence relation.

In equation 5, we have introduced redundant sums to structurally align the two expressions on the page; besides this rewriting, we see that equation 5's left hand side matches Theorem 3 resulting formula and that its right hand side is the desired formula of Theorem 1.

To prove equation 5, it suffices to show (for any f, \tilde{f}, i as above) that

$$|\text{Aut}_f(\rho)| = |\text{Aut}_{\tilde{f}}(\sigma)| \cdot |X|$$

We will prove this using the Orbit Stabilizer Theorem by presenting an action of $\text{Aut}_f(\rho)$ on X . We simply use precomposition so that $\psi \in \text{Aut}_f(\rho)$ sends $i \in X$ to $i \circ \psi$. Since $f \circ \psi = f$, $i \circ \psi \in X$. Moreover, the action is well-defined, because if $i \sim \hat{i}$ by ϕ , then $i \circ \psi \sim \hat{i} \circ \psi$ also by ϕ .

The stabilizer of i has size $|\text{Aut}_{\tilde{f}}(\rho)|$. For, when $i \sim i \circ \psi$ via $\phi \in \text{Aut}_{\tilde{f}}(\rho)$, we have $i \circ \psi = \phi \circ i$. This relation in fact induces a bijective correspondence: every ϕ induces a ψ via $\psi = i^{-1} \circ \phi \circ i$, so we have a map $\text{stabilizer}(i) \leftrightarrow \text{Aut}_{\tilde{f}}(\rho)$ seen to be well-defined and injective because structure set morphisms are by definition strictly increasing and because i must induce isomorphisms of geometric realizations. Conversely, every ψ that stabilizes enjoys *only* one ϕ via which $i \sim i \circ \phi$, again by the same (isomorphism and strict increase) properties. So the stabilizer has the claimed size.


Meanwhile, the orbit is all of $|X|$. Indeed, suppose $i_A, i_B \in X$. We will present $\psi \in \text{Aut}_f(\rho)$ such that $i_B \sim i_A \circ \psi$ by $\phi = \text{identity}$. We simply define $\psi = i_A^{-1} \circ i_B$, well-defined by the aforementioned (isomorphisms and strict increase) properties. It is then routine to verify that $f \circ \psi = \tilde{f} \circ i_A \circ i_A^{-1} \circ i_B = \tilde{f} \circ i_B = f$. So the orbit has the claimed size, and by the Orbit Stabilizer Theorem, the coefficients in the expansions of Theorems 1 and 3 match. \square

Proof of Theorem 2. Since we assumed Hessians are positive: for any m , the propagator $K^t = ((I - \eta H)^{\otimes m})^t$ exponentially decays to 0 (at a rate dependent on m). Since up to degree d only a finite number of diagrams exist and hence only a finite number of possible m s, the exponential rates are bounded away from 0. Moreover, for any fixed t_{big} , the number of diagrams — involving no exponent t exceeding t_{big} — is eventually constant as T grows. Meanwhile, the number involving at least one exponent t exceeding that threshold grows polynomially in T (with degree d). The exponential decay of each term overwhelms the polynomial growth in the number of terms, and the convergence statement follows. \square

B.6 How to modify proofs to handle variants

B.7 Proofs of corollaries



B.7.1 Corollary 1

Proof. The relevant irreducible diagram is  colored (amputated as in the previous subsection). An embedding of this diagram into $E = B = 1$ SGD's spacetime is determined by two durations — t from red to green and \tilde{t} from green to blue — obeying $t + \tilde{t} \leq T$. The automorphism group of each embedding has size 2: identity or switch the red nodes. So the answer is:

$$C_{\mu\nu} J_{\sigma}^{\rho\lambda} \left(\int_{t+\tilde{t} \leq T} (\exp(-t\eta H)\eta)^{\mu\rho} (\exp(-\tilde{t}\eta H)\eta)^{\nu\lambda} (\exp(-\tilde{t}\eta H)\eta)^{\sigma\pi} \right)$$

Standard calculus then gives the desired result. \square

B.7.2 Corollary 2's first part

Proof. The relevant irreducible diagram is  (which equals  because we are at a test minimum). This diagram has one embedding for each pair of same-row shaded cells, potentially identical, in spacetime; for GD, the spacetime has every cell shaded, so each *non-decreasing* pair of


durations in $[0, T]^2$ is represented; the symmetry factor for the case where the cells is identical is $1/2$, so we lose no precision by interpreting a automorphism-weighted sum over the *non-decreasing* pairs as half of a sum over all pairs. Each of these may embed into N many rows, hence the factor below of N . The two integration variables (say, t, \tilde{t}) separate, and we have:

$$\frac{N}{B^{\text{degree}}} \frac{C_{\mu\nu}}{2} \int_t (\exp(-t\eta H))^\mu_\lambda \int_{\tilde{t}} (\exp(-\tilde{t}\eta H))^\nu_\rho \eta^{\lambda\sigma} \eta^{\rho\pi} H_{\sigma\pi}$$

764 Since for GD we have $N = B$ and we are working to degree 2, the prefactor is $1/N$. Since $\int_t \exp(at) =$
 765 $(I - \exp(-aT))/a$, the desired result follows. \square

766 B.7.3 Corollary 2's second part

767 We apply the generalization gap modification (described in §??) to Theorem 1's result about test
 768 losses.

Proof. The relevant irreducible diagram is . This diagram has one embedding for each shaded cell of spacetime; for GD, the spacetime has every cell shaded, so each duration from 0 to T is represented. So the generalization gap is, to leading order,

$$+ \frac{C_{\mu\nu}}{N} \int_t (\exp(-t\eta H))^\mu_\lambda \eta^{\lambda\nu}$$

769 Here, the minus sign from the gen-gap modification canceled with the minus sign from the odd power
 770 of $-\eta$. Integration finishes the proof. \square







771 B.7.4 Corollaries 4 and 3

772 Corollary 4 and the first part of Corollary 3 follow from plugging appropriate values of M, N, B into
 773 the following proposition.

774 **Proposition 2.** *To order η^2 , the test loss of SGD — on N samples for M epochs with batch size B*
 775 *dividing N and with any shuffling scheme — has expectation*

$$\begin{aligned} l - MNG_\mu G^\mu + MN \left(MN - \frac{1}{2} \right) G_\mu H^\mu_\nu G^\nu \\ + MN \left(\frac{M}{2} \right) C_{\mu\nu} H^{\mu\nu} + MN \left(\frac{M - \frac{1}{B}}{2} \right) (\nabla_\mu C^\nu_\nu) G^\mu / 2 \end{aligned}$$

776 *of Proposition 2.* To prove Proposition 2, we simply count the embeddings of the diagrams, noting
 777 that the automorphism groups are all of size 1 or 2. Since we use fuzzy outlines instead of fuzzy ties,
 778 we allow untied nodes to occupy the same row, since the excess will be canceled out by the term
 779 subtract in the definition of fuzzy outlines. See Table B.7.4.


diagram	embed.s w/ $ \text{Aut}_f = 1$	embed.s w/ $ \text{Aut}_f = 2$
	1	0
	MNB	0
	$\binom{MNB}{2}$	0
	$N \binom{MB}{2}$	0
	$\binom{MNB}{2}$	0
	$N \binom{MB}{2}$	MNB

780 \square

781 *Proof of Corollary 3's second part.* **FILL IN** \square

782 B.7.5 Corollary 5

783 The corollary’s first part follows immediately from Remark ?? in the case that $d = 2$, $P = 2$, and
 784 $(\eta N)^d$ is considered fixed while N^{P-d-1} is considered changing.

Proof of second part. Because $\mathbb{E}[\nabla l]$ vanishes at initialization, all diagrams with a degree-one vertex that is a singleton vanish. Because we work at order η^3 , we consider 3-edged diagrams. Finally, because all first and second moments match between the two landscapes, we consider only diagrams with at least one partition of size at least 3. The only such test diagram is . This embeds in T ways (one for each spacetime cell) and has symmetry factor $1/3!$ for a total of

$$\frac{T\eta^3}{6} \mathbb{E}[\nabla^3 l] \mathbb{E}[\nabla l_{n_a} \nabla l_{n_b} \nabla l_{n_c}]$$

785

□

786 B.8 Future topics

787 Our diagrams invite exploration of Lagrangian formalisms and curved backgrounds:¹

788 **Question 2.** *Does some least-action principle govern SGD; if not, what is an essential obstacle to*
 789 *this characterization?*

790 Lagrange’s least-action formalism intimately intertwines with the diagrams of physics. Together,
 791 they afford a modular framework for introducing new interactions as new terms or diagram nodes. In
 792 fact, we find that some *higher-order* methods — such as the Hessian-based update $\theta \leftarrow \theta - (\eta^{-1} +$
 793 $\lambda \nabla \nabla l_t(\theta))^{-1} \nabla l_t(\theta)$ parameterized by small η, λ — admit diagrammatic analysis when we represent the
 794 λ term as a second type of diagram node. Though diagrams suffice for computation, it is Lagrangians
 795 that most deeply illuminate scaling and conservation laws.

796 Our work assumes a flat metric $\eta^{\mu\nu}$, but it might generalize to weight spaces curved in the sense of
 797 Riemann.² Such curvature finds concrete application in the *learning on manifolds* paradigm of Absil
 798 et al. [2007], Zhang et al. [2016], notably specialized to Amari [1998]’s *natural gradient descent*
 799 and Nickel and Kiela [2017]’s *hyperbolic embeddings*. While that work focuses on *optimization* on
 800 curved weight spaces, in machine learning we also wish to analyze *generalization*. Starting with the
 801 intuition that “smaller” hypothesis classes generalize better and that curvature controls the volume of
 802 small neighborhoods, we conjecture that sectional curvature regularizes learning:

803 **Conjecture 1** (Sectional curvature regularizes). *If $\eta(\tau)$ is a Riemann metric on weight space, smoothly*
 804 *parameterized by τ , and if the sectional curvature through every 2-form at θ_0 increases as τ grows,*
 805 *then the gen. gap attained by fixed- T SGD with learning rate $c\eta(\tau)$ (when initialized from θ_0)*
 806 *decreases as τ grows, for all sufficiently small $c > 0$.*

807 We are optimistic our formalism may resolve conjectures such as above.

¹ Landau and Lifshitz [1960, 1951] review these concepts.

² One may represent the affine connection as a node, thus giving rise to non-tensorial and hence gauge-dependent diagrams.

808 C Experimental methods

809 C.1 What artificial landscapes did we use?

810 We define three artificial landscapes, called GAUSS, ARCHIMEDES, and MEAN ESTIMATION.

811 C.1.1 GAUSS

Consider fitting a centered normal $\mathcal{N}(0, \sigma^2)$ to some centered standard normal data. We parameterize the landscape by $h = \log(\sigma^2)$ so that the Fisher information matches the standard dot product [Amari, 1998]. More explicitly, the GAUSS landscape is a probability distribution \mathcal{D} over functions $l_x : \mathbb{R}^1 \rightarrow \mathbb{R}$ on 1-dimensional weight space, indexed by standard-normally distributed 1-dimensional datapoints x and defined by the expression:

$$l_x(h) \triangleq \frac{1}{2} (h + x^2 \exp(-h))$$

812 The gradient at sample x and weight σ is then $g_x(h) = (1 - x^2 \exp(-h))/2$. Since $x \sim \mathcal{N}(0, 1)$, the
813 gradient $g_x(h)$ will be affinely related to a chi-squared, and in particular non-Gaussian.

814 To measure overfitting, we initialize at the true test minimum $h = 0$, then train and see how much the
815 test loss increases. At $h = 0$, the expected gradient vanishes, and the test loss of SGD involves only
816 diagrams that have no leaves of size one.

817 C.1.2 ARCHIMEDES

The ARCHIMEDES landscape has chirality, much like its namesake’s screw Vitruvius [circa $10^{1/2}$ b.c.e.]. Specifically, the ARCHIMEDES landscape has weights $\theta = (u, v, z) \in \mathbb{R}^3$, data points $x \sim \mathcal{N}(0, 1)$, and loss:

$$l_x(\theta) \triangleq \frac{1}{2} H(\theta) + x \cdot S(\theta)$$

Here,

$$H(\theta) = u^2 + v^2 + (\cos(z)u + \sin(z)v)^2$$

is quadratic in u, v , and

$$S(\theta) = \cos(z - \pi/4)u + \sin(z - \pi/4)v$$

818 is linear in u, v . Also, since $x \sim \mathcal{N}(0, 1)$, the $x \cdot S(\theta)$ term has expectation 0. In fact, the landscape has
819 a three-dimensional continuous screw symmetry consisting of translation along z and simultaneous
820 rotation in the $u - v$ plane. Our experiments are initialized at $u = v = z = 0$, which lies within a valley
821 of global minima defined by $u = v = 0$.


822 The paper body showed that SGD travels in ARCHIMEDES’ $+z$ direction. By topologically quotienting
823 the weight space, say by identifying points related by a translation by $\Delta z = 200\pi$, we may turn the
824 line-shaped valley into a circle-shaped valley. Then SGD eternally travels, say, counterclockwise.

825 C.1.3 MEAN ESTIMATION

The MEAN ESTIMATION family of landscapes has 1 dimensional weights θ and 1-dimensional datapoints x . It is defined by the expression:

$$l_x(\theta) \triangleq \frac{1}{2} H\theta^2 + xS\theta$$

826 Here, H, S are positive reals parameterizing the family; they give the hessian and (square root of)
827 gradient covariance, respectively.

For our hyperparameter-selection experiment (Figure 11 ) we introduce an l_2 regularization term as follows:

$$l_x(\theta, \lambda) \triangleq \frac{1}{2} (H + \lambda)\theta^2 + xS\theta$$

828 Here, we constrain $\lambda \geq 0$ during optimization using projections; we found similar results when
829 parameterizing $\lambda = \exp(h)$, which obviates the need for projection but necessitates a non-canonical
830 choice of initialization. We initialize $\lambda = 0$.

831 C.2 What image-classification landscapes did we use?

832 C.2.1 Architectures

833 In addition to the artificial loss landscapes GAUSS, ARCHIMEDES, and MEAN ESTIMATION, we tested our
834 predictions on logistic linear regression and simple convolutional networks (2 convolutional weight
835 layers each with kernel 5, stride 2, and 10 channels, followed by two dense weight layers with hidden
836 dimension 10) for the CIFAR-10 Krizhevsky [2009] and Fashion-MNIST datasets Xiao et al. [2017].
837 The convolutional architectures used tanh activations and Gaussian Xavier initialization. To set a
838 standard distance scale on weight space, we parameterized the model so that the Gaussian-Xavier
839 initialization of the linear maps in each layer differentially pulls back to standard normal initializations
840 of the parameters.

841 C.2.2 Datasets

842 For image classification landscapes, we regard the finite amount of available data as the true (sum of
843 diracs) distribution \mathcal{D} from which we sample test and training sets in i.i.d. manner (and hence “with
844 replacement”). We do this to gain practical access to a ground truth against which we may compare
845 our predictions. One might object that this sampling procedure would cause test and training sets to
846 overlap, hence biasing test loss measurements. In fact, test and training sets overlap only in reference,
847 not in sense: the situation is analogous to a text prediction task in which two training points culled
848 from different corpora happen to record the same sequence of words, say, “Thank you!”. In any
849 case, all of our experiments focus on the limited-data regime, e.g. 10^1 datapoints out of $\sim 10^{4.5}$ dirac
850 masses, so overlaps are rare.

851 C.3 Measurement process

852 C.3.1 Diagram evaluation on real landscapes

853 We implemented the formulae of §?? in order to estimate diagram values from real data measured at
854 initialization from batch averages of products of derivatives.

855 C.3.2 Descent simulations

856 We recorded test and train losses for each of the trials below. To improve our estimation of average
857 differences, when we compared two optimizers, we gave them the same random seed (and hence the
858 same training sets).

859 We ran $2 \cdot 10^5$ trials of GAUSS with SDE and SGD, initialized at the test minimum with $T = 1$ and η
860 ranging from $5 \cdot 10^{-2}$ to $2.5 \cdot 10^{-1}$. We ran $5 \cdot 10^1$ trials of ARCHIMEDES with SGD with $T = 10^4$ and η
861 ranging from 10^{-2} to 10^{-1} . We ran 10^3 trials of MEAN ESTIMATION with GD and STIC with $T = 10^2$, H
862 ranging from 10^{-4} to $4 \cdot 10^0$, a covariance of gradients of 10^2 , and the true mean 0 or 10 units away
863 from initialization.

864 We ran $5 \cdot 10^4$ trials of the CIFAR-10 convnet on each of 6 Glorot-Xavier initializations we fixed
865 once and for all through these experiments for the optimizers SGD, GD, and GDC, with $T = 10$ and
866 η between 10^{-3} and $2.5 \cdot 10^{-2}$. We did likewise for the linear logistic model on the one initialization
867 of 0.

868 We ran $4 \cdot 10^4$ trials of the Fashion-MNIST convnet on each of 6 Glorot-Xavier initializations we fixed
869 once and for all through these experiments for the optimizers SGD, GD, and GDC with $T = 10$ and η
870 between 10^{-3} and $2.5 \cdot 10^{-2}$. We did likewise for the linear logistic model on the one initialization of
871 0.

872 C.4 Implementing optimizers

873 We approximated SDE by refining time discretization by a factor of 16, scaling learning rate down
874 by a factor of 16, and introducing additional noise in the shape of the covariance in proportion as
875 prescribed by the Wiener process scaling.

Our GDC regularizer was implemented using the unbiased estimator

$$\hat{C} \triangleq (l_x - l_y)_\mu l_{xy}/2$$

876 For our tests of regularization based on Corollary 2, we exploited the low-dimensional special structure
877 of the artificial landscape in order to avoid diagonalizing to perform the matrix exponentiation:
878 precisely, we used that, even on training landscapes, the covariance of gradients would be degenerate
879 in all but one direction, and so we need only exponentiate a scalar.

880 C.5 Software frameworks and hardware

881 All code and data-wrangling scripts can be found on github.com/???????/perturb. This link will be
882 made available after the period of double-blind review.

883 Our code uses PyTorch 0.4.0 Paszke et al. [2019] on Python 3.6.7; there are no other substantive
884 dependencies. The code’s randomness is parameterized by random seeds and hence reproducible.

885 We ran experiments on a Lenovo laptop and on our institution’s clusters; we consumed about 100
886 GPU-hours.

887 C.6 Unbiased estimators of landscape statistics

888 We use the following method — familiar to some of our colleagues but hard to find writings on — for
889 obtaining unbiased estimates for various statistics of the loss landscape. The method is merely an
890 elaboration of Bessel’s factor [Gauss, 1823]. For completeness, we explain it here.

891 Given samples from a joint probability space $\prod_{0 \leq d < D} X_d$, we seek unbiased estimates of *multipoint*
892 *correlators* (i.e. products of expectations of products) such as $\langle x_0 x_1 x_2 \rangle \langle x_3 \rangle$. Here, angle brackets
893 denote expectations over the population. For example, say $D = 2$ and from $2S$ samples we’d like
894 to estimate $\langle x_0 x_1 \rangle$. Most simply, we could use $A_{0 \leq s < 2S} x_0^{(s)} x_1^{(s)}$, where A denotes averaging over the
895 sample. In fact, the following also works:

$$S \left(A_{0 \leq s < S} x_0^{(s)} \right) \left(A_{0 \leq s < S} x_1^{(s)} \right) + (1 - S) \left(A_{0 \leq s < S} x_0^{(s)} \right) \left(A_{S \leq s < 2S} x_1^{(s)} \right) \quad (6)$$

896 When multiplication is expensive (e.g. when each $x_d^{(s)}$ is a tensor and multiplication is tensor
897 contraction), we prefer the latter, since it uses $O(1)$ rather than $O(S)$ multiplications. This in turn
898 allows more efficient use of batch computations on GPUs. We now generalize this estimator to
899 higher-point correlators (and $D \cdot S$ samples).

For uniform notation, we assume without loss that each of the D factors appears exactly once in
the multipoint expression of interest; such expressions then correspond to partitions on D elements,
which we represent as maps $\mu : [D] \rightarrow [D]$ with $\mu(d) \leq d$ and $\mu \circ \mu = \mu$. Note that $|\mu| := \text{im}(\mu)$
counts μ ’s parts. We then define the statistic

$$\{x\}_\mu \triangleq \prod_{0 \leq d < D} A_{0 \leq s < S} x_d^{(\mu(d) \cdot S + s)}$$

and the correlator $\langle x \rangle_\mu$ we define to be the expectation of $\{x\}_\mu$ when $S = 1$. In this notation, 6 says:

$$\langle x \rangle_{\boxed{0} \boxed{1}} = \mathbb{E} \left[S \cdot \{x\}_{\boxed{0} \boxed{1}} + (1 - S) \cdot \{x\}_{\boxed{0} \boxed{1}} \right]$$

900 Here, the boxes indicate partitions of $[D] = [2] = \{0, 1\}$. Now, for general μ , we have:

$$\mathbb{E} \left[S^D \{x\}_\mu \right] = \sum_{\tau \leq \mu} \left(\prod_{0 \leq d < D} \frac{S!}{(S - |\tau(\mu^{-1}(d))|)!} \right) \langle x \rangle_\tau \quad (7)$$

where ‘ $\tau \leq \mu$ ’ ranges through partitions *finer* than μ , i.e. maps τ through which μ factors. In smaller steps, 7 holds because

$$\begin{aligned}\mathbb{E}[S^D \{x\}_\mu] &= \mathbb{E}\left[\sum_{(0 \leq s_d < S) \in [S]^D} \prod_{0 \leq d < D} x_d^{(\mu(d) \cdot S + s_d)}\right] \\ &= \sum_{\substack{(0 \leq s_d < S) \\ \in [S]^D}} \mathbb{E}\left[\prod_{0 \leq d < D} x_d^{(\min\{\tilde{d} : \mu(\tilde{d}) \cdot S + s_{\tilde{d}} = \mu(d) \cdot S + s_d\})}\right] \\ &= \sum_{\tau} \left| \left\{ \begin{array}{l} (0 \leq s_d < S) \in [S]^D : \\ (\mu(d) = \mu(\tilde{d})) \wedge s_d = s_{\tilde{d}} \end{array} \right\} \right| \langle x \rangle_{\tau} \\ &= \sum_{\tau \leq \mu} \left(\prod_{0 \leq d < D} \frac{S!}{(S - |\tau(\mu^{-1}(d))|)!} \right) \langle x \rangle_{\tau}\end{aligned}$$

Solving 7 for $\langle x \rangle_{\mu}$, we find:

$$\langle x \rangle_{\mu} = \frac{S^D}{S^{|\mu|}} \mathbb{E}[\{x\}_{\mu}] - \sum_{\tau < \mu} \left(\prod_{d \in \text{im}(\mu)} \frac{(S-1)!}{(S - |\tau(\mu^{-1}(d))|)!} \right) \langle x \rangle_{\tau}$$

This expresses $\langle x \rangle_{\mu}$ in terms of the batch-friendly estimator $\{x\}_{\mu}$ as well as correlators $\langle x \rangle_{\tau}$ for τ *strictly* finer than μ . We may thus (use dynamic programming to) obtain unbiased estimators $\langle x \rangle_{\mu}$ for all partitions μ . Symmetries of the joint distribution and of the multilinear multiplication may further streamline estimation by turning a sum over τ into a multiplication by a combinatorial factor. For example, in the case of complete symmetry:

$$\langle x \rangle_{\overline{012}} = S^2 \{x\}_{\overline{012}} - \frac{(S-1)!}{(S-3)!} \{x\}_{\overline{0} \overline{1} \overline{2}} - 3 \frac{(S-1)!}{(S-2)!} \{x\}_{\overline{0} \overline{12}}$$

C.7 Additional figures

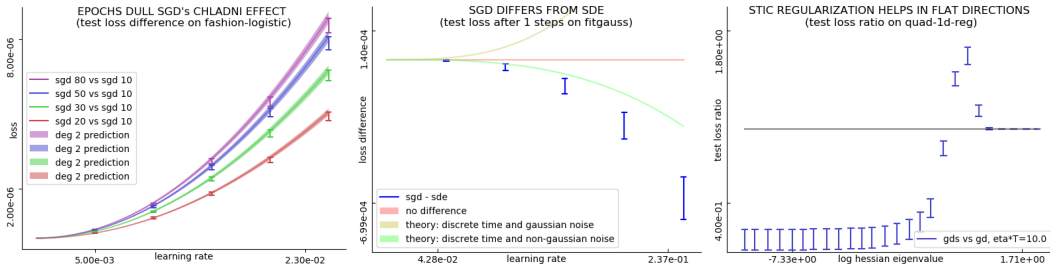


Figure 11: **Further experimental results.** **Left:** SGD with 2, 3, 5, 8 epochs incurs greater test loss than one-epoch SGD (difference shown in I bars) by the predicted amounts (predictions shaded) for a range of learning rates. Here, all SGD runs have $N = 10$; we scale the learning rate for E -epoch SGD by $1/E$ to isolate the effect of inter-epoch correlations away from the effect of larger ηT . **Center:** SGD’s difference from SDE after $\eta T \approx 10^{-1}$ with maximal coarseness on Gauss. Two effects not modeled by SDE — time-discretization and non-Gaussian noise oppose on this landscape but do not completely cancel. Our theory approximates the above curve with a correct sign and order of magnitude; we expect that the fourth order corrections would improve it further. **Right:** Blue intervals regularization using Corollary 2. When the blue intervals fall below the black bar, this proposed method outperforms plain GD. For MEAN ESTIMATION with fixed C and a range of H s, initialized a fixed distance *away* from the true minimum, descent on an l_2 penalty coefficient λ improves on plain GD for most Hessians. The new method does not always outperform GD, because λ is not perfectly tuned according to STIC but instead descended on for finite ηT .