A Perturbative Analysis of Stochastic Descent

Samuel C. Tenka
Computer Science and AI Lab
Massachusetts Institute of Technology
Cambridge, MA 02139
coli@mit.edu

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 §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.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

First, 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:

$$G \triangleq \mathbb{E}_x\left[\nabla l_x(\theta)\right] \triangleq$$

$$H \triangleq \mathbb{E}_x\left[\nabla\nabla l_x(\theta)\right] \triangleq \qquad C \triangleq \mathbb{E}_x\left[(\nabla l_x(\theta) - G)^2\right] \triangleq$$

$$J \triangleq \mathbb{E}_x\left[\nabla\nabla\nabla l_x(\theta)\right] \triangleq \qquad S \triangleq \mathbb{E}_x\left[(\nabla l_x(\theta) - G)^3\right] \triangleq$$

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 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 $C =$ with $H =$ to get . As another example, we may join two copies of $G =$ with two copies of $H =$ to get . Intuitively, each diagram represents the interaction of its components: of gradients (G), noise ($C, S, \cdots$) and curvature ($H, J, \cdots$). In fact, §A.6 physically interprets edges as carrying information between updates and toward the test measurement.

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 $S =$. In this case, there is one fewest-edged diagram — ; it results from joining S with $J =$. 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.

Remark 1. To understand Example 1's result, we specialize to isotropic curvature ($\eta H = \|\eta H\|_2 I$) and take $T \to \infty$, 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 $\lim_{\eta\to 0} q/p(\eta) = 0$ for some homogeneous degree-d polynomial p.

We fix a loss function $l : \mathcal{M} \to \mathbb{R}$ 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 $(l_n : 0 \le n < N)$ 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 tth 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 are also called column/row vectors.

1.3 Related work

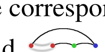
It was Kiefer and Wolfowitz [1952] who, in uniting gradient descent [Cauchy, 1847] with stochastic approximation [Robbins and Monro, 1951], invented SGD. Since the development of back-propagation for efficient differentiation [Werbos, 1974], SGD has been used to train connectionist models, e.g. neural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

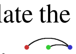
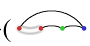
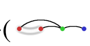
Several lines of work treat the overfitting of SGD-trained networks [Neyshabur et al., 2017a]. For example, Bartlett et al. [2017] controls the Rademacher complexity of deep hypothesis classes, leading to optimizer-agnostic generalization bounds. Yet SGD-trained networks generalize despite their ability 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 $0 \le t \le T$ 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 $\left|\mathrm{Aut}_f(D)\right|$ count automorphisms of D that preserve f. Up to unbiasing terms,[2] we construct the *re-summed value* $\mathrm{rvalue}_f(D)$ as follows:

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

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):

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

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.

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*

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

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 $\theta_\star$ is a local minimum of l and $H(\theta_\star)$ is strictly positive, then for SGD initialized sufficiently close to $\theta_\star$, the dth-order truncation of Theorem 1 converges as $T \to \infty$.*

Caution: the $T \to \infty$ 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, $D =$ 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 $\theta_\star$. The overfitting (test loss minus $l(\theta_\star)$) and gen. gap (test minus train loss) due to training are:*

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

The gen. gap tends to $C_{\mu\nu}(H^{-1})^{\mu\nu}/N$ as $T \to \infty$. 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.

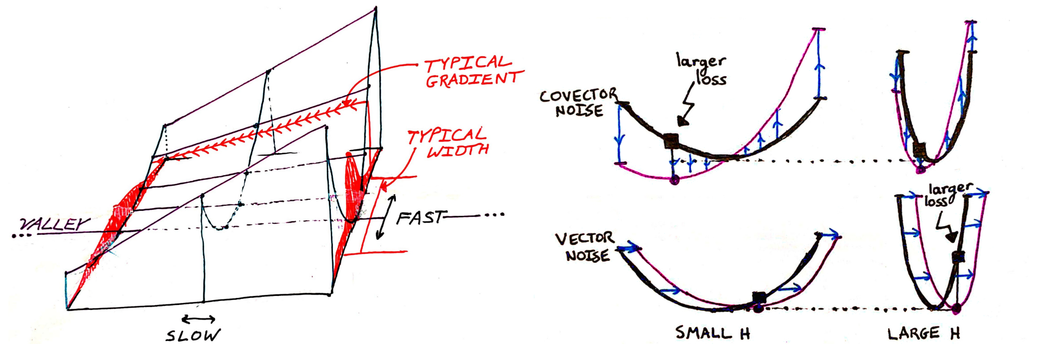


Figure 1: **Geometric intuition for curvature-noise interactions. Left**: Gradient noise pushes SGD toward minima that are flat *with respect to the covariance* (Corollary 1). The red densities show the typical θs, perturbed from the minimum due to noise C, in two cross sections of the loss valley. $J = \nabla H$ 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]. **Right**: Both curvature and the structure of noise affect overfitting. In each of the four subplots, the $\leftrightarrow$ axis represents weight space and the $\updownarrow$ axis represents loss. ☐▦: *covector*-perturbed landscapes favor large Hs. ☐▦: *vector*-perturbed landscapes favor small Hs. 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 $-\nabla C$, the effect is strongest when gradient noise is not averaged out by large batch sizes.

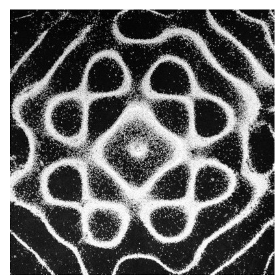


Figure 2: **Chladni plate.** Grains of sand on a vibrating plate tend toward stationary regions.

Corollary 3 ( ). *SGD avoids high-C regions more than GD: $l_C \triangleq \frac{N-1}{4N}\nabla^\mu C^\nu_\nu = \mathbb{E}\left[\theta_{GD} - \theta_{SGD}\right]^\mu - o(\eta^2)$. 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: $\mathbb{E}\left[\theta_{M=M_0} - \theta_{M=1}\right]^\mu = \left(\frac{M_0-1}{M_0}\right) N \left(\nabla^\mu C^\nu_\nu\right) + o(\eta^2)$.*

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]

[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, [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 ,). §C.7 supports Corollary 4's predictions about epoch number. Figure 3 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).

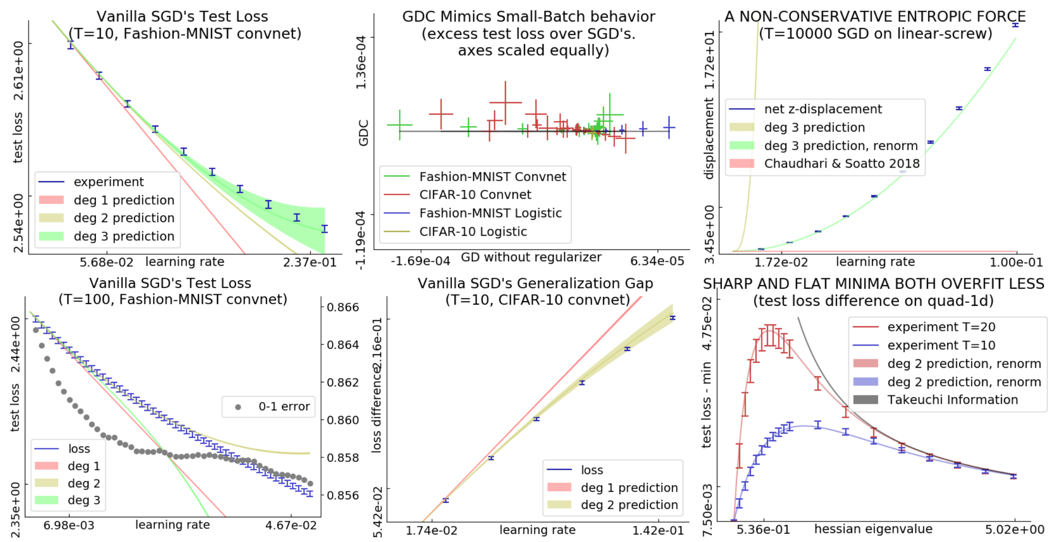


Figure 3: **Left: Perturbation models SGD for small ηT.** Fashion-MNIST convnet's test loss vs learning rate. In this small T setting, we choose to use our theory's simpler un-resummed values (A.4) instead of the more precise rvalues. : For all init.s tested (1 shown, 11 unshown), the order 3 prediction agrees with experiment through $\eta T \approx 10^0$, corresponding to a decrease in 0-1 error of $\approx 10^{-3}$. : 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, shows that GDC matches SGD (small vertical varianec) 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, $T = 10$. : 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. : 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. : For Mean Estimation with fixed C and a range of Hs, 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 Takeuchi information [Dixon and Ward, 2018] and thus on AIC.
Throughout:the *re-normed* predictions refer to those of Theorem 1, approximated as in Remark 2. Predictions not labeled as *re-normed* are more drastic (namely, polynomial) approximations of Theorem 1's result. We the authors should have written "re-summed" instead of "re-normed", and "single-epoch, singleton-batch" instead of "vanilla".

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 cyclic motion. 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, even for large T (Figure 3). Because SGD's motion depends smoothly on the landscape, the special case of ARCHIMEDES implies that non-conservativity is typical. We may have sought an "effective loss" so that, up to $\sqrt{T}$ diffusion terms, SGD on the old loss is like ODE on the new loss. The non-conservativity of SGD's velocity shows that there is no such "effective loss".

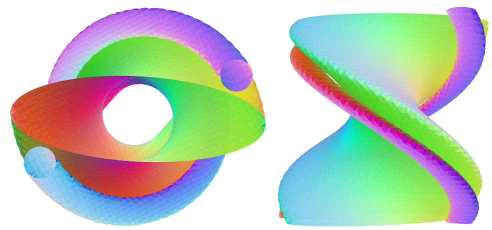


Figure 4: **Two views of ARCHIMEDES.** Green: a level surface of l twisting around a valley of minima (z axis) at its center; l is large outside this surface. Purple tubes: θs typical due to non-isotropic noise. The typical locations of θ are pulled toward lower loss (steepest descent toward the level 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 $(\eta C)^\mu_\nu$ in tangent space.

Broader impacts

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

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

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

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

delight,

surprise, and

beauty.

Acknowledgements

We would like to thank Sho Yaida, Dan A. Roberts, and Josh Tenenbaum for their patient guidance. Through their incisive questions, Dan A. Roberts, Ben R. Bray, Sasha Rakhlin, Sho Yaida, and Wenli Zhao led us to the questions we answer here. We appreciate the time and energy of Andy Banburski, Ben R. Bray, Jeff Lagarias, and Wenli Zhao spent in critiqueing our drafts. Without the encouragement of Jason Corso, Chloe Kleinfeldt, Alex Lew, Ari Morcos, and David Schwab, this paper would not be. Finally, we thank our anonymous reviewers for inspiring an improved presentation. This work was funded in part by MIT's Jacobs Presidential Fellowship.

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. Stochastic gradient descent 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 systémes d'équations simultanées. *Comptes rendus de l'Académie des Sciences*, 1847.

P. Chaudhari and S. Soatto. Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. *ICLR*, 2018.

E.F.F. Chladni. Entdeckungen über die theorie des klanges. *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's information criteria 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 Crypography*, 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.

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.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Sho Yaida. Fluctuation-dissipation relations for stochastic gradient descent. *ICLR*, 2019a.

Sho Yaida. A first law of thermodynamics for stochastic gradient descent. *Personal Communication*, 2019b.

Chiyuan Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *ICLR*, 2017.

Hongyi Zhang, S.J. Reddi, and S. Sra. Fast stochastic optimization on riemannian manifolds. *NeurIPS*, 2016.

Organization of the appendices

The following three appendices serve three respective functions:

- to explain how to calculate using diagrams;
- to precisely state and prove our results, then pose a conjecture;
- to specify our experimental methods and results.

In more detail, we organize the appendices as follows.

A How to calculate test losses: a practical guide

Our work introduces a novel technique for calculating the expected learning curves of SGD in terms of statistics of the loss landscape near initialization. Here, we explain this technique. Note that a new combinatorial object object — *spacetime* — arises as we relax the paper body's assumption that $E = B = 1$. This, too, we will explain. We note for now that there are are **four steps** to computing the expected test loss, or other quantities of interest, after a specific number of gradient updates:

- **Draw the spacetime grid** that encodes our chosen SGD hyperparameters (namely, batch size, training set size, and number of epochs).
- **Draw embeddings**, of diagrams into the spacetime, as needed for our desired precision.
- **Evaluate each diagram embedding**, whether exactly (via rvalues) or roughly (via uvalues).
- **Sum the embeddings' values** to obtain the quantity of interest as a function of η.

After presenting a small, complete example calculation that follows these four steps, we explain each of the respective steps in its own sub-section. We then discuss how diagrams often offer intuition as well as calculational help. Though we focus on the computation of expected test losses, we describe how a small change in the above four steps allows for the computation also of variances (instead of expectations), of train losses (instead of test losses), or of weight displacements (instead of losses). We conclude by noting that our mathematical theory may be phrased without reference to diagrams; we compare such direct calculation to the diagram method, pointing out when and why diagrams streamline computation.

A.1 An example calculation

A.2 How to identify the relevant space-time

A.3 How to identify the relevant diagram embeddings

A.4 How to evaluate each embedding

Make sure to also discuss un-re-summed values!!!

A.5 How to sum the embeddings' values

A.6 Interpreting diagrams to build intuition

Make sure to also discuss Chladni effect Chladni [1787]!!!

A.7 How to solve variant problems

A.8 Do diagrams streamline computation?

B Mathematics of the theory

B.1 Assumptions and definitions

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:*

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

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_tth 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

$$2\eta T b < \rho\tilde{\eta} \tag{2}$$

as norms, SGD after T steps on any train set will necessarily stay within the ρ-ball. [1] 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 $\rho(1-1/(2T))$. 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 ∇_ν 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 coeffients 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:

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

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

[1] In fact, the factor of 2 helps ensure that SGD initialized at any point within a $\rho/2$ ball will necessarily stay within the ρ-ball.

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

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

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. □

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 uvalue(D) instead of 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*

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

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

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

$$(-\eta)^d \sum_{\substack{(d_t : 0 \leq t < T) \\ \sum_t d_t = d}} \left(\prod_{0 \leq t < T} \left. \frac{(g\nabla)^{d_t}}{d_t!} \right|_{g = \nabla l_t(\theta)} \right) l(\theta) =$$

$$\sum_{\substack{D \in \mathrm{im}(\mathcal{F}) \\ \text{with } d \text{ edges}}} \left(\sum_{f : D \to \mathcal{F}(S)} \frac{1}{\left|\mathrm{Aut}_f(D)\right|} \right) \frac{\mathrm{uvalue}_\pi(D, f)}{B^d} \tag{4}$$

Here, uvalue$_\pi$ is the value of a diagram embedding before taking expectations over train sets. We have for all f that $\mathbb{E}\left[\mathrm{uvalue}_\pi(D, f)\right] = \mathrm{uvalue}(D)$. 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:

$$\nabla^{|M|} \prod_{k \in K} p_k = \sum_{\nu : M \to K} \prod_{k \in K} \left(\nabla^{|\nu^{-1}(k)|} p_k \right)$$

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

We proceed by joint induction on d and S. The base cases wherein S is empty or $d = 0$ both follow immediately from the Key Lemma, for then the only embedding is the unique embedding of the one-node diagram $\bullet$. For the induction step, suppose S is a sequence of $\mathcal{M} = \min S \subseteq S$ followed by 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 terms on the left hand side of desideratum 4. Applying the induction hypothesis with $\tilde{d} = d - d_0$, we find that that left hand side is:

$$\sum_{0 \leq d_0 \leq d} \sum_{\substack{\tilde{D} \in \mathrm{im}(\mathcal{F}) \\ \text{with } d - d_0 \text{ edges}}} \frac{1}{d_0!} \sum_{\tilde{f}: \tilde{D} \to \mathcal{F}(\tilde{S})} \left(\frac{1}{\left|\mathrm{Aut}_{\tilde{f}}(\tilde{D})\right|} \right) \cdot$$

$$(-\eta)^{d_0} \, (g\nabla)^{d_0}\Big|_{g = \nabla l_0(\theta)} \frac{\mathrm{uvalue}_\pi(\tilde{D}, \tilde{f})}{B^{d-d_0}}$$

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

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

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

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

by the Orbit Stabilizer Theorem of elementary group theory.

It is thus enough to show that

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

We will show this by a direct bijection. First, observe that $f = \beta \sqcup \tilde{f} : [d_0] \sqcup \tilde{D} \to \mathcal{M} \sqcup \tilde{S}$. So each automorphism $\phi : D \to D$ that commutes with f induces both an automorphism $\mathcal{A} = \phi|_{\tilde{D}} : \tilde{D} \to \tilde{D}$ that commutes with $\tilde{f}$ together with the data of a map $\mathcal{B} = \phi_{[d_0]} : [d_0] \to [d_0]$ that both commutes with β. However, not every such pair of maps arises from a ϕ. For, in order for $\mathcal{A} \sqcup \mathcal{B} : D \to D$ 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 $\phi \in \mathrm{Aut}_f(D)$. There are $\left|\mathrm{Aut}_{\tilde{f}}(\tilde{D})\right|$ 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 $\prod_{m \in \mathcal{M}} \left|(\beta \times \nu)^{-1}(m, i)\right|!$ 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. □

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:*

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

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

$$\sum_{f:\rho\to S} \sum_{\substack{\tilde{f}:\sigma\to S \\ \exists i_\star : f=\tilde{f}\circ i_\star}} \frac{1}{\left|\text{Aut}_{\tilde{f}}(\sigma)\right|} = \sum_{f:\rho\to S} \sum_{\substack{\tilde{f}:\sigma\to S \\ \exists i_\star : f=\tilde{f}\circ i_\star}} \sum_{\substack{i:\rho\to\sigma \\ f=\tilde{f}\circ i}} \frac{1}{\left|\text{Aut}_f(\rho)\right|} \tag{5}$$

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 is 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 tgat 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

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

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 $\left|\text{Aut}_{\tilde{f}}(\rho)\right|$. 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 stabilizer$(i) \hookleftarrow \text{Aut}_{\tilde{f}}(\rho)$ seen to be well-defined and injective because structure set morphisms are by definition strictly increasing and because is 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 $\psi \in \mathrm{Aut}_f(\rho)$ such that $i_B \sim i_A \circ \psi$ by $\phi =$ 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. □

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 ms, 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. □

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 colorred (amputated as in the previous subsubsection). 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:

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

Standard calculus then gives the desired result. □

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:

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

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

B.7.3 Corollary 2's second part

We apply the generalization gap modification (described in §**??**) to Theorem 1's result about test 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,

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

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

B.7.4 Corollaries 4 and 3

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

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

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

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

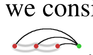
diagram	embed.s w/ $\lvert\mathrm{Aut}_f\rvert = 1$	embed.s w/ $\lvert\mathrm{Aut}_f\rvert = 2$
	1	0
	MNB	0
	$\binom{MNB}{2}$	0
	$N\binom{MB}{2}$	0
	$\binom{MNB}{2}$	0
	$N\binom{MB}{2}$	MNB

□

Proof of Corollary 3's second part. FILL IN □

B.7.5 Corollary 5

The corollary's first part follows immediately from Remark **??** in the case that $d = 2$, $P = 2$, and $(\eta N)^d$ 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

$$\frac{T\eta^3}{6}\mathbb{E}\left[\nabla^3 l\right]\mathbb{E}\left[\nabla l_{n_{t_a}} \nabla l_{n_{t_b}} \nabla l_{n_{t_c}}\right]$$

□

B.8 Future topics

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

Question 1. *Does some least-action principle govern SGD; if not, what is an essential obstacle to this characterization?*

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

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

Our work assumes a flat metric $\eta^{\mu\nu}$, but it might generalize to weight spaces curved in the sense of Riemann. [1] Such curvature finds concrete application in the *learning on manifolds* paradigm of Absil et al. [2007], Zhang et al. [2016], notably specialized to Amari [1998]'s *natural gradient descent* and Nickel and Kiela [2017]'s *hyperbolic embeddings*. While that work focuses on *optimization* on curved weight spaces, in machine learning we also wish to analyze *generalization*. Starting with the intuition that "smaller" hypothesis classes generalize better and that curvature controls the volume of small neighborhoods, we conjecture that sectional curvature regularizes learning:

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

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

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

C Experimental methods

C.1 What artificial landscapes did we use?

We define three artificial landscapes, called Gauss, Archimedes, and Mean Estimation.

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 $l_x : \mathbb{R}^1 \to \mathbb{R}$ on 1-dimensional weight space, indexed by standard-normally distributed 1-dimensional datapoints x and defined by the expression:

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

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 gradient $g_x(h)$ will be affinely related to a chi-squared, and in particular non-Gaussian.

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

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

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

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 a three-dimensional continuous screw symmetry consisting of translation along z and simulateous rotation in the $u - v$ plane. Our experiments are initialized at $u = v = z = 0$, which lies within a valley of global minima defined by $u = v = 0$.

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

Mean Estimation

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

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

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

For our hyperparameter-selection experiment (Figure 5 ▫▫▪) we introduce an l_2 regularization term as follows:

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

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

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

Architectures

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

Datasets

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

C.3 Measurement process

Diagram evaluation on real landscapes

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

Descent simulations

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

We ran $2 \cdot 10^5$ trials of Gauss with SDE and SGD, initialized at the test minimum with $T = 1$ and η ranging from $5 \cdot 10^{-2}$ to $2.5 \cdot 10^{-1}$. We ran $5 \cdot 10^1$ trials of Archimedeswith SGD with $T = 10^4$ and η ranging from 10^{-2} to 10^{-1}. We ran 10^3 trials of Mean Estimationwith GD and STIC with $T = 10^2$, H 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 from initialization.

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

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

C.4 Implementing optimizers

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

Our GDC regularizer was implemented using the unbiased estimator

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

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

C.5 Software frameworks and hardware

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

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

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

C.6 Unbiased estimators of landscape statistics

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

Given samples from a joint probability space $\prod_{0\leq d<D} X_d$, we seek unbiased estimates of *multipoint 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 denote expectations over the population. For example, say $D = 2$ and from $2S$ samples we'd like to estimate $\langle x_0 x_1\rangle$. Most simply, we could use $\mathsf{A}_{0\leq s<2S}\, x_0^{(s)} x_1^{(s)}$, where A denotes averaging over the sample. In fact, the following also works:

$$S\left(\mathop{\mathsf{A}}_{0\leq s<S} x_0^{(s)}\right)\left(\mathop{\mathsf{A}}_{0\leq s<S} x_1^{(s)}\right) + (1-S)\left(\mathop{\mathsf{A}}_{0\leq s<S} x_0^{(s)}\right)\left(\mathop{\mathsf{A}}_{S\leq s<2S} x_1^{(s)}\right) \tag{6}$$

When multiplication is expensive (e.g. when each $x_d^{(s)}$ is a tensor and multiplication is tensor contraction), we prefer the latter, since it uses $O(1)$ rather than $O(S)$ multiplications. This in turn allows more efficient use of batch computations on GPUs. We now generalize this estimator to 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 $\mu : [D] \to [D]$ with $\mu(d) \leq d$ and $\mu \circ \mu = \mu$. Note that $|\mu| := |im(\mu)|$ counts μ's parts. We then define the statistic

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

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

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

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

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

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

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

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

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

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:

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

C.7 Additional figures

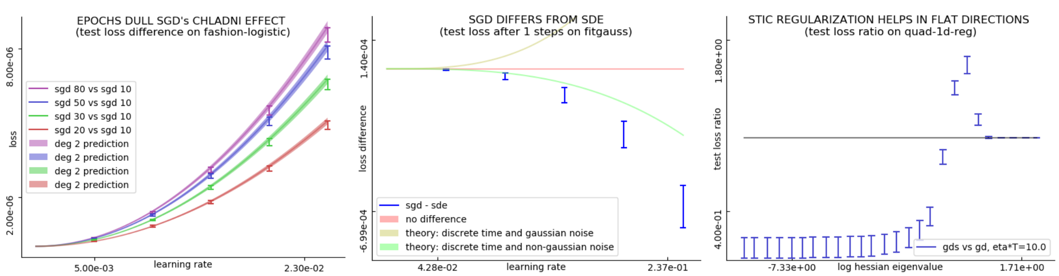


Figure 5: **Further experimental results**. **Left**: SGD with 2, 3, 5, 8 epochs incurs greater test loss than one-epoch SGD (difference shown in I bars) by the predicted amounts (predictions shaded) for a range of learning rates. Here, all SGD runs have $N = 10$; we scale the learning rate for E-epoch 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 Hs, 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.