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 and mean-field approximations, 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 has a radically different topology from that of deep half-rectified ones, 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

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. The empirical performance of SGD on these models is better than one could expect in generic, arbitrary non-convex loss surfaces, often aided by modifications yielding significant speedups Duchi et al. (2011); Hinton et al. (2012); Ioffe & Szegedy (2015); Kingma & Ba (2014). This raises a number of theoretical questions as to why neural network optimization does not suffer in practice from poor local minima.

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 Choromanska et al. (2015), in order to understand the macroscopic properties of the system, but at the expense of strongly simplifying the nonlinear nature of the model. Other authors have advocated that the real danger in high-dimensional setups are saddle points rather than poor local minima Dauphin et al. (2014), although recent results rigorously establish that gradient descent does not get stuck on saddle points Lee et al. (2016) but merely slowed down. Other notable recent contributions are Kawaguchi (2016), which further develops the spin-glass connection from Choromanska et al. (2015), Sagun et al. (2014) which also discusses the impact of stochastic vs plain gradient, Soudry & Carmon (2016), that studies Empirical Risk Minimization for piecewise multilayer neural networks

*Currently on leave from UC Berkeley.

under overparametrization (which needs to grow with the amount of available data), and Goodfellow et al. (2014), which provided insightful intuitions on the loss surface of large deep learning models and partly motivated our work. Lastly, the work Safran & Shamir (2015) studies some topological properties of homogeneous nonlinear networks and shows how overparametrization acts upon these properties, and the pioneering Shamir (2016) studied the distribution-specific hardness of optimizing non-convex objectives.

In this work, we do not make any linearity assumption and study conditions on the data distribution and model architecture that prevent the existence of bad local minima. The loss surface $F(\theta)$ of a given model can be expressed in terms of its level sets Ω_λ, which contain for each energy level λ all parameters θ yielding a loss smaller or equal than λ. 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. In absence of nonlinearities, deep (linear) networks have connected level sets Kawaguchi (2016). We first generalize this result to include ridge regression (in the two layer case) and provide an alternative, more direct proof of the general case. We then move to the half-rectified case and show that the topology is intrinsically different and clearly dependent on the interplay between data distribution and model architecture. Our main theoretical contribution is to prove that half-rectified single layer networks are asymptotically connected, and we provide explicit bounds that reveal the aforementioned interplay.

Beyond the question of whether the loss contains poor local minima or not, 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 exhibit more complex irregular structures, which correspond to regions where $F(\theta)$ has small curvature. In order to verify this intuition, we introduce an efficient algorithm to estimate the geometric regularity of these level sets by approximating geodesics of each level set starting at two random boundary points. Our algorithm uses dynamic programming and can be efficiently deployed to study mid-scale CNN architectures on MNIST, CIFAR-10 and RNN models on Penn Treebank next word prediction. Our empirical results show that these models have a nearly convex behavior up until their lowest test errors, with a single connected component that becomes more elongated as the energy decays. The rest of the paper is structured as follows. Section 2 presents our theoretical results on the topological connectedness of multilayer networks. Section 3 presents our path discovery algorithm and Section 4 covers the numerical experiments.

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

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

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

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

Our setup considers the case where $\mathcal{R}$ consists on either ℓ_1 or ℓ_2 norms, as we shall describe below. They correspond to well-known sparse and ridge regularization respectively.

2.1 Poor local minima characterization from topological connectedness

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.*

This proposition shows that a sufficient condition to prevent the existence of poor local minima is having connected level sets, but this condition is not necessary: one can have isolated local minima lying at the same energy level. This can be the case in systems that are defined up to a discrete symmetry group, such as multilayer neural networks. However, as we shall see next, this case puts the system in a brittle position, since one needs to be able to account for all the local minima (and there can be exponentially many of them as the parameter dimensionality increases) and verify that their energy is indeed equal.

2.2 The Linear Case

We first consider the particularly simple case where F is a multilayer network defined by

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

and the ridge regression $\mathcal{R}(\theta) = \|\theta\|^2$. This model defines a non-convex (and non-concave) loss $F_e(\theta)$. When $\kappa = 0$, it has been shown in Saxe et al. (2013) and Kawaguchi (2016) that in this case, every local minima is a global minima. We provide here an alternative proof of that result that uses a somewhat simpler argument and allows for $\kappa > 0$ in the case $K = 2$.

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 $j = 2 \dots K-1$. Then $\Omega_{F_e}(\lambda)$ (and Ω_{F_o}) is connected for all λ and all K when $\kappa = 0$, and for $\kappa > 0$ when $K = 2$; and therefore there are no poor local minima in these cases.*

This result highlights a certain mismatch between the picture of having no poor local minima and generalization. Incorporating regularization drastically changes the topology, and the fact that we are able to show connectedness only in the two-layer case with ridge regression is profound; we conjecture that extending it to deeper models requires a different regularization, perhaps using more general atomic norms Bach (2013). But we now move our interest to the nonlinear case, which is more relevant to our purposes.

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. 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 tessellation 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 $Y \mid X \stackrel{d}{=} \Phi(X;\theta)$. Since Φ is invariant under a subgroup of 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 tessellation and therefore won't satisfy $F_o(\gamma(t)) = 0$. Moreover, one can build on this counter-example to show that not only the level sets are disconnected, but also that there exist poor local minima. Let θ' be a different set of parameters, and $Y' \mid X \stackrel{d}{=} \Phi(X;\theta')$ be a different target distribution. Now consider the data distribution given by the mixture

$$X \mid p(x) \ , \ z \sim \text{Bernoulli}(\pi) \ , \ Y \mid X, z \stackrel{d}{=} z\Phi(X;\theta) + (1-z)\Phi(X;\theta') \ .$$

By adjusting the mixture component π we can clearly change the risk at θ and θ' and make them different, but we preserve the status of local minima of θ and θ'.

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 Choromanska et al. (2015), 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 and make some assumptions on the complexity of the regression task. 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 and ℓ_1 regularization: $\mathcal{R}(\theta) = \|\theta\|_1$.

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

$$e(m) = \min_{W_1 \in \mathbb{R}^{m \times n}, \|W_1(i)\|_2 \leq 1, W_2 \in \mathbb{R}^m} \mathbb{E}\{|\Phi(X;\theta) - Y|^2\} + \kappa \|W_2\|_1 \ . \tag{6}$$

to be the oracle risk using m hidden units with norm ≤ 1 and using sparse 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 quantity $[w_1, w_2]_Z := \mathbb{E}_P\{z(w_1)z(w_2)\}$ 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*

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

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

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

and similarly

$$[w_1, w_2]_Z \leq \langle w_1, w_2\rangle \|w_m\|_Z^2 \leq \|\Sigma_X\|\langle w_1, w_2\rangle \,.$$

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* of what is coming: increasing the hidden layer dimensionality m will increase the chances to encounter pairs of vectors w_1, w_2 with small angle; and with it some hope of approximating the previous linear behavior thanks to the small linearization error.

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

$$\delta_W(n, \alpha; m) = \min_{\|\gamma\|_0 \leq n, \sup_i |\angle(\tilde{w}_i, w_i)| \leq \alpha} \mathbb{E}\{|Y - \gamma Z(\tilde{W})|^2 + \kappa\|\gamma\|_1\} \,, \; (n \leq m) \,. \quad (8)$$

This quantity thus measures how easily one can compress the current hidden layer representation, by keeping only a subset of n its units, but allowing these units to move by a small amount controlled by α. It is a form of n-width similar to Kolmogorov width Donoho (2006) and is also related to robust sparse coding from Tang et al. (2013); Ekanadham et al. (2011).

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. *For any $\theta^A, \theta^B \in \mathcal{W}$ and $\lambda \in \mathbb{R}$ satisfying $F_o(\theta^{\{A,B\}}) \leq \lambda$, there exists a continuous path $\gamma : [0,1] \to \mathcal{W}$ such that $\gamma(0) = \theta^A$, $\gamma(1) = \theta^B$ and*

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

$$\epsilon = \inf_{n,\alpha}\left(\max\left\{e(n), \delta_{W_1^A}(m, 0; m), \delta_{W_1^A}(m-n, \alpha; m), \delta_{W_1^B}(m, 0; m), \delta_{W_1^B}(m-n, \alpha; m)\right\} + C_1\alpha + O(\alpha^2)\right) \,, \quad (10)$$

where C_1 is an absolute constant depending only on κ and P.

Some remarks are in order. First, our regularization term is currently a mix between ℓ_2 norm constraints on the first layer and ℓ_1 norm constraints on the second layer. We believe this is an artifact of our proof technique, and we conjecture that more general regularizations yield similar results. Next, this result uses the data distribution through the oracle bound $e(m)$ and the covariance term. The extension to empirical risk is accomplished by replacing the probability measure P by the empirical measure $\hat{P} = \frac{1}{L}\sum_l \delta((x,y) - (x_l, y_l))$. However, our asymptotic analysis has to be carefully reexamined to take into account and avoid the trivial regime when M outgrows L. A consequence of Theorem 2.4 is that as m increases, the model becomes asymptotically connected, as proven in the following corollary.

Corollary 2.5. *As m increases, the energy gap ϵ goes to zero and therefore the level sets become connected at all energy levels.*

This is consistent with the overparametrization results from Safran & Shamir (2015); Shamir (2016) and the general common knowledge amongst deep learning practitioners. Our next sections explore this question, and refine it by considering not only topological properties but also some rough geometrical measure of the level sets.

3 GEOMETRY OF LEVEL SETS

3.1 THE GREEDY ALGORITHM

The intuition behind our main result is that, for smooth enough loss functions and for sufficient overparameterization, it should be "easy" to connect two equally powerful models—i.e., two models with $F_o\theta^{A,B} \leq \lambda$. A sensible measure of this ease-of-connectedness is the normalized length of the geodesic connecting one model to the other: $|\gamma_{A,B}(t)|/|\theta_A - \theta_B|$. This length represents approximately how far of an excursion one must make in the space of models relative to the euclidean distance between a pair of models. Thus, convex models have a geodesic length of 1, because the geodesic is simply linear interpolation between models, while more non-convex models have geodesic lengths strictly larger than 1.

Because calculating the exact geodesic is difficult, we approximate the geodesic paths via a dynamic programming approach we call Dynamic String Sampling. We comment on alternative algorithms in Appendix A.

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 efficienly generate paths in the space of weights where the empirical loss along the path remains below L_0. These paths are continuous curves belonging to $\Omega_F(\lambda)$–that is, the level sets of the loss function of interest.

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. $\mathrm{max_t F_e(t\theta_i + (1-t)\theta_j)}$ 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 Appendix C.

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 Appendix E.

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.

To aid convergence, either of the choices in line 7 of the algorithm works in practice—choosing t^* at a local maximum can provide a modest increase in algorithm runtime, but can be unstable if the the calculated interpolated loss is particularly flat or noisy. $t^* = .5$ is more stable, but slower. 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. We provide further implementation details in 4.

4 Numerical Experiments

For our numerical experiments, we calculated normalized geodesic lengths for a variety of regression and classification tasks. In practice, this involved training a pair of randomly initialized models to the desired test loss value/accuracy/perplexity, and then attempting to connect that pair of models via the Dynamic String Sampling algorithm. We also tabulated the average number of "beads", or the number intermediate models needed by the algorithm to connect two initial models. For all of the below experiments, the reported losses and accuracies are on a restricted test set. For more complete architecture and implementation details, see our GitHub page.

The results are broadly organized by increasing model complexity and task difficulty, from easiest to hardest. Throughout, and remarkably, we were able to easily connect models for every dataset and architecture investigated except the one explicitly constructed counterexample discussed in Appendix E.1. Qualitatively, all of the models exhibit a transition from a highly convex regime at high loss to a non-convex regime at low loss, as demonstrated by the growth of the normalized length as well as the monotonic increase in the number of required "beads" to form a low-loss connection.

4.1 Polynomial Regression

We studied a 1-4-4-1 fully connected multilayer perceptron style architecture with sigmoid nonlinearities and RMSProp/ADAM optimization. For ease-of-analysis, we restricted the training and test data to be strictly contained in the interval $x \in [0, 1]$ and $f(x) \in [0, 1]$. The number of required beads, and thus the runtime of the algorithm, grew approximately as a power-law, as demonstrated in Table 1 Fig. 1. We also provide a visualization of a representative connecting path between two models of equivalent power in Appendix D.

The cubic regression task exhibits an interesting feature around $L_0 = .15$ in Table 1 Fig. 2, where the normalized length spikes, but the number of required beads remains low. Up until this point, the cubic model is strongly convex, so this first spike seems to indicate the onset of non-convex behavior and a concomitant radical change in the geometry of the loss surface for lower loss.

4.2 Convolutional Neural Networks

To test the algorithm on larger architectures, we ran it on the MNIST hand written digit recognition task as well as the CIFAR10 image recognition task, indicated in Table 1, Figs. 3 and 4. Again, the data exhibits strong qualitative similarity with the previous models: normalized length remains low until a threshold loss value, after which it grows approximately as a power law. Interestingly, the MNIST dataset exhibits very low normalized length, even for models nearly at the state of the art in classification power, in agreement with the folk-understanding that MNIST is highly convex and/or "easy". The CIFAR10 dataset, however, exhibits large non-convexity, even at the modest test accuracy of 80%.

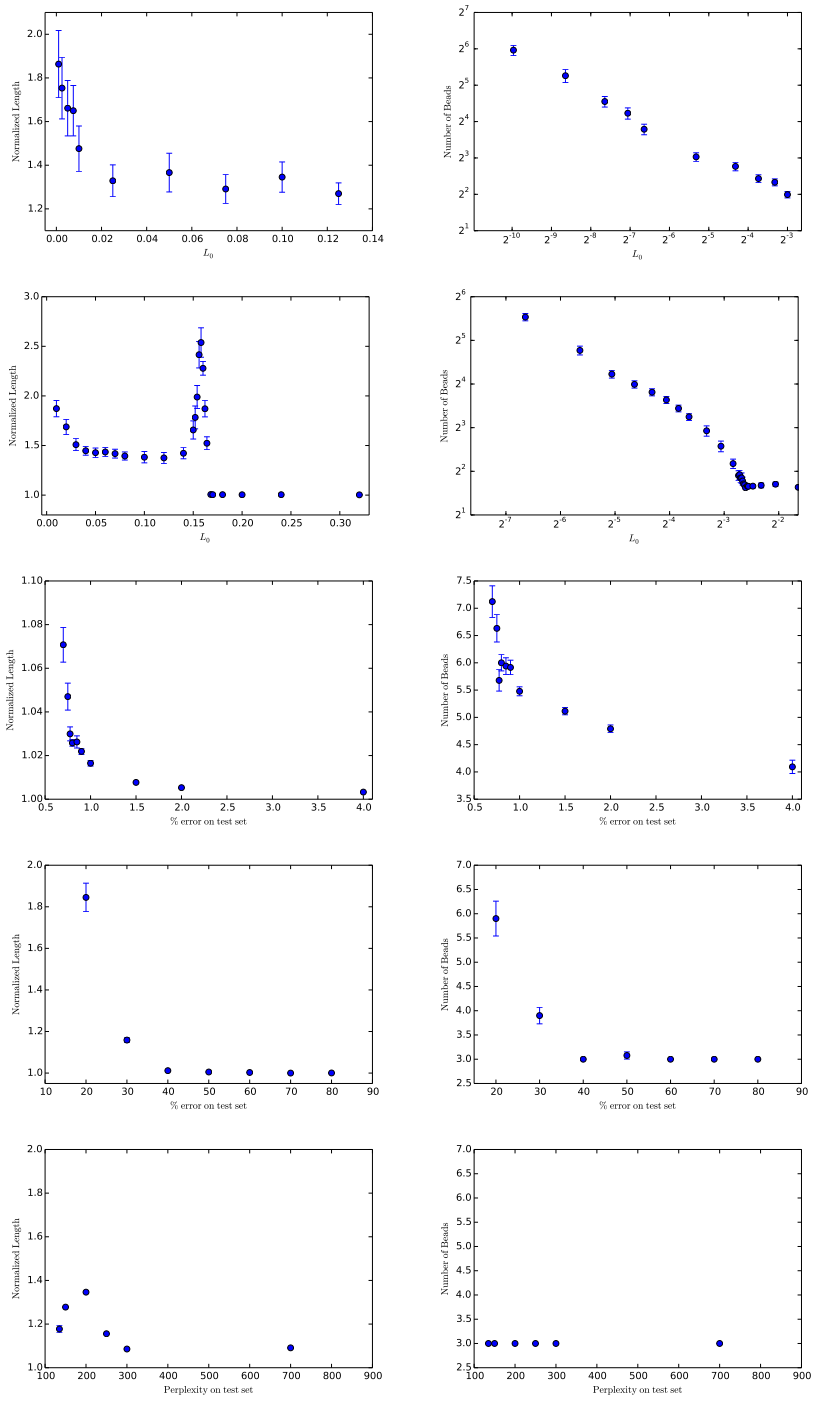

Figure 1: (Column a) Average normalized geodesic length and (Column b) average number of beads versus loss. (1) A quadratic regression task. (2) A cubic regression task. (3) A convnet for MNIST. (4) A convnet inspired by Krizhevsky for CIFAR10. (5) A RNN inspired by Zaremba for PTB next word prediction.

4.3 Recurrent Neural Networks

To gauge the generalizability of our algorithm, we also applied it to an LSTM architecture for solving the next word prediction task on the PTB dataset, depicted in Table 1 Fig. 5. Noteably, even for a

radically different architecture, loss function, and data set, the normalized lengths produced by the DSS algorithm recapitulate the same qualitative features seen in the above datasets—i.e., models can be easily connected at high perplexity, and the normalized length grows at lower and lower perplexity after a threshold value, indicating an onset of increased non-convexity of the loss surface.

5 DISCUSSION

We have addressed the problem of characterizing the loss surface of neural networks from the perspective of gradient descent algorithms. We explored two angles – topological and geometrical aspects – that build on top of each other.

On the one hand, we have presented new theoretical results that quantify the amount of uphill climbing that is required in order to progress to lower energy configurations in single layer ReLU networks, and proved that this amount converges to zero with overparametrization under mild conditions. On the other hand, we have introduced a dynamic programming algorithm that efficiently approximates geodesics within each level set, providing a tool that not only verifies the connectedness of level sets, but also estimates the geometric regularity of these sets. Thanks to this information, we can quantify how 'non-convex' an optimization problem is, and verify that the optimization of quintessential deep learning tasks – CIFAR-10 and MNIST classification using CNNs, and next word prediction using LSTMs – behaves in a nearly convex fashion up until they reach high accuracy levels.

That said, there is still a large number of limitations and open questions related to our framework. Amongst those, in the near future we shall concentrate on:

- *Extending Theorem 2.4 to the multilayer case*. We believe this is within reach, since the main analytic tool we use is that small changes in the parameters result in small changes in the covariance structure of the features. That remains the case in the multilayer case.
- *Empirical versus Oracle Risk*. A big limitation of our theory is that right now it does not inform us on the differences between optimizing the empirical risk versus the oracle risk. Understanding the impact of generalization error and stochastic gradient in the ability to do small uphill climbs is a major open line of research.
- *Influence of symmetry groups*. Our current model shows that under appropriate conditions, the presence of discrete symmetry groups does not prevent the loss from being connected, but does so at the expense of increasing the capacity. An important open question is whether one can significantly improve the asymptotic properties by relaxing connectedness to being connected up to discrete symmetry.
- *Improving numerics with Hyperplane method*. Our current numerical experiments employ a greedy (albeit faster) algorithm to discover connected components and estimate geodesics. We plan to perform experiments using the less greedy algorithm described in Appendix A.

ACKNOWLEDGMENTS

We would like to thank Mark Tygert for pointing out the reference to the ϵ-nets and Kolmogorov capacity. We would also like to thank Maithra Raghu and Jascha Sohl-Dickstein for enlightening discussions, as well as Yasaman Bahri for helpful feedback on an early version of the manuscript. CDF was supported by the NSF Graduate Research Fellowship under Grant DGE-1106400.

REFERENCES

Francis Bach. Convex relaxations of structured matrix factorizations. *arXiv preprint arXiv:1309.3117*, 2013.

Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Proc. AISTATS*, 2015.

Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems*, pp. 2933–2941, 2014.

David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

Chaitanya Ekanadham, Daniel Tranchina, and Eero P Simoncelli. Recovery of sparse translation-invariant signals with continuous basis pursuit. *IEEE transactions on signal processing*, 59(10): 4735–4744, 2011.

Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. Qualitatively characterizing neural network optimization problems. *arXiv preprint arXiv:1412.6544*, 2014.

Geoffrey Hinton, N Srivastava, and Kevin Swersky. Lecture 6a overview of mini–batch gradient descent. *Coursera Class*, 2012.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Kenji Kawaguchi. Deep learning without poor local minima. *arXiv preprint arXiv:1605.07110*, 2016.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent converges to minimizers. *University of California, Berkeley*, 1050:16, 2016.

Itay Safran and Ohad Shamir. On the quality of the initial basin in overspecified neural networks. *arXiv preprint arXiv:1511.04210*, 2015.

Levent Sagun, V Ugur Guney, Gerard Ben Arous, and Yann LeCun. Explorations on high dimensional landscapes. *arXiv preprint arXiv:1412.6615*, 2014.

Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

Ohad Shamir. Distribution-specific hardness of learning neural networks. *arXiv:1609.01037*, 2016.

Daniel Soudry and Yair Carmon. No bad local minima: Data independent training error guarantees for multilayer neural networks. *arXiv preprint arXiv:1605.08361*, 2016.

Gongguo Tang, Badri Narayan Bhaskar, Parikshit Shah, and Benjamin Recht. Compressed sensing off the grid. *IEEE Transactions on Information Theory*, 59(11):7465–7490, 2013.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

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. We did not use the algorithm presented in this section for our numerical studies.

Instead of training intermediate models via full SGD to a desired accuracy as in step 8 of the algorithm, intermediate models are be subject to a constraint that ensures they are "close" to the neighboring models on the string. Specifically, intermediate models are constrained to the unique hyperplane in weightspace equidistant from its two neighbors. This can be further modified by additional regularization terms to control the "springy-ness" of the string. These heuristics could be chosen to try to more faithfully sample the geodesic between two models.

In practice, for a given model on the string, θ_i, these two regularizations augment the standard loss by: $\tilde{F}(\theta) = F(\theta) + \zeta(\|\theta_{i-1} - \theta_i\| + \|\theta_{i+1} - \theta_i\|) + \kappa\| \frac{(\theta_{i-1}-\theta_{i+1})/2}{\|(\theta_{i-1}-\theta_{i+1})/2\|} \cdot \frac{(\theta_i-(\theta_{i-1}-\theta_{i+1})/2)}{\|(\theta_i-(\theta_{i-1}-\theta_{i+1})/2)\|} \|$. The ζ regularization term controls the "springy-ness" of the weightstring, and the κ regularization term controls how far off the hyperplane a new model can deviate.

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" in training. 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 $|\sigma_i - \sigma_j| < L_0$, 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, $L_0(t)$, of all models being trained approximately constant.

2. If the pairwise interpolated loss between σ_n and σ_{n+1} exceeds $L_0(t)$, insert a new model at the maximum of the interpolated loss (or halfway) between these two models.

3. Repeat steps (1) and (2) until all models (and interpolated errors) are below a threshold loss $L_0(t_{\text{final}}) := L_0$, or until a chosen failure condition (see 3.2).

B PROOFS

B.1 PROOF OF PROPOSITION 2.1

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

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

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 $\square$.

B.2 PROOF OF PROPOSITION 2.2

Let us first consider the case with $\kappa = 0$. We proceed by induction over the number of layers K. For $K = 1$, the loss $F(\theta)$ is convex. Let θ^A, θ^B be two arbitrary points in a level set Ω_λ. Thus $F(\theta^A) \leq \lambda$ and $F(\theta^B) \leq \lambda$. By definition of convexity, a linear path is sufficient in that case to connect θ^A and θ^B:

$$F((1-t)\theta^A + t\theta^B) \leq (1-t)F(\theta^A) + tF(\theta^B) \leq \lambda\,.$$

Suppose the result is true for $K-1$. Let $\theta^A = (W_1^A, \ldots, W_K^A)$ and $\theta^B = (W_1^B, \ldots, W_K^B)$ with $F(\theta^A) \leq \lambda$, $F(\theta^B) \leq \lambda$. Since $n_j \geq \min(n_1, n_K)$ for $j = 2 \ldots K-1$, we can find $k^* = \{1, K-1\}$ such that $n_{k^*} \geq \min(n_{k^*-1}, n_{k^*+1})$. For each $W_1, \ldots, W_K$, we denote $\tilde{W}_j = W_j$ for

$j \neq k^*, k^*-1$ and $\tilde{W}_{k^*} = W_{k^*-1}W_{k^*}$. By induction hypothesis, the loss expressed in terms of $\tilde{\theta} = (\tilde{W}_1, \ldots, \tilde{W}_{K-1})$ is connected between $\tilde{\theta}^A$ and $\tilde{\theta}^B$. Let $\tilde{W}_{k^*}(t)$ the corresponding linear path projected in the layer k^*. We need to produce a path in the variables $W_{k^*-1}(t)$, $W_{k^*}(t)$ such that:

i $W_{k^*-1}(0) = W^A_{k^*-1}, W_{k^*-1}(1) = W^B_{k^*-1}$,

ii $W_{k^*}(0) = W^A_{k^*}, W_{k^*}(1) = W^B_{k^*}$,

iii $W_{k^*}(t)W_{k^*-1}(t) = \tilde{W}_{k^*-1}(t)$ for $t \in (0,1)$.

We construct it as follows. Let

$$W_{k^*}(t) = tW^B_{k^*} + (1-t)W^A_{k^*} + t(1-t)V \ ,$$

$$W_{k^*-1}(t) = W_{k^*}(t)^\dagger \tilde{W}_{k^*-1}(t) \ ,$$

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 $t \in (0,1)$.

Finally, let us prove that the result is also true when $K = 2$ and $\kappa > 0$. We construct the path using the variational properties of atomic norms Bach (2013). When we pick the ridge regression regularization, the corresponding atomic norm is the nuclear norm:

$$\|X\|_* = \min_{UV^T = X} \frac{1}{2}(\|U\|^2 + \|V\|^2) \ .$$

The path is constructed by exploiting the convexity of the variational norm $\|X\|_*$. Let $\theta^A = (W^A_1, W^A_2)$ and $\theta^B = (W^B_1, W^B_2)$, and we define $\tilde{W} = W_1 W_2$. Since $\tilde{W}^{\{A,B\}} = W_1^{\{A,B\}} W_2^{\{A,B\}}$, it results that

$$\|\tilde{W}^{\{A,B\}}\|_* \leq \frac{1}{2}(\|W_1^{\{A,B\}}\|^2 + \|W_2^{\{A,B\}}\|^2) \ . \tag{11}$$

From (11) it results that the loss $F_o(W_1, W_2)$ can be minored by another loss expressed in terms of $\tilde{W}$ of the form

$$\mathbb{E}\{|Y - \tilde{W}X|^2\} + 2\kappa\|\tilde{W}\|_* \ ,$$

which is convex with respect to $\tilde{W}$. Thus a linear path in $\tilde{W}$ from $\tilde{W}^A$ to $\tilde{W}^B$ is guaranteed to be below $F_o(\theta^{\{A,B\}})$. Let us define

$$\forall\, t \ , \ W_1(t), W_2(t) = \arg \min_{UV^T = \tilde{W}(t)} (\|U\|^2 + \|V\|^2) \ .$$

One can verify that we can first consider a path $(\beta^A_1(s), \beta^A_2(s))$ from (W^A_1, W^A_2) to $(W_1(0), W_2(0)$ such that

$$\forall\, s\ \beta_1(s)\beta_2(s) = \tilde{W}^A \text{ and } \|\beta_1(s)\|^2 + \|\beta_2(s)\|^2 \text{ decreases} \ ,$$

and similarly for (W^B_1, W^B_2) to $(W_1(1), W_2(1)$. The path $(\beta^A_{\{1,2\}}(s), W_{\{1,2\}}(t), \beta^B_{\{1,2\}}(s))$ path satisfies (i-iii) by definition. We also verify that

$$\begin{aligned} \|W_1(t)\|^2 + \|W_2(t)\|^2 &= 2\|\tilde{W}(t)\|_* \\ &\leq 2(1-t)\|\tilde{W}(0)\|_* + 2t\|\tilde{W}(1)\|_* \\ &\leq (1-t)(\|W\|^2_1(0) + \|W\|^2_2(0)) + t(\|W\|^2_1(1) + \|W\|^2_2(1)) \ . \end{aligned}$$

which concludes the proof $\square$.

B.3 PROOF OF PROPOSITION 2.3

Let

$$A(w_1, w_2) = \{x \in \mathbb{R}^n;\, \langle x, w_1\rangle \geq 0\,,\, \langle x, w_2\rangle \geq 0\}\,.$$

By definition, we have

$$\begin{aligned}
\langle w_1, w_2\rangle_Z &= \mathbb{E}\{\max(0, \langle X, w_1\rangle)\max(0, \langle X, w_2\rangle)\} &(12)\\
&= \int_{A(w_1,w_2)} \langle x, w_1\rangle\langle x, w_2\rangle dP(x)\,, &(13)\\
&= \int_{Q(A(w_1,w_2))} \langle Q(x), w_1\rangle\langle Q(x), w_2\rangle (d\bar{P}(Q(x)))\,, &(14)
\end{aligned}$$

where Q is the orthogonal projection onto the space spanned by w_1 and w_2 and $d\bar{P}(x) = d\bar{P}(x_1, x_2)$ is the marginal density on that subspace. Since this projection does not interfere with the rest of the proof, we abuse notation by dropping the Q and still referring to $dP(x)$ as the probability density.

Now, let $r = \frac{1}{2}\|w_1 + w_2\| = \frac{1+\cos(\alpha)}{2}$ and $d = \frac{w_2 - w_1}{2}$. By construction we have

$$w_1 = rw_m - d\,,\; w_2 = rw_m + d\,,$$

and thus

$$\langle x, w_1\rangle\langle x, w_2\rangle = r^2|\langle x, w_m\rangle|^2 - |\langle x, d\rangle|^2\,. \tag{15}$$

By denoting $C(w_m) = \{x \in \mathbb{R}^n;\, \langle x, w_m\rangle \geq 0\}$, observe that $A(w_1, w_2) \subseteq C(w_m)$. Let us denote by $B = C(w_m) \setminus A(w_1, w_2)$ the disjoint complement. It results that

$$\begin{aligned}
\langle w_1, w_2\rangle_Z &= \int_{A(w_1,w_2)} \langle x, w_1\rangle\langle x, w_2\rangle dP(x)\\
&= \int_{C(w_m)} [r^2|\langle x, w_m\rangle|^2 - |\langle x, d\rangle|^2] dP(x) - r^2\int_B |\langle x, w_m\rangle|^2 dP(x) + \int_B |\langle x, d\rangle|^2 dP(x)\\
&= r^2\|w_m\|_Z^2 - \underbrace{r^2\int_B |\langle x, w_m\rangle|^2 dP(x)}_{E_1} - \underbrace{\int_{A(w_1,w_2)} |\langle x, d\rangle|^2 dP(x)}_{E_2}\,. &(16)
\end{aligned}$$

We conclude by bounding each error term E_1 and E_2 separately:

$$0 \leq E_1 \leq r^2|\sin(\alpha)|^2 \int_B \|x\|^2 dP(x) \leq r^2|\sin(\alpha)|^2 2\|\Sigma_X\|\,, \tag{17}$$

since every point in B by definition has angle greater than $\pi/2 - \alpha$ from w_m. Also,

$$0 \leq E_2 \leq \|d\|^2 \int_{A(w_1,w_2)} \|x\|^2 dP(x) \leq \frac{1-\cos(\alpha)}{2} 2\|\Sigma_X\| \tag{18}$$

by direct application of Cauchy-Schwartz. The proof is completed by plugging the bounds from (17) and (18) into (16) $\square$.

B.4 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 $(m-n)$-term 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

$$\overline{W} = [W^A; \tilde{W}_A] \ \in \mathbb{R}^{n \times 2m} \ .$$

and regression parameters

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

Clearly

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

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

$$\forall \, t \, , \overline{L}(t) = \mathbb{E}\{|Y - \eta(t)\overline{W}|^2\} + \kappa \|\eta(t)\|_1 \leq \max(\overline{L}(0), \overline{L}(1)) \ .$$

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 have that

$$\begin{aligned} F_o(\{\eta_1(t), \eta_2(t)\}) &= \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} + \kappa \|\eta_2(t)\|_1 \\ &\leq \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} + \kappa((1-t)\|\beta_1\|_1 + t\|\gamma_A\|_1) \\ &= \overline{L}(t) + \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} - \mathbb{E}\{|Y - (1-t)\beta_1 Z(W^A) - t\gamma_A Z(\tilde{W}_A)|^2\} \ . \end{aligned} \tag{19}$$

Finally, we verify that

$$\begin{aligned} &\left| \mathbb{E}\{|Y - \eta_2(t) Z(\eta_1(t))|^2\} - \mathbb{E}\{|Y - (1-t)\beta_1 Z(W^A) - t\gamma_A Z(\tilde{W}_A)|^2\} \right| \leq \\ &\leq 4\alpha \max(\mathbb{E}|Y|^2, \sqrt{\mathbb{E}|Y^2|}) \|\Sigma_X\| (\kappa^{-1/2} + \alpha \sqrt{\mathbb{E}|Y^2|} \kappa^{-1}) + o(\alpha^2) \ . \end{aligned} \tag{20}$$

Indeed, 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

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

which proves (20).

We have just constructed a path from θ^A to θ^B, in which all subpaths except (2) and (5) have energy maximized at the extrema due to convexity. For these two subpaths, (20) shows that it is sufficient to add the corresponding upper bound on the linear subpath. This concludes the proof. □

B.5 PROOF OF COROLLARY 2.5

Let us consider a generic first layer weight matrix $W \in \mathbb{R}^{n \times m}$. Without loss of generality, we can assume that $\|w_k\| = 1$ for all k, since increasing the norm of $\|w_k\|$ within the unit ball has no penalty in the loss, and we can compensate this scaling in the second layer thanks to the homogeneity of the half-rectification. Since this results in an attenuation of these second layer weights, they too are guaranteed not to increase the loss.

From Vershynin (2010) [Lemma 5.2] we verify that the covering number $\mathcal{N}(S^{n-1}, \epsilon)$ of the Euclidean unit sphere S^{n-1} satisfies

$$\mathcal{N}(S^{n-1}, \epsilon) \leq \left(1 + \frac{2}{\epsilon}\right)^n ,$$

which means that we can cover the unit sphere with an ϵ-net of size $\mathcal{N}(S^{n-1}, \epsilon)$.

Let $0 < \eta < n^{-1}(1 + n^{-1})^{-1}$, and let us pick, for each m, $\epsilon_m = m^{\frac{\eta - 1}{n}}$. Let us consider its corresponding ϵ-net of size

$$u_m = \mathcal{N}(S^{n-1}, \epsilon_m) \simeq \left(1 + \frac{2}{\epsilon_m}\right)^n \simeq m^{1-\eta} .$$

Since we have m vectors in the unit sphere, it results from the pigeonhole principle that at least one element of the net will be associated with at least $v_m = m u_m^{-1} \simeq m^\eta$ vectors; in other words, we are guaranteed to find amongst our weight vector W a collection Q_m of $v_m \simeq m^\eta$ vectors that are all at an angle at most $2\epsilon_m$ apart. Let us now apply Theorem 2.4 by picking $n = v_m$ and $\alpha = \epsilon_m$. We need to see that the terms involved in the bound all converge to 0 as $m \to \infty$.

The contribution of the oracle error $e(v_m) - e(m)$ goes to zero as $m \to \infty$ by the fact that $\lim_{m \to \infty} e(m)$ exists (it is a decreasing, positive sequence) and that $v_m \to \infty$.

Let us now verify that $\delta(m - v_m, \epsilon_m, m)$ also converges to zero. We are going to prune the first layer by removing one by one the vectors in Q_m. Removing one of these vectors at a time incurs in an error of the order of ϵ_m. Indeed, let w_k be one of such vectors and let β' be the solution of

$$\min_{\beta'} E(\beta') = \min_{\beta' = (\beta_f ; \beta_k) \in \mathbb{R}^k} \mathbb{E}\{|Y - \beta_f^T Z(W_{-k}) - \beta_k z(w_k)|^2\} + \kappa(\|\beta_f\|_1 + |\beta_k|) ,$$

where W_{-k} is a shorthand for the matrix containing the rest of the vectors that have not been discarded yet. Removing the vector w_k from the first layer increases the loss by a factor that is upper bounded by $E(\beta_p) - E(\beta)$, where

$$(\beta_p)_j = \begin{cases} \beta'_j & \text{for } j < k-1 , \\ \beta'_{k-1} + \beta'_k & \text{otherwise.} \end{cases} ,$$

since now β_p is a feasible solution for the pruned first layer.

Let us finally bound $E(\beta_p) - E(\beta)$.

Since $\angle(w_k, w_{k-1}) \leq \epsilon_m$, it results from Proposition 2.3 that

$$z(w_k) \stackrel{d}{=} z(w_{k-1}) + n ,$$

with $\mathbb{E}\{|n|^2\} \leq C\alpha^2$ for some constant C independent of m. By redefining $p_1 = Y - \beta_p^T Z(W_{-k}) - \frac{1}{2}n$ and $p_2 = \frac{1}{2}n$, we have

$$\begin{aligned} \mathbb{E}\{|Y - \beta_p^T Z(W_{-k})|^2\} - \mathbb{E}\{|Y - \beta'^T Z(W_{-k}) - \beta_k z(w_k)|^2\} &= \mathbb{E}\{|p_1 + p_2|^2\} - \mathbb{E}\{|p_1 - p_2|^2\} \\ &= 4\mathbb{E}\{|p_1 p_2|\} \\ &\leq \sqrt{\mathbb{E}\left\{\left|Y - \beta_p^T Z(W_{-k}) - \frac{1}{2}n\right|^2\right\}} \sqrt{\mathbb{E}\{|n|^2\}} \\ &\leq (C + \alpha)\alpha \simeq \epsilon_m , \end{aligned}$$

where C only depends on $\mathbb{E}\{|Y|^2\}$. We also verify that $\|\beta_p\|_1 \leq \|\beta'\|_1$.

It results that removing $|Q_m|$ of such vectors incurs an increase of the loss at most $|Q_m|\epsilon_m \simeq m^\eta m^{\frac{\eta-1}{n}} = m^{\eta + \frac{\eta-1}{n}}$. Since we picked η such that $\eta + \frac{\eta-1}{n} < 0$, this term converges to zero. The proof is finished. □

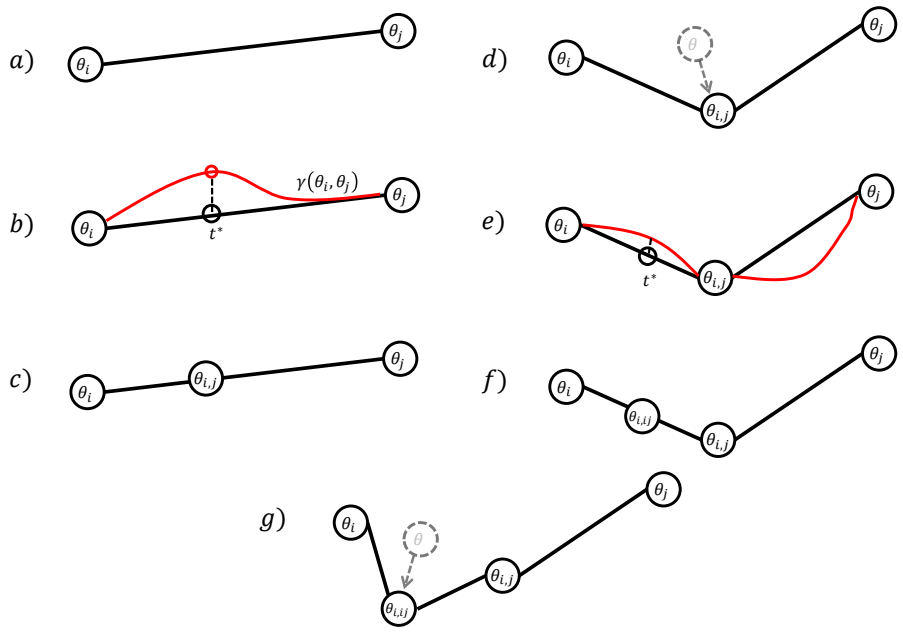

Figure 2: A cartoon of the algorithm. $a)$: The initial two models with approximately the same loss, L_0. $b)$: The interpolated loss curve, in red, and its global maximum, occuring at $t = t^*$. $c)$: The interpolated model $\Theta(\theta_i, \theta_j, t^*)$ is added and labeled $\theta_{i,j}$. $d)$: 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.

C Cartoon of Algorithm

Refer to Fig. 2.

D Visualization of Connection

Because the weight matrices are anywhere from high to extremely high dimensional, for the purposes of visualization we projected the models on the connecting path into a three dimensionsal subspace. Snapshots of the algorithm in progress for the quadratic regression task are indicated in Fig. **??**. This was done by vectorizing all of the weight matrices for all the beads for a given connecting path, and then performing principal component analysis to find the three highest weight projections for the collection of models that define the endpoints of segments for a connecting path—i.e., the θ_i discussed in the algorithm. We then projected the connecting string of models onto these three directions.

The color of the strings was chosen to be representative of the test loss under a log mapping, so that extremely high test loss mapped to red, whereas test loss near the threshold mapped to blue. An animation of the connecting path can be seen on our Github page.

Finally, projections onto pairs of principal components are indicated by the black curves.

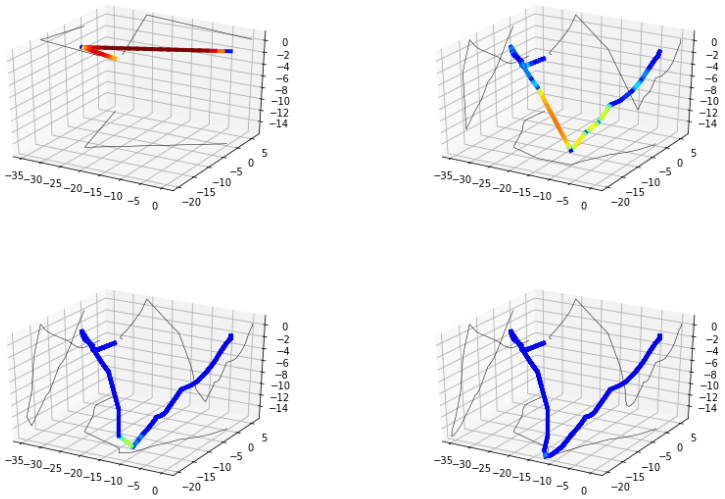

Figure 3: Snapshots of Dynamic String Sampling in action for the quadratic regression task. The string's coordinates are its projections onto the three most important principal axes of the fully converged string. (Top Left) One step into the algorithm, note the high loss between all of the vertices of the path. (Top Right) An intermediate step of the algorithm. Portions of the string have converged, but there are still regions with high interpolated loss. (Bottom Left) Near the end of the algorithm. Almost the entire string has converged to low loss. (Bottom Right) The algorithm has finished. A continuous path between the models has been found with low loss.

E A DISCONNECTION

E.1 A DISCONNECTION

As a sanity check for the algorithm, we also applied it to a problem for which we know that it is not possible to connect models of equivalent power by the arguments of section 2.3.1. The input data is 3 points in $\mathbb{R}^2$, and the task is to permute the datapoints, i.e. map $\{x_1, x_2, x_3\} \to \{x_2, x_3, x_1\}$. This map requires at least 12 parameters in general for the three linear maps which take $x_i \to x_j$ for $i, j \in \{\{1, 2\}, \{2, 3\}, \{3, 1\}\}$. Our archticture was a 2-3-2 fully connected neural network with a single relu nonlinearity after the hidden layer—a model which clearly has 12 free parameters by construction. The two models we tried to connect were a single model, θ, and a copy of θ with the first two neurons in the hidden layer permuted, $\tilde{\theta_\sigma}$. The algorithm fails to converge when initialized with these two models. We provide a visualization of the string of models produced by the algorithm in Fig. **??**.

In general, a persistent high interpolated loss between two neighboring beads on the string of models could arise from either a slowly converging, connected pair of models or from a truly disconnected pair of models. "Proving" a disconnection at the level of numerical experiments is intractable in general, but a collection of negative results—i.e., failures to converge—are highly suggestive of a true disconnection.

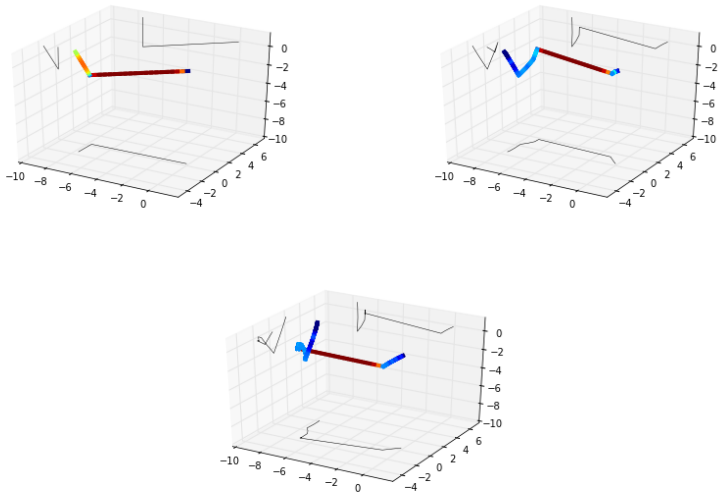

Figure 4: These three figures are projections of the components of the 12-dimensional weight matrices which comprise the models on the string produced by the DSS algorithm. The axes are the principal components of the weight matrices, and the colors indicate test error for the model. For more details on the figure generation, see Appendix D. (Left) The string of models after 1 step. Note the high error at all points except the middle and the endpoints. (Middle) An intermediate stage of the algorithm. Part of the string has converged, but a persistent high-error segment still exists. (Right) Even after running for many steps, the error persists, and the algorithm does not converge.