TOPOLOGY AND GEOMETRY OF DEEP RECTIFIED NETWORK OPTIMIZATION LANDSCAPES

C. Daniel Freeman
Department of Physics
University of California at Berkeley
Berkeley, CA 94720, USA
daniel.freeman@berkeley.edu

Joan Bruna
Courant Institute of Mathematical Sciences
New York University
New York, NY 10011, USA
bruna@cims.nyu.edu

ABSTRACT

The loss surface of deep neural networks has recently attracted interest in the optimization and machine learning communities as a prime example of high-dimensional non-convex problem. Some insights were recently gained using spin glass models, but at the expense of strongly simplifying the nonlinear nature of the model.

In this work, we do not make any such assumption and study conditions on the data distribution and model architecture that prevent the existence of bad local minima. Our theoretical work quantifies and formalizes two important *folklore* facts: (i) the landscape of deep linear networks is radically different from that of deep half-rectified ones, thus implying that mean-field approximations are unable to capture essential nonlinear behavior, and (ii) that the energy landscape in the non-linear case is fundamentally controlled by the interplay between the smoothness of the data distribution and model over-parametrization. These results are in accordance with empirical practice and recent literature.

The conditioning of gradient descent is the next challenge we address. We study this question through the geometry of the level sets, and we introduce an algorithm to efficiently estimate the regularity of such sets on large-scale networks. Our empirical results show that these level sets remain connected throughout all the learning phase, suggesting a near convex behavior, but they become exponentially more curvy as the energy level decays, in accordance to what is observed in practice with very low curvature attractors.

1 INTRODUCTION

- Context of the problem
- Related work: Spin glass, recent results from Shamir. Gradient Descent converges to minimizers (Jordan Recht et al).
- Topology of the level sets. Main result on connectedness of level sets.
- Geometry of the level sets. Algorithm to estimate the geodesics along level sets. Measure of curvature of these sets.

———

Optimization is a critical component in deep learning, governing its success in different areas of computer vision, speech processing and natural language processing. The prevalent optimization strategy is Stochastic Gradient Descent, invented by Robbins and Munro in the 50s. On the one hand, the performance of SGD is better than one could expect in generic, arbitrary non-convex loss surfaces. On the other hand, one

may wonder if this generic optimization strategy can be adapted to operate in the class of loss surfaces defined by deep neural networks. Some reasons to believe that such adaptation is possible come from the variety of modifications of SGD algorithms yielding significant speedups **????**. This raises a number of theoretical questions as to why neural network optimization does not suffer in practice from poor local minima. Likewise, it introduces opportunities to leverage training data into adapted optimization.

The loss surface of deep neural networks has recently attracted interest in the optimization and machine learning communities as a paradigmatic example of a hard, high-dimensional, non-convex problem. Recent work has explored models from statistical physics **?**, such as spin glasses **?**, in order to understand the macroscopic properties of the system, but at the expense of strongly simplifying the nonlinear nature of the model. In this proposal, we do not make any such assumption and study conditions on the data distribution and model architecture that prevent the existence of bad local minima. Together with recent results that rigorously establish that gradient descent does not get stuck on saddle points **?**, our first objective will yield guarantees that gradient descent converges to a global optimum in deep rectified networks.

The loss surface $F(\theta)$ of a given model can be expressed in terms of its level sets Ω_λ, which contain all parameters θ yielding a loss smaller or equal than the energy level λ: $\Omega_\lambda = \{\theta \; ; \; F(\theta) \leq \lambda\}$. A first question we address concerns the topology of these level sets, i.e. under which conditions they are connected. Connected level sets imply that one can always find a descent direction at each energy level, and therefore that no poor local minima can exist. One can verify that linear neural networks (cascaded linear operators without point-wise nonlinearities), despite defining a non-convex loss $F(\theta)$, have connected level sets and therefore can be optimized globally using gradient descent strategies **?**. It is also easy to verify that the non-linear case is not true in general **?**, and our first objective is thus to characterize the interplay between data distribution, overparametrization and model architecture that is needed to obtain connected level sets **?**.

Beyond the question of whether the loss contains poor local minima, the immediate follow-up question that determines the convergence of algorithms in practice is the local conditioning of the loss surface. It is thus related not to the topology but to the shape or geometry of the level sets. As the energy level decays, one expects the level sets to reveal more complex irregular structures, which correspond to regions where $F(\theta)$ has small curvature.

2 Topology of Level Sets

Let P be a probability measure on a product space $\mathcal{X} \times \mathcal{Y}$, where we assume $\mathcal{X}$ and $\mathcal{Y}$ are Euclidean vector spaces for simplicity. Let $\{(x_i, y_i)\}_i$ be an iid sample of size L drawn from P defining the training set. We consider the classic empirical risk minimization of the form

$$F_e(\theta) = \frac{1}{L} \sum_{l=1}^{L} \|\Phi(x_i; \theta) - y_i\|^2 \, , \tag{1}$$

where $\Phi(x; \theta)$ encapsulates the feature representation that uses parameters $\theta \in \mathbb{R}^S$. In a deep neural network, this parameter contains the weights and biases used in all layers. For convenience, in our analysis we will also use the oracle risk minimization:

$$F_o(\theta) = \mathbb{E}_{(X,Y)\sim P} \|\Phi(X; \theta) - Y\|^2 \, , \tag{2}$$

2.1 Poor local minima characterization from topological conectedness

We define the level set of $F(\theta)$ as

$$\Omega_F(\lambda) = \{\theta \in \mathbb{R}^S \; ; \; F(\theta) \leq \lambda\} \, . \tag{3}$$

The first question we study is the structure of critical points of $F_e(\theta)$ and $F_o(\theta)$ when Φ is a multilayer neural network. In particular, we are interested to know whether F_e has local minima which are not global minima. This question is answered by knowing whether $\Omega_F(\lambda)$ is connected at each energy level λ:

Proposition 2.1. *If $\Omega_F(\lambda)$ is connected for all λ then every local minima of $F(\theta)$ is a global minima.*

Proof: Suppose that θ_1 is a local minima and θ_2 is a global minima, but $F(\theta_1) > F(\theta_2)$. If $\lambda = F(\theta_1)$, then clearly θ_1 and θ_2 both belong to $\Omega_F(\lambda)$. Suppose now that $\Omega_F(\lambda)$ is connected. Then we could find a smooth (i.e. continuous and differentiable) path $\gamma(t)$ with $\gamma(0) = \theta_1$, $\gamma(1) = \theta_2$ and $F(\gamma(t)) \leq \lambda = F(\theta_1)$. In particular, as $t \to 0$, we have

$$\begin{aligned} F(\gamma(t)) &= F(\theta_1) + t\langle \nabla F(\theta_1), \dot{\gamma}(0)\rangle + \frac{t^2}{2}\left(\dot{\gamma}(0)^T HF(\theta_1)\dot{\gamma}(0) + \langle \nabla F(\theta_1), \ddot{\gamma}(0)\rangle\right) + o(t^2) \\ &= F(\theta_1) + \frac{t^2}{2}\dot{\gamma}(0)^T HF(\theta_1)\dot{\gamma}(0) + o(t^2)\ , \end{aligned}$$

which shows that $F(\gamma(t)) \leq F(\theta_1)$ for all t is incompatible with $H(\theta_1) \succeq 0$ and therefore $\Omega_F(\lambda)$ cannot be connected $\square$.

2.2 The Linear Case

A particularly simple but insightful case is when F is a multilayer network defined by

$$\Phi(x;\theta) = W_K \dots W_1 x\ ,\ \theta = (W_1, \dots, W_K)\ . \tag{4}$$

This model defines a non-convex (and non-concave) loss $F_e(\theta)$. It has been shown in **?** and **?** (concurrently with our work) that in this case, every local minima is a global minima. For completeness, we provide here an alternative proof of that result.

We have the following result.

Proposition 2.2. *Let $W_1, W_2, \dots, W_K$ be weight matrices of sizes $n_k \times n_{k+1}$, $k < K$, and let $F_e(\theta)$, $F_o(\theta)$ denote the risk minimizations using Φ as in (4). Assume that $n_j \geq \min(n_1, n_K)$ for $j = 2 \dots K-1$ [TODO I think this is not necessary]. Then $\Omega_{F_e}(\lambda)$ is connected for all λ, as well as Ω_{F_o}, and therefore there are no poor local minima.*

Proof: We proceed by induction over the number of layers K. For $K = 1$, the loss $F(\theta)$ is convex. Let θ_1, θ_2 be two arbitrary points in a level set Ω_λ. Thus $L(\theta_1) \leq \lambda$ and $L(\theta_2) \leq \lambda$. We have

$$L(t\theta_1 + (1-t)\theta_2) \leq tL(\theta_1) + (1-t)L(\theta_2) \leq \lambda\ ,$$

and thus a linear path is sufficient in that case to connect θ_1 and θ_2.

Suppose the result is true for $K-1$. Let $\theta_1 = (W_1^1, \dots, W_K^1)$ and $\theta_2 = (W_1^2, \dots, W_K^2)$ with $L(\theta_1) \leq \lambda$, $L(\theta_2) \leq \lambda$. For each $W_1, \dots, W_K$, we denote $\tilde{W}_j = W_j$ for $j < K-1$ and $\tilde{W}_{K-1} = W_K W_{K-1}$. By induction hypothesis, the loss expressed in terms of $\tilde{\theta} = (\tilde{W}_1, \dots, \tilde{W}_{K-1})$ is connected between $\tilde{\theta}_1$ and $\tilde{\theta_2}$. Let $\tilde{W}_{K-1}(t)$ the corresponding path projected in the last layer. We just need to produce a path in the variables $W_{K-1}(t)$, $W_K(t)$ such that (i) $W_{K-1}(0) = W_{K-1}^1$, $W_{K-1}(1) = W_{K-1}^2$, (ii) $W_K(0) = W_K^1$, $W_K(1) = W_K^2$, and (iii) $W_K(t)W_{K-1}(t) = \tilde{W}_{K-1}(t)$ for $t \in (0,1)$. We construct it as follows. Let

$$W_K(t) = tW_K^2 + (1-t)W_K^1 + t(1-t)V\ ,$$
$$W_{K-1}(t) = W_K(t)^\dagger \tilde{W}_{K-1}(t)\ ,$$

where $W_K(t)^\dagger = (W_K(t)^T W_K(t))^{-1} W_K(t)^T$ denotes the pseudoinverse and V is a $n_{K-1} \times n_K$ matrix drawn from a iid distribution. Conditions (i) and (ii) are immediate from the definition, and condition (iii) results from the fact that

$$W_K(t)W_K(t)^\dagger = \mathbf{I}_{N_K}\ ,$$

since $W_K(t)$ has full rank for all $t \in (0,1)$. $\square$.

2.3 HALF-RECTIFIED NONLINEAR CASE

We now study the setting given by

$$\Phi(x;\theta) = W_K \rho W_{K-1} \rho \dots \rho W_1 x \ , \ \theta = (W_1, \dots, W_K) \ , \tag{5}$$

where $\rho(z) = \max(0, z)$. The biases can be implemented by replacing the input vector x with $\overline{x} = (x, 1)$ and by rebranding each parameter matrix as

$$\overline{W}_i = \left(\begin{array}{c|c} W_i & b_i \\ \hline 0 & 1 \end{array} \right) \ ,$$

where b_i contains the biases for each layer. For simplicity, we continue to use W_i and x in the following.

2.3.1 NONLINEAR MODELS ARE GENERALLY DISCONNECTED

One may wonder whether the same phenomena of global connectedness also holds in the half-rectified case. A simple motivating counterexample shows that this is not the case in general. Consider a simple setup with $X \in \mathbb{R}^2$ drawn from a mixture of two Gaussians $\mathcal{N}_{-1}$ and $\mathcal{N}_1$, and let $Y = (X - \mu_Z) \cdot Z$, where Z is the (hidden) mixture component taking $\{1, -1\}$ values. Let $\hat{Y} = \Phi(X; \{W_1, W_2\})$ be a single-hidden layer ReLU network, with two hidden units, illustrated in Figure ??. Let θ^A be a configuration that bisects the two mixture components, and let θ^B the same configuration, but swapping the bisectrices. One can verify that they can both achieve arbitrarily small risk by letting the covariance of the mixture components go to 0. However, any path that connects θ^A to θ^B must necessarily pass through a point in which W_1 has rank 1, which leads to an estimator with risk at least $1/2$.

In fact, it is easy to see that this counter-example can be extended to any generic half-rectified architecture, if one is allowed to adversarially design a data distribution. For any given $\Phi(X;\theta)$ with arbitrary architecture and current parameters $\theta = (W_i)$, let $\mathcal{P}_\theta = \{\mathcal{A}_1, \dots, \mathcal{A}_S\}$ be the underlying tesselation of the input space given by our current choice of parameters; that is, $\Phi(X;\theta)$ is piece-wise linear and $\mathcal{P}_\theta$ contains those pieces. Now let X be any arbitrary distribution with density $p(x) > 0$ for all $x \in \mathbb{R}^n$, for example a Gaussian, and let $Y \mid X = \Phi(X;\theta)$. Since Φ is invariant under permutations θ_σ of its hidden layers, it is easy to see that one can find two parameter values $\theta_A = \theta$ and $\theta_B = \theta_\sigma$ such that $F_o(\theta_A) = F_o(\theta_B) = 0$, but any continuous path $\gamma(t)$ from θ_A to θ_B will have a different tesselation and therefore won't satisfy $F_o(\gamma(t)) = 0$.

This illustrates an intrinsic difficulty in the optimization landscape if one is after *universal* guarantees that do not depend upon the data distribution. This difficulty is non-existent in the linear case and not easy to exploit in mean-field approaches such as **?**, but is easily detected as soon as one considers a non-linear model, and shows that in general we should not expect to obtain connected level sets. However, connectedness can be recovered if one is willing to accept a small increase of energy. Our main result shows that the amount by which the energy is allowed to increase is upper bounded by a quantity that trades-off model overparametrization and smoothness in the data distribution.

For that purpose, we start with a characterization of the oracle loss, and for simplicity let us assume $Y \in \mathbb{R}$ and let us first consider the case with a single hidden layer.

2.3.2 PRELIMINARIES

Before proving our main result, we need to introduce preliminary notation and results. We first describe the case with a single hidden layer of size m.

We define

$$e(m) = \min_{W_1 \in \mathbb{R}^{m \times n}, W_2 \in \mathbb{R}^m} \mathbb{E}\{|\Phi(X;\theta) - Y|^2\} + \kappa \|W_2\|^2 \ . \tag{6}$$

to be the oracle risk using m hidden units with Ridge regression. It is a well known result by Hornik and Cybenko that a single hidden layer is a universal approximator under very mild assumptions, i.e. $\lim_{m\to\infty} e(m) = 0$. This result merely states that our statistical setup is consistent, and it should not be surprising to the reader familiar with classic approximation theory. A more interesting question is the rate at which $e(m)$ decays, which depends on the smoothness of the joint density $(X, Y) \sim P$ relative to the nonlinear activation family we have chosen.

For convenience, we redefine $W = W_1$ and $\beta = W_2$ and $Z(W) = \max(0, WX)$. We also write $z(w) = \max(0, \langle w, X\rangle)$ where $(X, Y) \sim P$ and $w \in \mathbb{R}^N$ is any deterministic vector. Let $\Sigma_X = \mathbb{E}_P XX^T \in \mathbb{R}^{N\times N}$ be the covariance operator of the random input X. We assume $\|\Sigma_X\| < \infty$.

A fundamental property that will be essential to our analysis is that, despite the fact that Z is nonlinear, the "pseudo-metric" $\langle w_1, w_2\rangle_Z := \mathbb{E}_P\{z(w_1)z(w_2)\}$ is locally equivalent to the linear metric $\langle w_1, w_2\rangle_X = \mathbb{E}_P\{w_1^T XX^T w_2\} = \langle w_1, \Sigma_X w_2\rangle$, and that the linearization error decreases with the angle between w_1 and w_2. Without loss of generality, we assume here that $\|w_1\| = \|w_2\| = 1$, and we write $\|w\|_Z^2 = \mathbb{E}\{|z(w)|^2\}$.

Proposition 2.3. *Let $\alpha = \cos^{-1}(\langle w_1, w_2\rangle)$ be the angle between unitary vectors w_1 and w_2 and let $w_m = \frac{w_1+w_2}{\|w_1+w_2\|}$ be their unitary bisector. Then*

$$\frac{1+\cos\alpha}{2}\|w_m\|_Z^2 - 2\|\Sigma_X\|\left(\frac{1-\cos\alpha}{2} + \sin^2\alpha\right) \leq \langle w_1, w_2\rangle_Z \leq \frac{1+\cos\alpha}{2}\|w_m\|_Z^2 \,. \tag{7}$$

The term $\|\Sigma_X\|$ is overly pessimistic: we can replace it by the energy of X projected into the subspace spanned by w_1 and w_2 (which is bounded by $2\|\Sigma_X\|$). When α is small, a Taylor expansion of the trigonometric terms reveals that

$$\begin{aligned}
\frac{2}{3\|\Sigma_X\|}\langle w_1, w_2\rangle &= \frac{2}{3\|\Sigma_X\|}\cos\alpha = \frac{2}{3\|\Sigma_X\|}(1 - \frac{\alpha^2}{2} + O(\alpha^4)) \\
&\leq (1-\alpha^2/4)\|w_m\|_Z^2 - \|\Sigma_X\|(\alpha^2/4 + \alpha^2) + O(\alpha^4) \\
&\leq \langle w_1, w_2\rangle_Z + O(\alpha^4) \,,
\end{aligned}$$

and similarly

$$\langle w_1, w_2\rangle_Z \leq \langle w_1, w_2\rangle\|w_m\|_Z^2 \,.$$

The local behavior of parameters w_1, w_2 on our regression problem is thus equivalent to that of having a linear layer, provided w_1 and w_2 are sufficiently close to each other. This result can be seen as a spoiler that increasing the hidden layer dimensionality M will increase the chances to encounter pairs of vectors w_1, w_2 with small angle; and therefore some hope of approximating the previous linear behavior thanks to the small linearization error.

In order to control the connectedness, we will also require another quantity. Given a hidden layer of size m with current parameters $W \in \mathbb{R}^{m\times n}$, we define a "robust compressibility" factor as

$$\delta_W(n, \alpha; m) = \min_{\|\gamma\|_0 \leq n, \sup_i |\angle(\tilde{w}_i, w_i)| \leq \alpha} \mathbb{E}\{|Y - \gamma Z(\tilde{W})|^2 + \kappa\|\gamma\|^2\} \,, \; (n \leq m) \,. \tag{8}$$

This quantity thus measures how easily one can compress the current hidden layer representation, by keeping only a subset of its units, but allowing these units to rotate by a small amount. It is a form of n-width similar to Kolmogorov width **?**.

2.3.3 Main result

Our main result considers now a non-asymptotic scenario given by some fixed size M of the hidden layer. Given two parameter values $\theta^A = (W_1^A, W_2^A) \in \mathcal{W}$ and $\theta^B = (W_1^B, W_2^B)$ with $F_o(\theta^{\{A,B\}}) \leq \lambda$, we show that there exists a continuous path $\gamma : [0, 1] \to \mathcal{W}$ connecting θ^A and θ^B such that its oracle risk is uniformly bounded by $\max(\lambda, \epsilon)$, where ϵ decreases with model overparametrization.

Theorem 2.4. *There exists a continuous path $\gamma : [0, 1] \to \mathcal{W}$ such that $\gamma(0) = \theta^A$, $\gamma(1) = \theta^B$ and*

$$F_o(\gamma(t)) \leq \max(\lambda, \epsilon) \ , \ \textit{with} \tag{9}$$

$$\epsilon = \inf_{n,\alpha} \left(\max \left\{ e(n), \delta_{W_1^A}(M, 0; M), \delta_{W_1^A}(M - n, \alpha; M), \delta_{W_1^B}(M, 0; M), \delta_{W_1^B}(M - n, \alpha; M) \right\} + \mathbb{E}(|Y|^2)\kappa^{-1}\alpha \right) . \tag{10}$$

Corollary 2.5. *If M increases, the energy gap ϵ goes to zero. Here we use the fact that $\delta(\lambda m; \epsilon(m); m) \to 0$ as $m \to \infty$. (find the rate).*

Remarks:

- Ridge regression term.
- Extension to empirical risk. It is straightforward just change the metric.
- Extension to several layers.

3 Geometry of Level Sets

3.1 The Greedy Algorithm

For a pair of models with network parameters θ_i, θ_j, each with $F_e(\theta)$ below a threshold L_0, we aim to efficiently generate paths in the space of weights where the empirical loss along the path remains below the threshold. These paths are continuous curves belonging to $\Omega_F(\lambda)$–that is, the level sets of the loss function of interest.

We provide a greedy algorith, Dynamic String Sampling, which finds such a path below.

Algorithm 1 Greedy Dynamic String Sampling

$L_0 \leftarrow$ Threshold below which path will be found
$\Phi_1 \leftarrow$ randomly initialize θ_1, train $\Phi(x_i\ \theta_1)$ to L_0
$\Phi_2 \leftarrow$ randomly initialize θ_2, train $\Phi(x_i\ \theta_2)$ to L_0
BeadList $\leftarrow(\Phi_1, \Phi_2)$
Depth $\leftarrow 0$
procedure FindConnection(Φ_1, Φ_2)
    $t^* \leftarrow$ t such that $\left.\frac{d\gamma(\theta_1,\theta_2,t)}{dt}\right|_t = 0$ OR $t = 0.5$
    $\Phi_3 \leftarrow$ train $\Phi(x_i; t^*\theta_1 + (1 - t^*)\theta_2)$ to L_0
    BeadList $\leftarrow$ insert(Φ_3, after Φ_1, BeadList)
    $MaxError_1 \leftarrow max_t(F_e(t\theta_3 + (1 - t)\theta_1))$
    $MaxError_2 \leftarrow max_t(F_e(t\theta_2 + (1 - t)\theta_3))$
    if $MaxError_1 > L_0$ **then return** FindConnection(Φ_1, Φ_3)
    if $MaxError_2 > L_0$ **then return** FindConnection(Φ_3, Φ_2)
    Depth $\leftarrow$ Depth+1

The algorithm recursively builds a string of models in the space of weights which continuously connect θ_i to θ_j. Models are added and trained until the pairwise linearly interpolated loss, i.e. $\max_t F_e(t\theta_i + (1 - t)\theta_j)$ for $t \in (0, 1)$, is below the threshold, L_0, for every pair of neighboring models on the string. We provide a cartoon of the algorithm in Fig. 1.

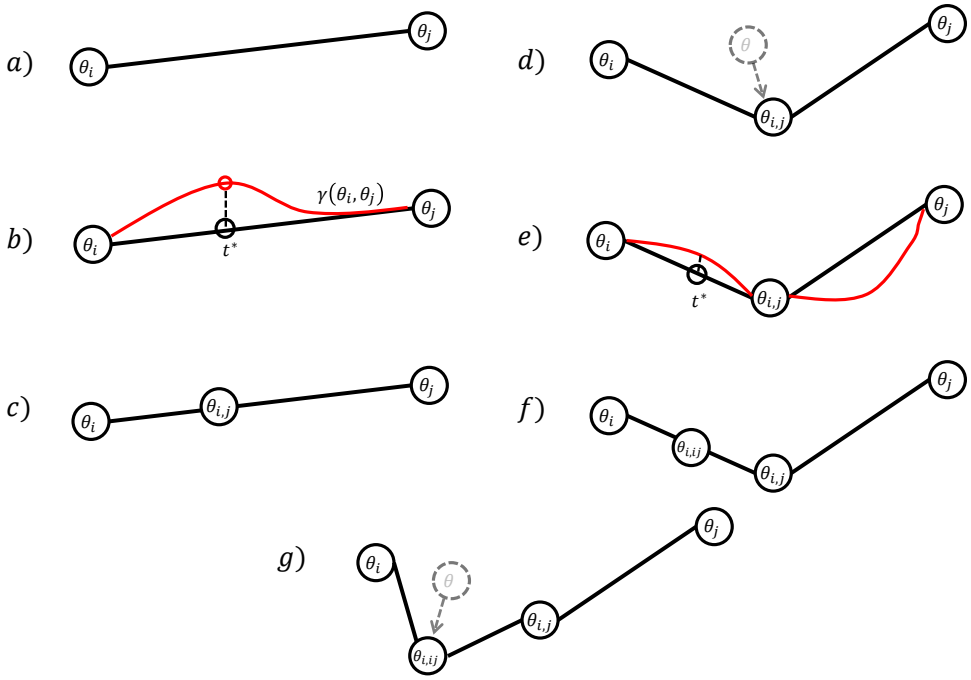

Figure 1: A cartoon of the algorithm. $a)$: The initial two models with approximately the same loss, L_0. $b)$: The interpolated loss curve, in red, and its global maximum, occuring at $t = t^*$. $c)$: The interpolated model $\Theta(\theta_i, \theta_j, t^*)$ is added and labeled $\theta_{i,j}$. $d)$: Stochastic gradient descent is performed on the interpolated model until its loss is below αL_0. $e)$: New interpolated loss curves are calculated between the models, pairwise on a chain. $f)$: As in step $c)$, a new model is inserted at the maxima of the interpolated loss curve between θ_i and $\theta_{i,j}$. $g)$: As in step $d)$, gradient descent is performed until the model has low enough loss.

3.2 FAILURE CONDITIONS AND PRACTICALITIES

While the algorithm presented will faithfully certify two models are connected if the algorithm converges, it is worth emphasizing that the algorithm does not guarantee that two models are disconnected if the algorithm fails to converge. In general, the problem of determining if two models are connected can be made arbitrarily difficult by choice of a particularly pathological geometry for the loss function, so we are constrained to heuristic arguments for determining when to stop running the algorithm. Thankfully, in practice, loss function geometries for problems of interest are not intractably difficult to explore. We comment more on diagnosing disconnections more carefully in section SYMMETRYDISCONNECT.

Further, if the $\mathbf{MaxError}$ exceeds L_0 for every new recursive branch as the algorithm progresses, the worst case runtime scales as $O(\exp(\mathbf{Depth}))$. Empirically, we find that the number of new models added at each depth does grow, but eventually saturates, and falls for a wide variety of models and architectures, so that the typical runtime is closer to $O(\mathrm{poly}(\mathbf{Depth}))$—at least up until a critical value of L_0. We comment more on this in section NUMERICALDISCUSSION.

Finally, we find that training Φ_3 to αL_0 for $\alpha < 1$ in line 8 of the algorithm tends to aid convergence without noticeably impacting our numerics.

4 NUMERICAL EXPERIMENTS

For our numerical experiments, we aimed to extract qualitative features of both small, toy networks, as well as of larger workhorse networks suitable for use on real world tasks (e.g. MNIST). At its core, the maximum interpolated error (i.e., (**??**)) is a measure of problem nonconvexity—or, more precisely, of the nonconvexity of the loss surface of a given architecture on a particular learning problem.

4.1 POLYNOMIAL REGRESSION

Polynomial function regression is a task for which small neural networks can achieve extremely high accuracy. For our numerical experiments, we studied a 1-4-4-1 fully connected multilayer perceptron style architecture with RELU activation and RMSProp optimization. For ease-of-analysis, we restricted the family of polynomials to be strictly contained in the interval $x \in [0, 1]$ and $f(x) \in [0, 1]$.

Discussion of different Loss functions

etc.

4.2 CONVOLUTIONAL NEURAL NETWORKS

4.3 RECURRENT NEURAL NETWORKS

5 DISCUSSION

- Future: Generalization Error Question.

A CONSTRAINED DYNAMIC STRING SAMPLING

While the algorithm presented in Sec. 3.1 is fast for sufficiently smooth families of loss surfaces with few saddle points, here we present a slightly modified version which, while slower, provides more control over the convergence of the string. Instead of training intermediate models via full SGD to a desired accuracy,

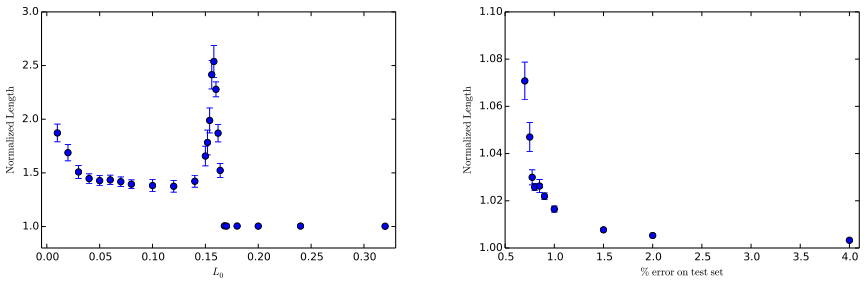

Figure 2: Normalized geodesic length as a function of the energy level for two different models. Left: a "low-dimensional" neural network attempting to fit a cubic polynomial. Right: a convolutional neural network on MNIST. Whereas the cubic fitting displays a heavily non-convex structure at mid-energy values, the MNIST example qualitatively behaves as a convex loss, in which geodesics approach straight lines.

intermediate models will be subject to a constraint that ensures they are "close" to the neighboring models on the string. Specifically, intermediate models will be constrained to the unique hyperplane in weightspace equidistant from its two neighbors. This is similar to a sort of "L_1 regularization" where the loss function for a given model on the string, θ_i, has an additional term $\tilde{L}(\theta) = L(\theta) + \zeta(\|\theta_{i-1} - \theta_i\| + \|\theta_{i+1} + \theta_i\|)$. The strength of the ζ regularization term controls the "springy-ness" of the weightstring. note: make this more precise, the hyperplane constraint is stronger than the L_1 constraint...L_1 only keeps the model in a ball close to the midpoint between the models.

Because adapting DSS to use this constraint is straightforward, here we will describe an alternative "breadth-first" approach wherein models are trained in parallel until convergence. This alternative approach has the advantage that it will indicate a disconnection between two models "sooner" insofar as it will be clear two models cannot be connected once the loss on either of the two initial models, θ_1 or θ_2, is less than $\Gamma(\theta_1, \theta_2)$. The precise geometry of the loss surface will dictate which approach to use in practice.

Given two random models σ_i and σ_j where $|\sigma_i - \sigma_j| < \kappa$, we aim to follow the evolution of the family of models connecting σ_i to σ_j. Intuitively, almost every continuous path in the space of random models connecting σ_i to σ_j has, on average, the same (high) loss. For simplicity, we choose to initialize the string to the linear segment interpolating between these two models. If this entire segment is evolved via gradient descent, the segment will either evolve into a string which is entirely contained in a basin of the loss surface, or some number of points will become fixed at a higher loss. These fixed points are difficult to detect directly, but will be indirectly detected by the persistence of a large interpolated loss between two adjacent models on the string.

The algorithm proceeds as follows:

(0.) Initialize model string to have two models, σ_i and σ_j.

1. Begin training all models to the desired loss, keeping the instantaneous loss of all models being trained approximately constant..

2. If the pairwise interpolated loss $\gamma(\sigma_n, \sigma_{n+1})$ exceeds a tolerance α_1, insert a new model at the maximum of the interpolated loss between these two models. For simplicity, this tolerance is chosen to be $(1 + \alpha_1^*)$ times the instantaneous loss of all other models on the string.

3. Repeat steps (1) and (2) until all models (and interpolated errors) are below a threshold loss L_0, or until a chosen failure condition (see 3.2).

B PROOFS

B.1 PROOF OF PROPOSITION 2.3

Cut the integral into the three cones and use trivial bounds.

B.2 PROOF OF THEOREM 2.4

A path from θ^A to θ^B will be constructed as follows:

1. from θ^A to θ_{lA}, the best linear predictor using the same first layer.
2. from θ_{lA} to θ_{sA}, the best $M-n$ approximation using perturbed atoms,
3. from θ_{sA} to θ^* the oracle n term approximation,
4. from θ^* to θ_{sB},
5. from θ_{sB} to θ_{lB},
6. from θ_{lB} to θ^B.

The subpaths (1) and (6) only involve changing the parameters of the second layer, which define a convex loss. Therefore a linear path is sufficient. Subpaths (3) and (4) can also be constructed using only parameters of the second layer, by observing that one can fit into a single $n \times M$ parameter matrix both the $M-n$ term approximation and the best n-term approximation. A linear path is therefore also sufficient.

We finally need to show how to construct the subpaths (2) and (5). Let $\tilde{W}_A$ be the resulting perturbed first-layer parameter matrix with $M-n$ sparse coefficients γ_A. Let us consider an auxiliary regression of the form

$$\overline{W} = [W^A; \tilde{W}_A]$$

and regression parameters

$$\overline{\beta}_1 = [\beta_1; 0] \ , \ \overline{\beta}_2 = [0; \gamma_A] \ .$$

Clearly

$$\mathbb{E}\{|Y - \overline{\beta}_1 \overline{W}|^2\} + \kappa \|\overline{\beta}_1\|^2 = \mathbb{E}\{|Y - \beta_1 W^A|^2\} + \kappa \|\beta_1\|^2$$

and similarly for $\overline{\beta}_2$. The augmented linear path $\eta(t) = (1-t)\overline{\beta}_1 + t\overline{\beta}_2$ thus satisfies

$$\forall\, t \ , \overline{L}(t) = \mathbb{E}\{|Y - \eta(t)\overline{W}|^2\} + \kappa \|\eta(t)\|^2 \leq \max(\overline{L}(0), \overline{L}(1)) \ .$$

Let us now approximate this augmented linear path with a path in terms of first and second layer weights. We consider

$$\eta_1(t) = (1-t)W^A + t\tilde{W}_A \ , \text{ and } \eta_2(t) = (1-t)\beta_1 + t\gamma_A \ .$$

We verify that

$$\begin{aligned} F_o(\{\eta_1(t), \eta_2(t)\}) &= \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} + \kappa \|\eta_2(t)\|^2 \\ &\leq \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} + \kappa((1-t)\|\beta_1\|^2 + t\|\gamma_A\|^2) \\ &= \overline{L}(t) + \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} - \mathbb{E}\{|Y - (1-t)\beta_1 Z(W^A) - t\gamma_A Z(\tilde{W}_A)|^2\} \end{aligned} \quad (11)$$

Finally, we verify that

$$\left| \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} - \mathbb{E}\{|Y - (1-t)\beta_1 Z(W^A) - t\gamma_A Z(\tilde{W}_A)|^2\} \right| \leq 4 \max(1, \sqrt{\mathbb{E}|Y^2|}) \|\Sigma_X\| \alpha(\kappa^{-1/2} + M\kappa^{-1}) + O(\alpha^2) \quad (12)$$

From Proposition 2.3, and using the fact that

$$\forall\, i \leq M,\, t \in [0,1]\ ,\ \left|\angle((1-t)w_i^A + t\tilde{w}_i^A; w_i^A)\right| \leq \alpha\ ,\ \left|\angle((1-t)w_i^A + t\tilde{w}_i^A; \tilde{w}_i^A)\right| \leq \alpha$$

we can write

$$(1-t)\beta_1[i]z(w_i^A) - t\gamma_A[i]z(\tilde{w}_i^A) \stackrel{d}{=} \eta_2(t)[i]z(\eta_1(t)[i]) + n_i\ ,$$

with $\mathbb{E}\{|n_i|^2\} \leq 4|\eta_2(t)[i]|^2\|\Sigma_X\|\alpha^2 + O(\alpha^4)$ and $\mathbb{E}|n_i| \leq 2|\eta_2(t)[i]|\alpha\sqrt{\|\Sigma_X\|}$ using concavity of the moments. Thus

$$\begin{aligned}
&\left|\mathbb{E}\{|Y - \eta_2(t)Z(\eta_1(t))|^2\} - \mathbb{E}\{|Y - (1-t)\beta_1 Z(W^A) - t\gamma_A Z(\tilde{W}_A)|^2\}\right| \\
\leq\ & 2\mathbb{E}\{\sum_i (Y - \eta_2(t)Z(\eta_1(t)))n[i]\} + \mathbb{E}\{|\sum_i n[i]|^2\} \\
\leq\ & 4\left(\sqrt{\mathbb{E}|Y^2|}\|\Sigma_X\|\alpha\|\eta_2\| + \alpha^2(\|\eta_2\|_1)^2\|\Sigma_X\|\right) \\
\leq\ & 4\max(1, \sqrt{\mathbb{E}|Y^2|})\|\Sigma_X\|\alpha(\|\eta_2\| + M\|\eta_2\|^2) + O(\alpha^2) \\
\leq\ & 4\max(1, \sqrt{\mathbb{E}|Y^2|})\|\Sigma_X\|\alpha(\kappa^{-1/2} + M\kappa^{-1}) + O(\alpha^2)\ .
\end{aligned}$$

B.3 Proof of Corollary 2.5

Use pigeonhole principle to control how many directions are within an angle smaller than ϵ.