A Space-Time Approach to Analyzing Stochastic Gradient Descent

Anonymous Authors[1]

Abstract

We present a diagrammatic calculus for reasoning about the behavior, at small learning rates, of SGD and its variants. We interpret the diagrams as histories of scattering events, thus offering a new physical analogy for descent. Illustrating this technique, we construct a regularizing term that causes large-batch GD to emulate small-batch SGD, present a model-selection heuristic that depends only on statistics measured before optimization, and exhibit a counter-intuitive loss landscape wherein SGD eternally cycles counterclockwise around a circle of minima.

1. Introduction

Stochastic gradient descent (SGD) decreases an unknown objective l by performing discrete-time steepest descent on noisy estimates of l. A key question is how the noise affects the final objective value. We connect SGD dynamics to physical scattering theory, thus providing a quantitative and qualitative toolkit for answering this question.

Specifically, we derive a diagram-based formalism for reasoning about SGD via a path integral over possible interactions between weights and data. The formalism permits perturbative analysis, leading to predictions of learning curves for small η. Unlike the continuous-time limits of previous work, this framework models discrete time, and with it, the potential **non-Gaussianity** of noise. We thus obtain new results quantifying the **effect of epoch number, batch size, and momentum** on SGD test loss. We also contrast SGD against popular continuous-time approximations such as ordinary or stochastic differential equations (ODE, SDE).

Path integrals offer not only quantitative predictions but also an exciting new viewpoint — that of iterative optimization as a **scattering process**. Much as individual Feynman diagrams (see Dyson (1949a)) depict how local particle interactions compose into global outcomes, our diagrams depict how individual SGD updates influence each other before affecting a final test loss. In fact, we import from physics tools such as **crossing symmetries** (see Dyson (1949b)) and **re-normalization** (see Gell-Mann & Goldberger (1954)) to simplify our calculations and refine our estimates. The diagrams' combinatorial properties immediately yield several precise qualitative conclusions as well, for instance that to order η^2, **inter-epoch** shuffling does not affect expected test loss.

2. Background and Notation

We adopt the standard summation notation for Greek but not Roman indices, suppressing indices when convenient and clear. To expedite dimensional analysis, we follow (Bonnabel, 2013) in considering the learning rate as an inverse metric $\eta^{\mu\nu}$ that converts a gradient (row vector) into a displacement (column vector).

Then SGD with learning rate $\eta^{\mu\nu}$ decreases an objective l by updating on a sequence of smooth, identically distributed unbiased estimates $(l_n : 0 \leq n < N)$ of l:

$$\theta_{t+1} := \theta_t - \eta^{\mu\nu} \nabla_\mu l_t(\theta_t) \qquad (0 \leq t < T = N) \tag{1}$$

The l_n are typically not independent, and our framework will model this.

Gradient-Based Optimization and Generalization

Perturbative Scattering Theory

Motivating Example

Gradient descent makes a first order approximation

Intuitively, each descent step displaces θ by $-\eta\nabla l$ and hence decreases the loss $l(\theta)$ by $\eta(\nabla l)^2$; thus, we expect after T steps a net decrease of $T\eta(\nabla l)^2$:

$$l(\theta_T) \approx l(\theta_0) - T \cdot \eta \cdot (\nabla l(\theta_0))^2 \tag{2}$$

This intuition fails to capture two crucial facts: **curvature** — that as θ changes during training, so may $\nabla l(\theta)$ — and **noise** — that l_n and l may differ. We may account for curvature, i.e. $\nabla\theta$'s evolution, in analogy with (2)'s estimate of $l(\theta)$'s evolution: each step displaces θ by $-\eta(\nabla l)$ and hence changes $\nabla l(\theta)$ by $-\eta(\nabla^2 l)(\nabla l)$; thus, we expect that

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

$\nabla l(\theta_t)$ differs from $l(\theta_0)$ by $-t\eta(\nabla^2 l)(\nabla l)$. Recalling that $\sum_{0\le t<T} t = \binom{T}{2}$, we estimate the net displacement of θ as $\Delta\theta = -T\eta\nabla l + \binom{T}{2}\eta^2(\nabla^2 l)(\nabla l)$. We thus refine (2) by substituting $\Delta\theta$ into a Taylor expansion of l at θ_0 ($l, \cdots$ unadorned are evaluated at θ_0):

$$\begin{aligned}
l(\theta_T) \approx l &+ \left(-T\eta\nabla l + \binom{T}{2}\eta^2(\nabla^2 l)(\nabla l)\right)\nabla l && (3)\\
&+\frac{1}{2}\left(-T\eta\nabla l + \binom{T}{2}\eta^2(\nabla^2 l)(\nabla l)\right)^2 \nabla l^2 && (4)\\
\approx l &- T\cdot\eta\cdot(\nabla l)^2 && (5)\\
&+ T\left(T-\frac{1}{2}\right)\cdot\eta^2\cdot\nabla^2 l(\nabla l)^2 && (6)
\end{aligned}$$

So far, we have essentially recovered Theorem 2.1.14 of Nesterov (2004).

After establishing some general-purpose results, we will be able to FILL IN.

The more complicated the direct computation, the greater the savings of using diagrams.

3. Diagram Calculus for SGD

Suppose s is smooth on weight space; for example, s may be a test or train loss. We may track $s(\theta)$ as θ is updated by SGD as follows:

Proposition 1. *The formal Maclaurin series of $s(\theta_T)$ with respect to η is:*

$$\sum_{0\le d<\infty} (-\eta)^d \sum_{\substack{(d_t:0\le t<T)\\ \sum_t d_t=d}} \left(\prod_{0\le t<T} \left.\frac{(g\nabla)^{d_t}}{d_t!}\right|_{g=\nabla l_t(\theta)}\right) s(\theta_0)$$

In averaging over training sets (and hence over the sequence $(l_t : 0 \le t < T)$ considered as a random variable), we may factor the expectation of the above product according to independence relations between the l_t. We view various training procedures (e.g. GD, SGD with(out) inter-epoch shuffling) as **prescribing different independence relations** that lead to different factorizations and hence to potentially different generalization behavior at each order of η.

The book keeping

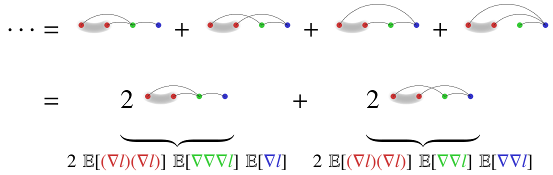
Physical Dictionary

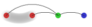
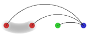
Evaluating Diagrams

An instance of the above product (for $s = l_a$ drawn from a test set and $0 \le c \le b < T$) is

$$\begin{aligned}
\eta^3(\nabla l_c\nabla)^2(\nabla l_b\nabla)l_a =&(\nabla^\lambda l_c)(\nabla^\mu l_c)(\nabla_\lambda\nabla_\mu\nabla^\nu l_b)(\nabla_\nu l_a)\\
&+(\nabla^\lambda l_c)(\nabla^\mu l_c)(\nabla_\lambda\nabla^\nu l_b)(\nabla_\mu\nabla_\nu l_a)\\
&+(\nabla^\lambda l_c)(\nabla^\mu l_c)(\nabla_\mu\nabla^\nu l_b)(\nabla_\lambda\nabla_\nu l_a)\\
&+(\nabla^\lambda l_c)(\nabla^\mu l_c)(\nabla^\nu l_b)(\nabla_\lambda\nabla_\mu\nabla_\nu l_a)
\end{aligned}$$

where we use η to raise indices. To reduce clutter, we adapt the string notation of Penrose (1971). Then, in expectation over (l_c, l_b, l_a) drawn i.i.d.:


$\cdots =$
$+$
$+$
$+$
$=$
2
$+$
2
$2\ \mathbb{E}[(\nabla l)(\nabla l)]\ \mathbb{E}[\nabla\nabla\nabla l]\ \mathbb{E}[\nabla l]$
$2\ \mathbb{E}[(\nabla l)(\nabla l)]\ \mathbb{E}[\nabla\nabla l]\ \mathbb{E}[\nabla\nabla l]$

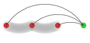
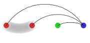
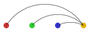
Above, each node corresponds to a loss function (here, red for l_c, green for l_b, blue for l_a), differentiated d times for a degree-d node (for instance, l_b is differentiated thrice in the first diagram and twice in the second). **Thin "edges"** mark contractions by η. **Fuzzy "ties"** denote independence relationships by connecting identical loss functions (here, l_c with l_c): nodes not connected by a path of fuzzy ties are independent. The colors are redundant with the fuzzy ties and used only so that we may concisely refer to a specific node in prose. Crucially, for a fixed, i.i.d. distribution over (l_c, l_b, l_a), **the topology of a diagram determines its expected value**. For instance, $\mathbb{E}$  $= \mathbb{E}$  because both are trees with two leaves tied. Thus follows the simplification on the second line above. As shown with braces, we may convert back to explicit tensor expressions, invoking independence between untied nodes to factor the expression. However, as we will see, the diagrammatic form of a tensor expression offers us physical intuition and guides us toward constructing unbiased estimators of the statistics they represent.

The recipes for writing down test (or train) losses of SGD (or GD and other variants) are straight-forward in the diagram notation because they reduce the problem of evaluating the previous dynamical expressions to the problem of counting isomorphic graphs. An appendix provides details and proofs for a variety of instances. For now, we illustrate how to compute the test loss of vanilla SGD.

Recipe for the Test Loss of Vanilla SGD

The diagrams relevant to order $(-\eta)^d$ of this Taylor expansion are trees with d thin edges, none of which connect fuzzily-tied nodes, and such that the rightmost node is not fuzzily tied. We regard a diagram as a partial order (by reading the thin edges as Hasse diagram) equipped with a partition of nodes (induced by the fuzzy ties). If a diagram's nodes partition into P sets of size $(d_p : 0 \leq p < P)$, then the Taylor coefficient on that diagram's isomorphism class is

$$\frac{(-\eta)^d}{d!}\binom{T}{P}\binom{d}{d_0, \cdots, d_{P-1}} \cdot K_{D\to[d+1]} \in \Theta\left((\eta T)^d T^{P-d-1}\right)$$

where $K_{D\to[d+1]}$ counts the total orders that extend the partial order of D and in which each of the P parts appears as a contiguous segment. For example, at order $(-\eta)^3\binom{T}{3}$ there are two isomorphism classes respectively with 4 and 2 total orderings, respectively; at order $(-\eta)^3\binom{T}{2}$ there are three classes respectively with 2, 2, and 3 orderings; at order $(-\eta)^3\binom{T}{1}$ there is only one class and it has 1 ordering (see Table 1). Intuitively, we regard ηT as measuring the total **physical time** of the optimization process, and $1/T$ as measuring the **coarseness** of time discretization. More precisely, we have a double-series in $(\eta T)^d T^{P-d-1}$, where d counts thin edges and $d + 1 - P$ counts fuzzy ties; the $P = d + 1$ terms give us an ODE (hence noiseless) approximation to SGD, while $P \leq d$ terms give us the effect of time-discretization and hence noise on SGD test loss. For Gaussian gradients,  $= 3$  $- 2$  in expectation, but above, we make no such assumption.

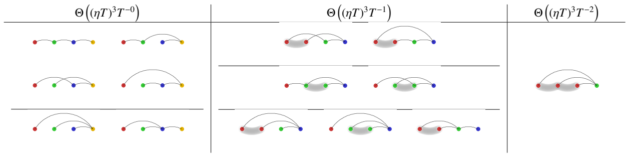

$\Theta\left((\eta T)^3 T^{-0}\right)$
$\Theta\left((\eta T)^3 T^{-1}\right)$
$\Theta\left((\eta T)^3 T^{-2}\right)$

Table 1. Degree-3 scattering diagrams for test loss of vanilla SGD. **Left:** $(d, P) = (3, 3)$. Diagrams for ODE behavior. **Center:** $(d, P) = (3, 2)$. 1st order deviation of SGD away from ODE. **Right:** $(d, P) = (3, 1)$. 2nd order deviation of SGD from ODE with appearance of non-Gaussian statistics.

Crossing Symmetries and Renormalization

Descent as Scattering: Experimental

4. Predictions and Applications

Emulating Small Batches with Large Ones

Analyzing Second Order Methods

Epochs and Overfitting

Myopic Model Selection

Comparison to Continuous Time

Thermodynamic Engine

We clarify

5. Related Work

It was Kiefer & Wolfowitz (1952) who, in uniting gradient descent (Cauchy, 1847) with stochastic approximation (Robbins & Monro, 1951), invented SGD. Since the development of back-propagation for efficient differentiation (Werbos, 1974), SGD and its variants have been used to train connectionist models including neural networks (Bottou, 1991), in recent years to remarkable success (LeCun et al., 2015).

Several lines of work predict the overfitting of SGD-trained networks (Neyshabur et al., 2017a). For instance, Bartlett et al. (2017) controls the Rademacher complexity of deep hypothesis classes, leading to generalization bounds that are post hoc or optimizer-agnostic. However, since deep networks trained via SGD generalize despite their seeming ability to shatter large sets (Zhang et al., 2017), one infers that generalization arises from the aptness to data of not only architecture but also optimization (Neyshabur et al., 2017b). Others have focused on the implicit regularization of SGD itself, for instance by modeling descent via stochastic differential equations (e.g. Chaudhari & Soatto (2018)). However, as explained by Yaida (2019), such continuous-time analyses cannot treat covariance correctly, and so they err when interpreting results about SDEs as results about SGD.

Following Roberts (2018), we avoid making a continuous-time approximation by instead Taylor-expanding around the learning rate $\eta = 0$. In fact, we develop a diagrammatic language for evaluating each Taylor term that is similar to and inspired by the field theory methods popularized by Dyson (1949a). Using this technique, we quantify the overfitting effects of batch size and epoch number, and based on this analysis, propose a regularizing term that causes large-batch GD to emulate small-batch SGD, thus establishing a precise version of the Covariance-BatchSize-Generalization relationship conjectured in Jastrzębski et al.

(2018).

While we make rigorous, architecture-agnostic predictions of learning curves, these predictions become vacuous for large η. Other discrete-time dynamical analyses allow large η by treating neural generalization phenomenologically, whether by fitting to an empirically-determined correlate of Rademacher bounds (Liao et al., 2018), by exhibiting generalization of local minima **flat** with respect to the standard metric (see Hoffer et al. (2017), Keskar et al. (2017), citetwa18), or by exhibiting generalization of local minima **sharp** with respect to the standard metric (see Stein (1956), Dinh et al. (2017), Wu et al. (2018)). Our work, which makes explicit the dependence of generalization on the underlying metric and on the form of noise present, provides a framework to reconcile those latter, seemingly clashing claims.

TODO: COMPARE TO CHAUDHARI!

TODO: COMPARE TO DYER!

6. Conclusion

Applying this formalism, we propose a regularizing term that **causes large-batch GD to emulate small-batch SGD**, thus completing a project suggested 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). We verify the correctness and practicality of this regularizer on typical smooth CIFAR-10 and MNIST architectures. We also present a model-selection heuristic that depends only on statistics measured before any optimization. Finally, we construct a counter-intuitive loss landscape wherein, for arbitrarily small learning rates, SGD cycles counterclockwise around a circle of minima. Our mechanism differs from that discovered by Chaudhari & Soatto (2018), and we discuss the thermodynamic significance of both.

Role of Covariance

New Questions

Acknowledgements

We thank Dan A. Roberts and Sho Yaida for patient introductions to their work and for precisely posing several of the questions we answer here. We feel deeply grateful to Sho Yaida and Josh B. Tenenbaum for their compassionate guidance. We appreciate the generosity of Andrzej Banburski and Wenli Zhao in offering crucial feedback on writing.

References

References

Bartlett, P., Foster, D., and Telgarsky, M. Spectrally-normalized margin bounds for neural networks. *NeurIPS*, 2017.

Bonnabel, S. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 2013.

Bottou, L. Stochastic gradient learning in neural networks. *Neuro-Nîmes*, 1991.

Cauchy, A.-L. Méthode générale pour la résolution des systémes d'équations simultanées. *Comptes rendus de l'Académie des Sciences*, 1847.

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

Dinh, L., Pascanu, R., Bengio, S., and Bengio, Y. Sharp minima can generalize for deep nets. *ICLR*, 2017.

Dyson, F. The radiation theories of tomonaga, schwinger, and feynman. *Physical Review*, 1949a.

Dyson, F. The s matrix in quantum electrodynamics. *Physical Review*, 1949b.

Gell-Mann, M. and Goldberger, M. Scattering of low-energy photons by particles of spin $\frac{1}{2}$. *Physical Review*, 1954.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd. *Data @ Scale*, 2018.

Hoffer, E., Hubara, I., and Soudry, D. Train longer, generalize better. *NeurIPS*, 2017.

Jastrzębski, S., Kenton, Z., Arpit, D., N., B., Fischer, A., Y., B., and A., S. Three factors influencing minima in sgd. *Arxiv Preprint*, 2018.

Keskar, N., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. On large-batch training for deep learning: Generalization gap and sharp minima. *ICLR*, 2017.

Kiefer, J. and Wolfowitz, J. Stochastic estimation of the maximum of a regression function. *Annals of Mathematical Statistics*, 1952.

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

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

Nesterov, Y. Lectures on convex optimization: Minimization of smooth functions. *Springer Applied Optimization 87, Section 2.1*, 2004.

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

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

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

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

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

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

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

Wu, L., C., M., and E, W. How sgd selects the global minima in over-parameterized learning. *NeurIPS*, 2018.

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

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

A. Derivation of Diagram Rules

Dyson Series for Iterative Optimizers

If a density ρ governs a point θ in weight space, then after a sequence of updates $\theta \mapsto \theta - \eta^{\mu\nu}\nabla_\mu l(\theta)$ on losses $(l_t : 0 \leq t < T)$, the following density (up to an error term whose Taylor series vanishes; all perturbative results will implicitly carry such terms) will govern the new point:

$$\exp\left(+\eta^{\mu\nu}\nabla_\mu l_{T-1}(\theta)\nabla_\nu\right)\cdots\exp\left(+\eta^{\mu\nu}\nabla_\mu l_0(\theta)\nabla_\nu\right)\rho \tag{7}$$

or $\prod \exp\left(+\eta\nabla l\nabla\right)\rho$ for short. The exponent above is a linear operator that acts on a space of sufficiently smooth maps; in particular, the ∇_ν does not act on the $\nabla_\mu l(\theta)$ with which it pairs. Integrating by parts, we write the expectation over initial values after T steps of a function s of weight space (e.g. s may be test or train loss) as:

$$\int_\theta \rho(\theta)\left(\prod_{0\leq t\leq T}\exp\left(-\eta^{\mu\nu}\nabla_\mu l(\theta)\nabla_\nu\right)s\right)(\theta) \tag{8}$$

Since the exponentials above might not commute, we may not compose the product of exponentials into an exponential of a sum. We instead compute an expansion in powers of η. Setting the initialization $\rho(\theta) = \delta(\theta - \theta_0)$ to be deterministic, and labeling as θ_t the weight after t steps, we find:

$$s(\theta_T) = \sum_{0\leq d<\infty}(-\eta)^d \sum_{\substack{(d_t:0\leq t<T)\\ \sum_t d_t=d}}\left(\prod_{0\leq t<T}\frac{(\nabla l_t(\theta)\nabla)^{d_t}}{d_t!}\right)s(\theta_0) \tag{9}$$

B. Tutorial on Diagram Rules

C. Diagram Rules vs Direct Perturbation

D. The η-Series' Domain of Convergence