Proceedings of Machine Learning Research vol 99:1–24, 2019 32nd Annual Conference on Learning Theory

How do infinite width bounded norm networks look in function space?

Pedro Savarese SAVARESE@TTIC.EDU
Toyota Technological Institute at Chicago, Chicago IL, USA

Itay Evron EVRON.ITAY@GMAIL.COM
Department of Electrical Engineering, Technion, Haifa, Israel

Daniel Soudry DANIEL.SOUDRY@GMAIL.COM
Department of Electrical Engineering, Technion, Haifa, Israel

Nathan Srebro NATI@TTIC.EDU
Toyota Technological Institute at Chicago, Chicago IL, USA

Editors: Alina Beygelzimer and Daniel Hsu

Abstract

We consider the question of what functions can be captured by ReLU networks with an unbounded number of units (infinite width), but where the overall network Euclidean norm (sum of squares of all weights in the system, except for an unregularized bias term for each unit) is bounded; or equivalently what is the minimal norm required to approximate a given function. For functions $f : \mathbb{R} \to \mathbb{R}$ and a single hidden layer, we show that the minimal network norm for representing f is $\max(\int |f''(x)| \,\mathrm{d}x \,, \, |f'(-\infty) + f'(+\infty)| \,)$, and hence the minimal norm fit for a sample is given by a linear spline interpolation.

1. Introduction

Empirical and theoretical results suggest that neural network models used in practice are not constrained by their size (number of units, or number of weights) but perhaps achieve complexity control by controlling the magnitude of the weights (e.g. Bartlett, 1997; Neyshabur et al., 2014; Zhang et al., 2016), either explicitly or implicitly (Soudry et al., 2018; Gunasekar et al., 2018).

In fact, it is reasonable to think of networks as essentially having infinite size (having an infinite number of units), and controlled only through some norm of the weights. Using infinite (or unbounded) size networks we can approximate virtually any function, and so in training such a model we are essentially searching over the space of all functions.

Learning can be thought of as searching over the entire function space for a function with small *representation cost*, given by the minimal norm required to represent it in the chosen infinite size architecture. Understanding learning with infinite (or unbounded) size networks thus relies crucially on understanding this "representation cost", which is the actual inductive bias of learning.

Equivalently, we can think of this question as asking what functions can be represented, or approximated arbitrarily well, with an infinite-size bounded-norm network. There has been considerable work for over three decades on the question of what functions can be approximated by neural networks, establishing that any[1] function can be approximated by a large enough network, and studying the *size* (number of units) required to achieve good approximation (e.g. Hornik et al.,

[1] Any continuous or appropriately smooth function, depending on the precise model and topology.

1989; Cybenko, 1989; Barron, 1993; Pinkus, 1999). However, we are not aware of prior work establishing what *norm* is required for approximation, which, based on our current understanding of deep learning, is arguably the more important question (See Appendix B for a detailed comparison to Barron (1993)).

There is a significant difference in studying approximation in terms of size (number of units) vs norm: most smooth functions require unbounded size in order to be approximated arbitrarily well (after all, functions that can be represented with a bounded number of weights form a finite dimensional class and thus occupy only zero measure in the infinite dimensional function space). Therefore, in classical approximation results the size goes to infinity as the approximation error goes to zero, and results focus on how quickly the size goes to infinity, or on approximation up to finite error. In contrast, as we shall see here, broad classes of smooth functions can be approximated arbitrarily well, and even perfectly represented, with a bounded norm—the norm need not increase for the approximation to improve. Can we characterize this class of functions?

In this paper we consider infinite-width ReLU networks where the overall Euclidean norm (sum of squares of all weights in the system) is controlled. More specifically, we consider networks with a single hidden layer, consisting of an unbounded number of rectified linear units (ReLUs)—see Section 2 for a precise definition. Such two-layer infinite networks were studied in the context of generalization (Neyshabur et al., 2015) and optimization (Bach, 2017; Chizat and Bach, 2018) and are were shown by Neyshabur et al. (2014) to be equivalent to "convex neural nets" as studied by Bengio et al. (2006).

In Theorem 3.1 we show that for univariate functions $f : \mathbb{R} \to \mathbb{R}$, i.e. with a single real-valued input, optimizing a network's parameters while controlling the overall Euclidean norm is equivalent to fitting a function by controlling:

$$\max\left(\int \left|f''(x)\right| \mathrm{d}x, \left|f'(-\infty) + f'(+\infty)\right|\right).$$

We further show that fitting data while minimizing this complexity yields to linear spline interpolation. Interestingly, such linear splines are exactly the predictors recently studied by Belkin et al. (2018) as a model for understanding interpolation learning, though they fell short of connecting such models to neural networks. Our work thus closes the loop and establishes a concrete connection between their analysis and neural network learning.

We see then that even for univariate functions, Euclidean norm regularization on the weights already gives rise to very rich and natural induced bias in function space, that is not at all obvious and perhaps even surprising. We of course view this as a starting point for studying multivariate functions, and in Section 5 discuss how our techniques can be extended, and conjecture an answer related to the integral of the nuclear norm of the Hessian.

Our derivation of the induced complexity can be seen as an application of Green's function of the second derivative, and in Section 4 we elaborate on this view, and describe how fitting a two-layer network can be viewed as a method for solving a variational problem using Green's functions.

2. Infinite Width ReLU Networks

We consider 2-layer networks, with a single hidden layer consisting of an unbounded number of rectified linear units (ReLUs), defined by:

$$h_\theta(\mathbf{x}) = \sum_{i=1}^{k} w_i^{(2)} \left[\left\langle \mathbf{w}_i^{(1)}, \mathbf{x} \right\rangle + b_i^{(1)}\right]_+ + b^{(2)} \tag{1}$$

over $\mathbf{x} \in \mathbb{R}^d$, where $\mathbf{w}_i^{(1)}$ are the rows of $W^{(1)}$ and

$$\theta \in \Theta_2 = \Big\{\theta = \left(k, W^{(1)}, \mathbf{b}^{(1)}, \mathbf{w}^{(2)}, b^{(2)}\right) \;\Big|\; k \in \mathbb{N}, W^{(1)} \in \mathbb{R}^{k\times d}, \mathbf{b}^{(1)} \in \mathbb{R}^k, \mathbf{w}^{(2)} \in \mathbb{R}^k, \mathbf{b}^{(2)} \in \mathbb{R}\Big\} \tag{2}$$

We associate with each network the squared Euclidean norm of the non-bias weights[2]:

$$C(\theta) = \frac{1}{2}\left(\left\|\mathbf{w}^{(2)}\right\|_2^2 + \left\|W^{(1)}\right\|_F^2\right) = \frac{1}{2}\sum_{i=1}^{k}\left((w_i^{(2)})^2 + \left\|\mathbf{w}_i^{(1)}\right\|_2^2\right) \tag{3}$$

and consider the minimum norm required to implement a given function $f : \mathbb{R}^d \to \mathbb{R}$,

$$R\,(f) = \inf_{\theta\in\Theta_2} C(\theta) \;\text{ s.t. }\; h_\theta = f \tag{4}$$

Following[3] Neyshabur et al. (2014, Theorem 1), minimizing $C(\theta)$ is equivalent to constraining the norm of the weights $\mathbf{w}_i^{(1)}$ of each hidden unit and minimizing the ℓ_1 norm of the top layer. That is, we can express (4) as:

$$R(f) = \inf_{\theta\in\Theta_2} \left\|\mathbf{w}^{(2)}\right\|_1 \qquad \text{s.t. } h_\theta = f \;,\; \forall i : \left\|\mathbf{w}_i^{(1)}\right\|_2 = 1 \tag{5}$$

The above definitions of $R(f)$ require f to be exactly implementable by some finite-size ReLU network, and so $R(f)$ is finite only for piece-wise linear functions with a finite number of pieces. However, any continuous function can be approximated by increasing the number of units arbitrarily. Since we are not concerned with the number of units, only the norm, our central object of study is thus the norm required to capture a function as the number of units increases. This is captured by:

$$\overline{R}\,(f) = \lim_{\epsilon\to 0}\left(\inf_{\theta\in\Theta_2} C(\theta) \;\text{ s.t. }\; \|h_\theta - f\|_\infty \le \epsilon\right) \tag{6}$$

To understand $\overline{R}(f)$ observe that by (5), the sub-level set $\mathcal{F}_B = \{f | R(f) \le B\}$ is a scaling of the symmetric convex hull of ReLUs, plus arbitrary constant functions:

$$\mathcal{F}_B = B \cdot \text{conv}\left\{\mathbf{x} \mapsto \pm[\langle \mathbf{w}, \mathbf{x}\rangle + b]_+ \middle| \mathbf{w} \in \mathbb{S}^{d-1}, b \in \mathbb{R}\right\} + \{\mathbf{x} \mapsto b_0 | b_0 \in \mathbb{R}\}\;, \tag{7}$$

[2] Removing the output unit bias term $b^{(2)}$ will not change any of the definitions or results, since it can be simulated at no cost by a unit with infinitely large $b_i^{(1)}$, infinitesimally small $w_i^{(2)}$, and $\mathbf{w}_i^{(1)} = 0$. The unregularized bias terms $b_i^{(1)}$ in the hidden layer are important to our analysis, and removing them or regularizing them would substantially change the definitions and results.

[3] Neyshabur et al. do not consider an unregularized bias, although this does not change the essence of their arguments. In Appendix A we replicate their result, explicitly allowing for an unregularized bias.

where $\mathbb{S}^{d-1} = \{\mathbf{w} \in \mathbb{R}^d | \, \|\mathbf{w}\|_2 = 1\}$. The sub-level sets of $\overline{R}(f)$ are the closures $\overline{\mathcal{F}_B}$ of the above convex hulls, or in other words the set obtained by taking all expectations w.r.t all possible distributions, as opposed to only finite convex combinations (i.e. only expectations w.r.t. uniform discrete measures). We therefore have an infinite dimensional L_1 regularized problem:

$$\overline{R}(f) = \inf_{\alpha \in \mathcal{A}, c \in \mathbb{R}} \|\alpha\|_1 \quad \text{s.t. } \ h_{\alpha,c} = f \tag{8}$$

where $\mathcal{A}$ is the set of all signed measures on $\mathbb{S}^{d-1} \times \mathbb{R}$, $\|\alpha\|_1 = \int \mathrm{d}|\alpha|$, and

$$h_{\alpha,c} = \int_{\mathbb{S}^{d-1} \times \mathbb{R}} [\langle \mathbf{w}, \mathbf{x} \rangle + b]_+ \, \mathrm{d}\alpha(\mathbf{w}, b) + c. \tag{9}$$

Learning an unbounded width ReLU network h_θ by fitting some loss functional $L(\cdot)$ while controlling the norm $C(\theta)$ by minimizing

$$\min_{\theta \in \Theta_2} L(h_\theta) + \lambda \cdot C(\theta) \tag{10}$$

is thus equivalent to learning a function f while controlling $\overline{R}(f)$:

$$\min_{f:\mathbb{R} \to \mathbb{R}} L(f) + \lambda \cdot \overline{R}(f) \tag{11}$$

From (8), it is clear that $\overline{R}(f)$, and so also (11) are convex (although (10) is not!). Furthermore, for any lower semi-continuous[4] scalar loss function $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, if we consider the empirical loss $L(f) = \sum_{i=1}^{N} \ell(f(x_i); y_i)$ over N points, (10) has a minimizer where α is supported on at most N weight vectors, *i.e.,* uses at most $N+1$ units (Rosset et al., 2007). Thus, as long as the number of units is at least as large as the number of data points, we already have that (10) is equivalent to (11) (strictly speaking: any global minimum of (10) also minimizes (11)). Thus (11) not only tells us what happens in some infinite limit, but also precisely describes what we are minimizing with a finite, but sufficiently large, number of units.

Our goal is thus to calculate $\overline{R}(f)$, for any function $f : \mathbb{R}^d \to \mathbb{R}$, and in particular characterize when it is finite, i.e. understand which function can be approximated arbitrarily well with bounded norm but unbounded width ReLU networks.

3. One-dimensional Functions

Our main result is an exact and complete characterization of $\overline{R}(f)$ for univariate functions:

Theorem 3.1 *For any $f : \mathbb{R} \to \mathbb{R}$, we have:*

$$\overline{R}(f) = \max\left(\int_{-\infty}^{\infty} |f''(x)| \, \mathrm{d}x \, , \, |f'(-\infty) + f'(\infty)| \right) \leq \int_{-\infty}^{\infty} |f''(x)| \, \mathrm{d}x + 2 \inf_x |f'(x)|$$

where f'' is the weak (distributional) 2nd derivative and so the integral is well defined even if f is not differentiable, and is equal to the total variation of f'. We also denote $f'(\infty) = \lim_{x \to \infty} f'(x)$ and $f'(-\infty) = \lim_{x \to -\infty} f'(x)$, noting that if $\int_{-\infty}^{\infty} |f''(x)| \, \mathrm{d}x$ is finite, both limits must exist. The inequality and final expression are interpretable when f is differentiable (though this can be relaxed).

[4] Lower semi-continuous in its first argument. This is required only to ensure the minimum of (10) is attained.

In Theorem 3.1, its proof, and throughout, we work with distributions, rather than only functions, and as is common, we slightly abuse notation and use the same symbols to refer to both a function or measure and its associated distribution.

Proof In one dimension, we have that $w \in \mathbb{S}^{d-1} = \{\pm 1\}$ and it will be convenient for us to reparametrize the ReLUs as $[w(x-b)]_+$ instead of $[wx+b]_+$ (by transforming $(w,b) \mapsto (w, -wb)$, which does not change $\overline{R}(f)$). In this form, b exactly captures the threshold where a unit with parameters (w,b) activates ($w=+1$) or deactivates ($w=-1$). For any representation $f = h_{\alpha,c}$ of this form we have:

$$f(x) = \int_{\mathbb{R}} \left(\alpha(1,b)\,[x-b]_+ + \alpha(-1,b)\,[b-x]_+ \right) \mathrm{d}b + c \tag{12}$$

where we are treating the measures over b as distributions. Taking the derivative twice w.r.t. x:

$$f'(x) = \int_{\mathbb{R}} \left(\alpha(1,b) H(x-b) - \alpha(-1,b) H(b-x) \right) \mathrm{d}b \tag{13}$$

$$\begin{aligned} f''(x) &= \int_{\mathbb{R}} \left(\alpha(1,b) + \alpha(-1,b) \right) \delta_x(b)\, \mathrm{d}b \\ &= \alpha(1,x) + \alpha(-1,x) = \alpha_+(x) \end{aligned} \tag{14}$$

where $H(z)$ is the Heaviside step function ($H(z) = 1$ when $z > 0$ and zero otherwise), whose distributional derivative is the dirac distribution $\delta_a(z)$, which assigns point-mass to $z = a$, and we defined $\alpha_+(b) = \alpha(1,b) + \alpha(-1,b)$.

We see that the measure α that represents a function f is *almost* unique: the component α_+ corresponding to the total (signed) mass is unique and determined precisely the second derivative of f. The only flexibility is in shifting mass between the forward and backward sloping ReLUs with threshold b, *i.e.,* between $\alpha(1,b)$ and $\alpha(-1,b)$. We denote this component by $\alpha_-(b) = \alpha(1,b) - \alpha(-1,b)$, which together with α_+ defines α as $\alpha(w,b) = \frac{1}{2}(\alpha_+(b) + w \cdot \alpha_-(b))$. As the following calculation shows the component α_- only contributes an affine component to $f = h_{\alpha,c}$:

$$\begin{aligned} f(x) &= \int_{\mathbb{R}} \left(\alpha(1,b)\,[x-b]_+ + \alpha(-1,b)\,[b-x]_+ \right) \mathrm{d}b + c && (15) \\ &= \frac{1}{2} \int_{\mathbb{R}} \alpha_+(b)\,|x-b|\, \mathrm{d}b + \frac{1}{2} \int_{\mathbb{R}} \alpha_-(b)(x-b)\, \mathrm{d}b + c && (16) \\ &= \frac{1}{2} \int_{\mathbb{R}} f''(b)\,|x-b|\, \mathrm{d}b + \left(\frac{1}{2} \int_{\mathbb{R}} \alpha_-(b)\, \mathrm{d}b \right) x + \left(- \int_{\mathbb{R}} b\alpha_-(b)\, \mathrm{d}b + c \right) && (17) \end{aligned}$$

Our only constraint in choosing α_- is thus in getting the correct linear term in (17), as we can always adjust the constant term using the bias c without affecting $\overline{R}(f)$. To understand what this linear correction must be, we can use (13) to evaluate $f'(-\infty)$ and $f'(+\infty)$ (note that if $\int |f''|\, \mathrm{d}x = \int |\alpha_+(b)|\, \mathrm{d}b \leq \int |\mathrm{d}\alpha|$ is finite, then f' must converge at $\pm\infty$) and we get:

$$f'(-\infty) + f'(+\infty) = \int_{\mathbb{R}} (0 - \alpha(-1,b))\, \mathrm{d}b + \int_{\mathbb{R}} (\alpha(1,b) - 0)\, \mathrm{d}b = \int_{\mathbb{R}} \alpha_-(b)\, \mathrm{d}b \tag{18}$$

Any measure α_- that integrates as (18), in conjunction with $\alpha_+ = f''$ and an appropriate c, will yield $f = h_{\alpha,c}$ with

$$\|\alpha\|_1 = \frac{1}{2} \int_{\mathbb{R}} \left(\left| f''(b) + \alpha_-(b) \right| + \left| f''(b) - \alpha_-(b) \right| \right) \mathrm{d}b. \tag{19}$$

To minimize $\|\alpha\|_1$ we must therefore solve the following convex program:

$$\begin{aligned} \min_{\alpha_-} \frac{1}{2} & \int_{\mathbb{R}} \left(\left|f''(b) + \alpha_-(b)\right| + \left|f''(b) - \alpha_-(b)\right| \right) \mathrm{d}b \\ \text{s.t.} & \int_{\mathbb{R}} \alpha_-(b)\,\mathrm{d}b = f'(-\infty) + f'(\infty) \end{aligned} \tag{20}$$

Introducing the Lagrange multiplier $\lambda \in \mathbb{R}$, and setting the derivative of the Lagrangian $\mathcal{L}$ w.r.t. α_- to zero we have:

$$0 \in \frac{\partial \mathcal{L}}{\partial \alpha_-} = \frac{1}{2} \left(\operatorname{sign}\left(f'' + \alpha_-\right) - \operatorname{sign}\left(f'' - \alpha_-\right) + 2\lambda \right) \tag{21}$$

Consider the possible values λ might take:

$\lambda = 0$: We have $\operatorname{sign}\left(f'' + \alpha_-\right) = \operatorname{sign}\left(f'' - \alpha_-\right)$, and hence $|\alpha_-| \leq |f''|$ pointwise, and so from (19) we can calculate $\|\alpha\|_1 = \int_{\mathbb{R}} |f''(b)|\,\mathrm{d}b$. For the constraint in (20) to hold, we have:

$$\left|f'(-\infty) + f'(\infty)\right| = \left|\int_{\mathbb{R}} \alpha_-(b)\,\mathrm{d}b\right| \leq \int_{\mathbb{R}} |\alpha_-(b)|\,\mathrm{d}b \leq \int_{\mathbb{R}} \left|f''(b)\right| \mathrm{d}b. \tag{22}$$

$\lambda < 0$: For (21) to hold with $\lambda < 0$ we must have $f'' + \alpha_- \geq 0$ and $f'' - \alpha_- \leq 0$ pointwise, hence $\alpha_- \geq |f''|$ and from (19) and the constraint in (20): $\|\alpha\|_1 = \int_{\mathbb{R}} \alpha_-(b)\,\mathrm{d}b = f'(-\infty) + f'(\infty)$. This case is possible if we can make the constraint hold, *i.e.,* if and only if $f'(-\infty) + f'(\infty) \geq \int_{\mathbb{R}} |f''(b)|\,\mathrm{d}b$

$\lambda > 0$: Symmetric to $\lambda < 0$. We get $\|\alpha\|_1 = \int_{\mathbb{R}} (-\alpha_-)\,\mathrm{d}b = -(f'(-\infty) + f'(\infty))$ and this happens when $f'(-\infty) + f'(\infty) \leq -\int_{\mathbb{R}} |f''(b)|\,\mathrm{d}b$.

Combining the cases above, we have $\|\alpha\|_1 = \max(\int |f''|\,\mathrm{d}b, |f'(-\infty) + f'(+\infty)|)$. To get the inequality in the Theorem statement, note that for any x:

$$\begin{aligned} \left|f'(-\infty) + f'(+\infty)\right| &= \left| \left(f'(x) - \int_{-\infty}^{x} f''(b)\,\mathrm{d}b \right) + \left(f'(x) + \int_{x}^{\infty} f''(b)\,\mathrm{d}b \right) \right| \\ &\leq 2\left|f'(x)\right| + \int_{\infty}^{\infty} \left|f''\right| \mathrm{d}b. \end{aligned} \tag{23}$$

■

Corollary 3.2 *For any loss function $L(f)$ over $f : \mathbb{R} \to \mathbb{R}$, fitting a regularized infinite width ReLU network by minimizing* $\arg\min_{\theta \in \Theta_2} L(h_\theta) + \lambda \cdot C(\theta)$ *is equivalent to :*

$$\operatorname*{arg\,min}_{f:\mathbb{R}\to\mathbb{R}} L(f) + \lambda \cdot \max\left(\int_{-\infty}^{\infty} \left|f''(x)\right| \mathrm{d}x\,,\; \left|f'(-\infty) + f'(\infty)\right| \right) \tag{24}$$

Corollary 3.2 shows the learning with norm-regularized ReLU networks is essentially equivalent to learning by minimizing the total variation of the derivative. How does fitting data while minimizing this total variation look like? What kind of functions would we get? Consider first perfectly fitting (interpolating) a finite set of points $S = (x_n, y_n)_{n=1}^N,\ x_n, y_n \in \mathbb{R}$, given by:

$$\operatorname*{arg\,min}_{h_\theta\,:\,\theta\in\Theta_2} C(\theta)\ s.t.\ \left(\forall_{n\in[N]}, h_\theta(x_n) = y_n\right) \quad = \quad \operatorname*{arg\,min}_{f:\mathbb{R}\to\mathbb{R}} \overline{R}(f)\ s.t.\ \left(\forall_{n\in[N]}, f(x_n) = y_n\right) \tag{25}$$

Theorem 3.3 *For any dataset $S = (x_n, y_n)_{n=1}^N,\ x_n, y_n \in \mathbb{R}$, where w.l.o.g. $x_1 < x_2 < \cdots < x_N$, (25) is minimized by the linear spline interpolation:*

$$\tilde{f}(x) = \begin{cases} y_1 + l_0(x - x_1) & x \le x_1 \\ y_n + l_n(x - x_n) & x_n \le x \le x_{n+1} \quad \textit{for some } n \in [N] \\ y_N + l_N(x - x_N) & x \ge x_N \end{cases} \tag{26}$$

where $l_n = \frac{y_{n+1} - y_n}{x_{n+1} - x_n}$ for $n \in \{1, \cdots, N-1\}$, and l_0, l_N are chosen so as to minimize

$$\max\left(\sum_{n=0}^{N-1} |l_{n+1} - l_n|\,, |l_0 + l_N|\right) \tag{27}$$

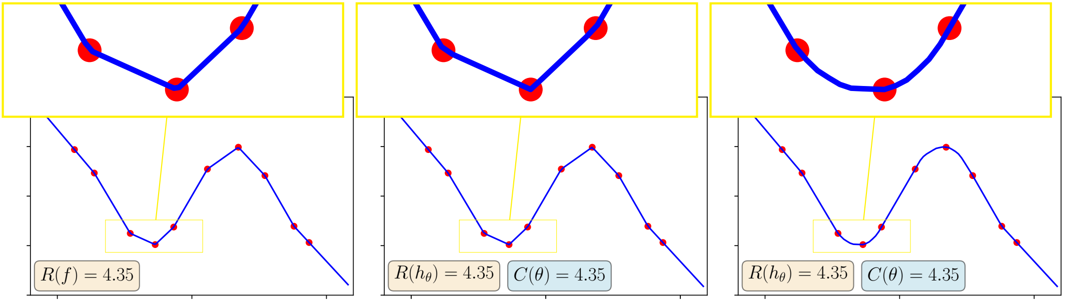
Proof First, verify that $\int_{\mathbb{R}} \left|\tilde{f}''(x)\right| \mathrm{d}x = \sum_{n=0}^{N-1} |l_{n+1} - l_n|$ and so $\overline{R}(\tilde{f})$ is given by (27). We now need to show that (27) is a lower bound on the value in (25). Following Rosset et al. (2007), and when we consider (25) as minimization w.r.t. the measure α and $c \in \mathbb{R}$, it is always minimized by $f^* = h_{\alpha,c}$ where α is discrete with support of size at most $N + 1$, corresponding to a piece-wise linear f^* with at most $N + 1$ breakpoints. But instead of working with piece-wise linear functions, we will rely on the fact that by smoothing f^* we can always approach it with a sequence of twice continuously differentiable functions[5] $f_r \to f^*$, with $\overline{R}(f_r) \to \overline{R}(f^*)$. It is thus sufficient to prove that $\liminf \overline{R}(f_r) \ge \overline{R}(\tilde{f})$. For f_r s.t. $\|f^* - f_r\| < \epsilon$, since f_r is continuously differentiable, for all $n = 1, \ldots, N-1$, there exists a midpoint $x_n \le z_n \le x_{n+1}$ such that $|f_r'(z_n) - l_n| \le 2\epsilon/\delta$ where $\delta = \min_n (x_{n+1} - x_n)$. Recalling that $\overline{R}(\tilde{f})$ is the minimum of (27) over l_0, l_N, and plugging in $l_0 = f_r'(-\infty)$ and $l_N = f_r'(+\infty)$, we have that $\overline{R}(f_r) > \overline{R}(\tilde{f}) - 4N\epsilon/\delta$. Taking $r \to \infty$ and so $\epsilon \to 0$, we have $\overline{R}(f^*) = \lim_{r\to\infty} \overline{R}(f_r) \ge \overline{R}(\tilde{f})$, which establishes that $\tilde{f}$ minimizes (25). ■

Theorem 3.1 guarantees that the linear spline interpolation is *a* global minimum of (25), but it will in general not be unique, as demonstrated in Figure 1, and networks learned in practice might implement much "smoother" functions. The same type of solutions will also be obtained when minimizing *any* loss over a finite sample:

Corollary 3.4 (Optimality of Linear Interpolation) *For any dataset $S = (x_n, y_n)_{n=1}^N,\ x_n, y_n \in \mathbb{R}$, and any lower semi-continuous loss $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R} \cup \{\infty\}$ and any $\lambda > 0$, consider:*

$$\operatorname*{arg\,min}_{h_\theta:\theta\in\Theta_2} \sum_{n=1}^{N} \ell(h_\theta(x_n), y_n) + \lambda \cdot C(\theta). \tag{28}$$

[5] Take f_r to be a convolution of f^* with $\exp(-rx^2)$, so that $\|f^* - f\|_\infty \to 0$ and $\|f_*' - f_r'\|_1 \to 0$.

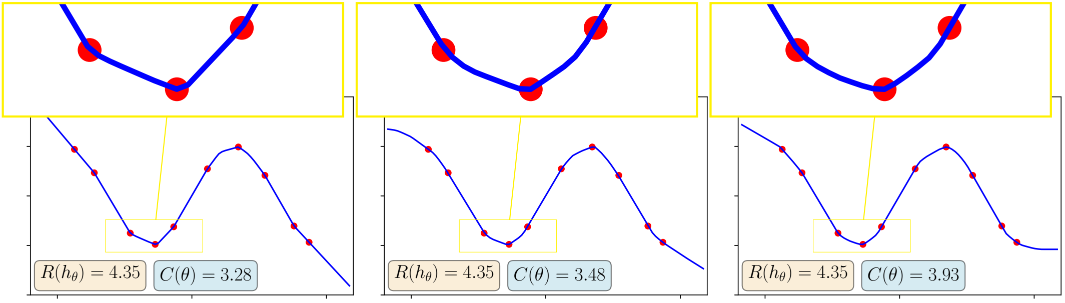


Linear spline

2 layers, 20 units

2 layers, 100 units

Figure 1: Linear interpolation (**left**) and two trained ReLU networks with 1 hidden layer consisting of 20 (**middle**) and 100 (**right**) units respectively, optimized to perfectly fit a set of 10 points, for which the minimum cost of perfect fitting is $\overline{R}(f^*) = 4.35$. Training was done by minimizing the squared loss with a small regularization of $\lambda = 10^{-5}$. All three functions achieve the optimal cost $\overline{R}(\cdot)$ in function space, and both networks yield optimal cost in parameter space $C(\theta)$. The two networks arrived at different global minima in function space, with the same value of $\overline{R}(f)$. For example, in the area highlighted, changing the derivative gradually instead of abruptly does not effect its total variation, and so also yields an optimal solution.



3 layers, 100 units

4 layers, 100 units

5 layers, 100 units

Figure 2: Deeper ReLU networks with 2 (**left**), 3 (**middle**) and 4 (**right**) hidden layers, each with 100 hidden units, trained on the same set of points as the 1-hidden layer networks on Figure 1, also with $\lambda = 10^{-5}$.

Then (28) will always have a global minimum which is a piece-wise linear function with at most $N+1$ pieces, with breakpoints at the data points $x_1, \ldots, x_N$.

Proof A minimizer h_{θ^*} of (28) is also a minimizer of (25) with the alternative labels $y_n = h_{\theta^*}(x_n)$. ■

4. An Interpertation in Terms of Green's Functions

The key component in the proof of Theorem 3.1 is in writing a two-layer network as a convolution with ReLU functions as in (12), and taking the second derivative of this convolution (in Equation (14)), noticing that the ReLU is a Green's function for the second derivative. Fitting a ReLU network can therefore be seen as the reverse of using Green's function to solve a differential equation: we know $f : \mathbb{R} \to \mathbb{R}$ and would like to calculate it's 2nd derivative $u = f''$. We can do this by fitting an infinite width ReLU network and then reading off the second derivative from the resulting weights.

In terms of the representation (12), we get the second derivative directly from the measure α, as $u(x) = \alpha_+(x) = \alpha(1, x) + \alpha(-1, x)$. For the more standard integral representation (9), we have $u(x) = \alpha(1, x) + \alpha(-1, -x)$. Fitting an actual network with discrete units and weights on each unit, as in (1), corresponds to an approximation of the form:

$$u \approx \sum_i w_i^{(2)} \left| w_i^{(1)} \right| \delta_{b/w_i^{(1)}} \tag{29}$$

where to get a smoother approximation we might want to replace the delta function with a smoother bump.

Ignoring for the moment the linear term, we saw in Corollary 3.2 that fitting a two-layer ReLU network while controlling the norm of the weights corresponds essentially to the variational problem:

$$\min_f L(f) + \int_{\mathbb{R}} \left| f''(x) \right| \mathrm{d}x. \tag{30}$$

An interpretation of how this is done using the ReLU network is as follows: we first introduce an auxiliary function variable u, and rewrite (30) as:

$$\min_{f,u} L(f) + \int_{\mathbb{R}} |u(x)| \, \mathrm{d}x \quad \text{s.t. } u = f''. \tag{31}$$

We then use the fact that the ReLU is a Green's function for the second derivative to write

$$f(x) = \int_{\mathbb{R}} u(b)[x - b]_+ \, \mathrm{d}b + ax + c \tag{32}$$

where $ax + c$, for some $a, x \in \mathbb{R}$, is the affine term left undetermined by the second derivative. Plugging (32) back into (31) we get:

$$\min_u L\left(x \mapsto \int_{\mathbb{R}} u(b)[x - b]_+ \, \mathrm{d}b + ax + c\right) + \int_{\mathbb{R}} |u(b)| \, \mathrm{d}b \tag{33}$$

which amounts to fitting an infinite ReLU network with L_1 regularization on the top layer, which in turn, using the equivalence between (5) and (4) , is equivalent to ℓ_2-regularization on both layers. The additional term $|f'(-\infty), f'(+\infty)|$ in Theorem 3.1 comes from the fact that unlike in the derivation above, we are also constraining the linear term.

5. Approximating Higher Dimensional Functions

Next, we discuss how our results may generalize to higher dimensions, i.e. for networks with multiple input units where the input $\mathbf{x}$ is in $\mathbb{R}^d$, and help us in characterizing $\overline{R}(f)$ for general $f : \mathbb{R}^d \to \mathbb{R}$. First, it easy to see that we can extend our Green's function view for higher dimensional inputs. For any representation $f = h_{\alpha,c}$ as in (9) we have:

$$\frac{\partial}{\partial x_i} f(\mathbf{x}) = \int_{\mathbb{S}^{d-1} \times \mathbb{R}} \mathrm{d}\mathbf{w} \, \mathrm{d}b \left(w_i \alpha(\mathbf{w}, b) \, H\left(\mathbf{w}^\top \mathbf{x} - b\right)\right) \tag{34}$$

where we recall H is the Heaviside step function, and

$$\begin{aligned}\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j}f\left(\mathbf{x}\right) &= \int_{\mathbb{S}^{d-1}\times\mathbb{R}} \mathrm{d}\mathbf{w}\,\mathrm{d}b\left(w_j w_i \alpha\left(\mathbf{w},b\right)\delta\left(\mathbf{w}^\top\mathbf{x}-b\right)\right)\\ &= \int_{\mathbb{S}^{d-1}} \mathrm{d}\mathbf{w}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)w_j w_i\,.\end{aligned}$$

Therefore, the Hessian can be written as

$$\nabla^2 f\left(\mathbf{x}\right) = \int_{\mathbb{S}^{d-1}} \mathrm{d}\mathbf{w}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)\mathbf{w}\mathbf{w}^\top \tag{35}$$

and the Laplacian as

$$\begin{aligned}\triangle f\left(\mathbf{x}\right) &= \mathrm{Tr}\left[\nabla^2 f\left(\mathbf{x}\right)\right] = \sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}f\left(\mathbf{x}\right)\\ &= \int_{\mathbb{S}^{d-1}} \mathrm{d}\mathbf{w}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)\left\|\mathbf{w}\right\|^2 = \int_{\mathbb{S}^{d-1}} \mathrm{d}\mathbf{w}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)\,. \end{aligned}\tag{36}$$

What remains is to minimize $\left\|\alpha\right\|_1$ under this constraint. As an indicative result, we can calculate $\left\|\alpha\right\|_1$ for the special case where it is non-negative, and so $f = f_{\alpha,c}$ is convex and the Hessian is p.s.d.:

Claim 5.1 *If $\alpha\left(\mathbf{w},b\right)\geq 0$, $\forall\mathbf{w}\in\mathbb{S}^{d-1}$ and $\forall b\in\mathbb{R}$, then*

$$\left\|\alpha\right\|_1 = \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\left\|\mathbf{x}\right\|_2\leq r}\mathrm{Tr}\left[\nabla^2 f\left(\mathbf{x}\right)\right]d\mathbf{x} = \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\oiint_{\left\|\mathbf{x}\right\|_2=r}\nabla f(\mathbf{x})^\top\,\mathrm{d}\hat{\boldsymbol{n}}(\mathbf{x})$$

where $\mathbb{V}^d$ is the volume of a unit radius d-ball, and $\mathrm{d}\hat{\boldsymbol{n}}(\mathbf{x})$ is the outward pointing unit normal.

Proof The right equality is a direct consequence of the divergence theorem. It remains to prove the left equality. From (36) we have

$$\begin{aligned}&\lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\left\|\mathbf{x}\right\|_2\leq r}\mathrm{Tr}\left[\nabla^2_{\mathbf{x}} f\left(\mathbf{x}\right)\right]\mathrm{d}\mathbf{x} = \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\left\|\mathbf{x}\right\|_2\leq r}\mathrm{d}\mathbf{x}\int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)\\
&= \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\left[\int_{\left\|\mathbf{x}\right\|_2\leq r}\mathrm{d}\mathbf{x}\alpha\left(\mathbf{w},\mathbf{w}^\top\mathbf{x}\right)\right]\\
&\overset{(1)}{=} \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\left[\int_{-r}^{r}\mathrm{d}x_1\alpha\left(\mathbf{w},x_1\right)\int_{\mathcal{A}_r(x_1)}\prod_{i=2}^{d}\mathrm{d}x_i\right]\\
&\overset{(2)}{=} \lim_{r\to\infty}\frac{1}{r^{d-1}\mathbb{V}^{d-1}}\int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\left[\int_{\mathbb{R}}\mathrm{d}x_1\alpha\left(\mathbf{w},x_1\right)\mathcal{I}\left[x_1\in\left[-r,r\right]\right]\mathbb{V}^{d-1}\left(r^2-x_1^2\right)^{\frac{d-1}{2}}\right]\\
&\overset{(3)}{=} \int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\left[\int_{\mathbb{R}}\mathrm{d}x_1\alpha\left(\mathbf{w},x_1\right)\lim_{r\to\infty}\left(\mathcal{I}\left[x_1\in\left[-r,r\right]\right]\left[1-\left(x_1/r\right)^2\right]^{\frac{d-1}{2}}\right)\right]\\
&\overset{(4)}{=} \int_{\mathbb{S}^{d-1}}\mathrm{d}\mathbf{w}\int_{\mathbb{R}}\mathrm{d}b\alpha\left(\mathbf{w},b\right)\overset{(5)}{=}\left\|\alpha\right\|_1\end{aligned}$$

where in (1) we assume, without loss of generality, that $\mathbf{w}^\top = (1, 0, \ldots, 0)$ inside the square brackets and define the following $d-1$ dimensional ball

$$\mathcal{A}_r\left(z\right) \triangleq \left\{ \mathbf{x} \in \mathbb{R}^d | \left\|\mathbf{x}\right\|_2 \leq r, x_1 = z \right\} \,, \tag{37}$$

in (2) we define $\mathcal{I}$ as an indicator function and calculate the surface area of $\mathcal{A}_r\left(z\right)$, in (3) we switch the order of the limit and integration using the bounded convergence theorem, in (4) we denote $x_1 = b$, and in (5) we use our assumption that $\alpha\left(\mathbf{w}, b\right) \geq 0$. ■

We do not expect that in general α would be non-negative, nor that the Hessian would be p.s.d. Recall that in the one dimensional case, $\overline{R}(f)$ is related to the integral of the *absolute value* of f''. Therefore, in order to generalize Claim 5.1 to mixed sign eigenvalues, a naive conjecture would be that $\overline{R}(f)$ is given by the integral of the *nuclear norm* of the Hessian, up to a linear function accounting for the boundary conditions as in the one dimensional case. That is, the sum of the absolute values of the eigenvalues of the Hessian, as opposed to the Laplacian which is the sum of signed eigenvalues.

Unfortunately, such a conjecture cannot be accurate. The issue is that, for some functions with vanishing Hessian, any norm of the Hessian, normalized as in Claim 5.1, would be equal to zero. For example,

Claim 5.2 *For the function*

$$h(\mathbf{x}) = \int_{\mathbb{S}^{d-1}} d\mathbf{w} \left(\left[\mathbf{w}^\top \mathbf{x} + 1\right]_+ - 2\left[\mathbf{w}^\top \mathbf{x}\right]_+ + \left[\mathbf{w}^\top \mathbf{x} - 1\right]_+ \right) \,, \tag{38}$$

for any norm, we have

$$\lim_{r\to\infty} \frac{1}{r^{d-1}} \int_{\|\mathbf{x}\| \leq r} d\mathbf{x} \left\| \nabla^2 h(\mathbf{x}) \right\| = 0 \,. \tag{39}$$

The proof is given in appendix D. Importantly, this function is written as the output of an infinite neural network with a finite nonzero norm $\left\|\alpha\right\|_1 = 4\int_{\mathbb{S}^{d-1}} d\mathbf{w} > 0$. Moreover, we can perfectly fit any finite set of points $\left\{\mathbf{x}_n, y_n\right\}_{n=1}^N$ using a linear combination of scaled and shifted $h(\mathbf{x})$ (each $h(\mathbf{x})$ has a radial "bump" shape, which vanishes at infinity). Therefore, the true expression for $\bar{R}(f)$, must be different from a normalized integral of some norm of the Hessian (as in eq. 39) — otherwise we would get $\bar{R}(f) = 0$ for such a fit, in contradiction to that $\left\|\alpha\right\|_1 > 0$ in this case. We leave it to future work to find if there is some function space cost $\bar{R}(f)$ corresponding to $\left\|\alpha\right\|_1$ minimization.

6. Discussion

As we are realizing that (explicit or implicit) regularization plays a key role in deep learning, it is crucial to understand how simple norm control on the weights in parameter space induces rich complexity control in function space. In a sense, for infinite size networks, the *only* role of the architecture is to give rise to and shape this rich complexity control. Recent work studied this question in *linear* neural networks, and showed how in regularizing the weights in a convolutional network yields rich complexity control inducing sparsity in the frequency domain (Gunasekar et al., 2018). Here we go beyond linear networks and study infinite ReLU networks.

Much in the same as linear convolutional neural networks can represent any *linear* function, and the architecture's only role is to induce complexity control in that space, infinite width ReLU networks can represent any *continuous* function, and the role of the architecture, in our view, is to induce complexity control over function space. We see that indeed, even for univariate functions, the architecture already induces a natural complexity control that is not obvious nor explicit. Furthermore, we are particularly excited that this complexity control exactly matches the learning rules studied in recent work on interpolation learning (Belkin et al., 2018), and provides a concrete connection of that work to deep learning. We are eager to find out whether this connection carries also to the multivariate case.

We also hope that our study will reinvigorate approximation theory work on neural networks, studying approximation by networks of bounded *norm* rather than a bounded number of units.

Similarly, we would argue that the study of the importance of depth in neural networks should focus not on gaps in the *size* (number of units) required to fit a function, but whether deeper networks allow lower *norm* representation, and how depth changes the inductive bias induced by norm control over the weights. Gunasekar et al. (2018) showed that for linear convolutional networks, the depth meaningfully changes the induced inductive bias, with depth L networks corresponding to an $\ell_{2/L}$ bridge penalty, but in fully connected linear network the depth has no effect. In Appendix C we study an architecture with infinitely many deep parallel ReLU networks, and show that also for such an architecture depth L gives rise to similar $\ell_{2/L}$ sparsity-inducing bridge penalties, replacing the ℓ_1 penalty for two-layer networks.

It would be interesting to study if and how depth changes the induced complexity control in infinitely wide fully connected L-layer ReLU networks. We can naturally extend our setup, letting $C_L(\theta)$ refer to the sum of the square of all weights in the system, and defining $\overline{R}_L$ analogously to (6), where the infimum is over all depth L ReLU networks, with any number of units per layer (similar to $\gamma^L_{2,2}(f)$ as defined by Neyshabur et al. (2015)). In an anecdotal empirical example presented in Figure 2, depth does not appear to change the inductive bias, as with any depth network we recover a function minimizing $\overline{R}(f)$ as calculated in Theorem 3.1. Another natural extension is to consider multiple output units—for two layer linear networks this corresponds to Frobenious norm control on a matrix factorization which induces nuclear norm regularization on the linear mapping from input to output (Fazel et al., 2001; Srebro et al., 2004). How does this play out for $f : \mathbb{R}^d \to \mathbb{R}^k$ in function space with ReLU networks?

Acknowledgements

We are in debt to Charlie Smart (University of Chicago) for pointing out the connection to Green's functions, and would also like to thank Jason Lee (UCS), Holden Lee (Princeton) and Arturs Backurs (TTIC) for helpful discussions. PS and NS were partially supported by NSF awards 1546500 and 1764032.

References

Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 2017.

Andrew R Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information theory*, 39(3):930–945, 1993.

Andrew R Barron. Approximation and estimation bounds for artificial neural networks. *Machine learning*, 14, 1994.

Peter L Bartlett. For valid generalization the size of the weights is more important than the size of the network. In *Advances in neural information processing systems*, 1997.

Mikhail Belkin, Daniel Hsu, and Partha Mitra. Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate. *Advances in neural information processing systems*, 32, 2018.

Yoshua Bengio, Nicolas L. Roux, Pascal Vincent, Olivier Delalleau, and Patrice Marcotte. Convex neural networks. In *Advances in Neural Information Processing Systems 18*. MIT Press, 2006.

Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. Wide & deep learning for recommender systems. *CoRR*, abs/1606.07792, 2016. URL http://arxiv.org/abs/1606.07792.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. *arXiv preprint arXiv:1805.09545*, 2018.

George Cybenko. Approximations by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2, 1989.

M. Fazel, H. Hindi, and S. P. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *Proceedings of the 2001 American Control Conference*, 2001.

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Implicit bias of gradient descent on linear convolutional networks. In *NeurIPS*, 2018.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2, 1989.

Holden Lee, Rong Ge, Tengyu Ma, Andrej Risteski, and Sanjeev Arora. On the ability of neural nets to express distributions. In *Conference on Learning Theory*, pages 1271–1296, 2017.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv preprint arXiv:1412.6614*, 2014.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, 2015.

Fedor Petrov. Proving an infinite norm minimization problem has finite support (non-convex p-norms). MathOverflow, 2019. URL https://mathoverflow.net/q/321004.

Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta numerica*, 8, 1999.

Saharon Rosset, Grzegorz Swirszcz, Nathan Srebro, and Ji Zhu. 1 regularization in infinite dimensional feature spaces. In *International Conference on Computational Learning Theory*. Springer, 2007.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *CoRR*, 2017.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *JMLR*, 2018.

Nathan Srebro, Jason D. M. Rennie, and Tommi S. Jaakkola. Maximum-margin matrix factorization. In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, NIPS, 2004.

Ryan J Tibshirani et al. The lasso problem and uniqueness. *Electronic Journal of Statistics*, 2013.

Colin Wei, Jason D Lee, Qiang Liu, and Tengyu Ma. On the margin theory of feedforward neural networks. *arXiv preprint arXiv:1810.05369*, 2018.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Appendix

Appendix A. Equivalence of overall ℓ_2 control to ℓ_1 control on the output layer

In this Appendix we formally show that regularizing the overall ℓ_2 norm on the weights in two layers (i.e. $C(\theta)$, the sum of squares of all weights in the system), is equivalent to constraining the ℓ_2 norm of incoming weights for each unit in the hidden layer, and regularizing the ℓ_1 norm of weights in the output layer. This was already observed and proved for networks without an unregularized bias as Theorem 1 of Neyshabur et al. (2014), and in somewhat more general form as Theorem 10 of Neyshabur et al. (2015). The exact same arguments are valid also when an unregularized bias is allowed, and in this Appendix we make this precise, and repeat the arguments of Neyshabur et al. (2014, 2015) for completeness.

Recall the definition of 2-layer ReLU networks h_θ for $\theta \in \Theta_2$:

$$h_\theta(\mathbf{x}) = \sum_{i=1}^{k} w_i^{(2)} \left[\left\langle \mathbf{w}_i^{(1)}, \mathbf{x} \right\rangle + b_i^{(1)}\right]_+ + b^{(2)}$$

Lemma 1

(4) and (5) are equivalent. More specifically:

$$\begin{array}{llcll} \inf\limits_{\theta \in \Theta_2} & \frac{1}{2}\sum\limits_{i=1}^{k}\left((w_i^{(2)})^2 + \left\|\mathbf{w}_i^{(1)}\right\|_2^2\right) & = & \inf\limits_{\theta\in\Theta_2} & \left\|\mathbf{w}^{(2)}\right\|_1 \\ s.t. & h_\theta = f & & s.t. & h_\theta = f\ ,\ \forall i : \left\|\mathbf{w}_i^{(1)}\right\|_2 = 1 \end{array}$$

Proof For any $\theta \in \Theta_2$, consider the rescaled parameters $\tilde{\theta}$ given by $\tilde{\mathbf{w}}_i^{(1)} = c_i \mathbf{w}_i^{(1)}$, $\tilde{w}_i^{(2)} = \frac{w_i^{(2)}}{c_i}$, $\tilde{b}_i^{(1)} = c_i b_i^{(1)}$, for some $c_i > 0$. Now, check that, for all i:

$$\tilde{w}_i^{(2)} \left[\left\langle \tilde{\mathbf{w}}_i^{(1)}, \mathbf{x} \right\rangle + \tilde{b}_i^{(1)}\right]_+ = \frac{w_i^{(2)}}{c_i}\left[c_i\left(\left\langle \mathbf{w}_i^{(1)}, \mathbf{x}\right\rangle + b_i^{(1)}\right)\right]_+ = w_i^{(2)} \left[\left\langle \mathbf{w}_i^{(1)}, \mathbf{x} \right\rangle + b_i^{(1)}\right]_+$$

Therefore $h_\theta = h_{\tilde{\theta}}$. Moreover, we have that, from the inequality between arithmetic and geometric means:

$$\frac{1}{2}\sum_{i=1}^{k}\left((w_i^{(2)})^2 + \left\|\mathbf{w}_i^{(1)}\right\|_2^2\right) \geq \sum_{i=1}^{k} \left|w_i^{(2)}\right| \cdot \left\|\mathbf{w}_i^{(1)}\right\|_2$$

where a rescaling given by $c_i = \sqrt{\left|w_i^{(2)}\right| / \left\|\mathbf{w}_i^{(1)}\right\|_2}$ minimizes the left-hand side and achieves equality. Since the right-hand side is invariant to rescaling, we can arbitrarily set $\left\|\mathbf{w}_i^{(1)}\right\|_2 = 1$ for all i, yielding $\sum_{i=1}^{k}\left|w_i^{(2)}\right| = \left\|\mathbf{w}^{(2)}\right\|_1$. ■

Appendix B. Relationship to Barron's Analysis

Barron (1993, 1994) studies function approximation using two-layer neural network with sigmoidal activation, bounding both the *number* of units required for approximation, and also the ℓ_1 *norm* of the weights *in the output layer*. For a function $f : \mathbb{R}^d \to \mathbb{R}$, Barron defined a quantity C_f, which we refer to as the "Barron Norm"[6]. The core component of Barron's analysis is showing how to approximate f with an infinite number of units, using a measure α over weights, similar to our representation $h_{\alpha,c}$ as in (9) (but with different activation functions), and bounding the ℓ_1 norm $\|\alpha\|_1$ of this measure in terms of C_f. Approximations using a finite number of units can then be obtained by sampling from this measure. Barron's analysis is therefore in many ways similar to ours, suggesting the Barron Norm C_f as the induced complexity measure in function space. However, we point out several important differences between Barron's approach and ours.

One important difference is that while Barron's norm C_f controls the norm of the output layer, it does *not* control the overall norm of the weights across both layers. To understand this better, recall that Barron's analysis is based on first considering cosine activations, then approximating the cosine activations with step functions, and finally approximating the step functions with sigmoidal units. Working with cosine activations, Barron *does* control a norm of the "weights" of each cosine, and so does in a sense provide overall norm control. However, when approximating the cosine with step functions and then in turn with sigmoids, the norm of the weights of the sigmoidal units must increase to infinity in order for them to approximate step functions. The resulting sigmoidal network, although having controlled norm in the output layer, has weights going to infinity in the hidden layer, and thus the overall norm of the network increases to infinity in an controlled and unanalyzed way as we seek better and better approximations (i.e. as $\|f - h_{\alpha,c}\| \to 0$). Controlling the norm of the output layer without controlling the norm in the hidden layer is meaningful for sigmoidal networks because the output of each unit is bounded regardless of its weights, and so the ℓ_∞ norm of the output vector of the hidden layer is always bounded, hence bounding the ℓ_1 norm of the output layer's weights is meaningful. For ReLU networks, it does not make sense to control the weights in one layer without the other, since the activation is positive-homogeneous and the scale of the weights interact.

Furthermore, even though Barron's norm C_f does provide an upper bound on the ℓ_1 norm of the top layer (if we ignore the norm of the weights in the hidden layer), for step or sigmoidal activation this upper bound is not tight. And so, even if we were to regularized only the norm of the the output layer in a sigmoidal network, the induced complexity control in function space is *not* captured by the Barron norm C_f (C_f is only an upper bound on the induced complexity function, and so its form or behaviour might be radically different). Viewed as an approximation theory result, it provides a sufficient, but not a necessary condition for approximability, and so is not, for example, sufficient for studying depth separation. E.g., Lee et al. (2017) provide depth separation results for a generalization of Barron's norm, but this does *not* translate to any meaningful depth separation results for sigmoidal (and certainly not ReLU) neural networks.

A final technical but important issue is that Barron's norm is only useful for approximation results when applied to functions over the entire space $\mathbb{R}^d$, but provides approximation guarantees only in a ball of bounded radius, with a strong dependence on this radius. That is, to study approximability of a function inside a bounded ball, the true quantity this suggests is the minimum Barron norm over all extensions of the function to the entire space, but it is not clear how such a minimum norm extension

[6] The "Barron Norm" can be finite or infinite, and is a semi-norm over the convex set of functions over which it is finite.

would behave. Furthermore, the Barron norm is specific to approximation over Euclidean balls. It can be generalized also to approximation over other compact domains, but the shape of the domain would then change the definition of the norm.

Appendix C. The effect of neural networks depth on the inductive bias

We now turn to a different infinite width architecture, where we study the effect of depth. The networks we consider have a parallel structure common in many state of the art system (Cheng et al., 2016; Shazeer et al., 2017). Consider a deep neural network with L layers and a parallel architecture, so the output is a sum of k sub-networks with $L-1$ layers. Such a network is parameterized by $\theta = \left(k, \mathcal{W}_1, \ldots, \mathcal{W}_k, \mathbf{w}^{(L)}\right)$, where $\mathcal{W}_i = \left(W_i^{(1)}, \ldots, W_i^{(L-1)}\right)$is the set of weight matrices of the ith network, and $\mathbf{w}^{(L)}$ is the weight vector of the last layer, linearly combining the (scalar) outputs of all k parallel sub-networks. The parameter class Θ_L is therefore defined as

$$\Theta_L = \left\{ \theta = \left(k, \mathcal{W}_1, \ldots, \mathcal{W}_k, \mathbf{w}^{(L)}\right) \;\middle|\; k \in \mathbb{N}, \mathbf{w}^{(L)} \in \mathbb{R}^k, \right.$$
$$\left. \forall i \in [k] : \begin{pmatrix} W_i^{(1)} \in \mathbb{R}^{m \times d}, \\ W_i^{(2)} \in \mathbb{R}^{m \times m}, \ldots, W_i^{(L-2)} \in \mathbb{R}^{m \times m}, \\ W_i^{(L-1)} \in \mathbb{R}^{1 \times m} \end{pmatrix} \right\},$$

where $m \in \mathbb{N}$ is the (fixed) width of the layers of the parallel networks.

The network function is defined as

$$h_\theta^L(\mathbf{x}) = \sum_{i=1}^{k} \mathbf{w}_i^{(L)} v\left(\mathcal{W}_i, \mathbf{x}\right), \tag{40}$$

where

$$v\left(\mathcal{W}_i, \mathbf{x}\right) = \left(h^{(L-1)}_{W_i^{(L-1)}} \circ \cdots \circ h^{(1)}_{W_i^{(1)}}\right)(\mathbf{x}), \tag{41}$$

and for $i \in [k], l \in [L-1]$:

$$h^{(i)}_{W_i^{(l)}}(\mathbf{x}) = \left[W_i^{(l)} \mathbf{x}\right]_+, \tag{42}$$

Unlike the networks in Section 2, the networks in this section do **not** have biases.

The squared Euclidean norm of the weights (averaged over all layers) is now defined as

$$C_L(\theta) = \frac{1}{L}\left(\left\|\mathbf{w}^{(L)}\right\|_2^2 + \sum_{i=1}^{k}\sum_{l=1}^{L-1}\left\|W_i^{(l)}\right\|_F^2\right), \tag{43}$$

yielding the following definition of the infimum norm required to implement a function with this parallel architecture:

$$P_L(f) = \inf_{\theta \in \Theta_L} C_L(\theta) \quad \text{s.t. } h_\theta^L = f. \tag{44}$$

In the following Theorem we show that the ℓ_2 norm control can be written equivalently in terms of minimizing a sparsity-inducing ℓ_p bridge penalty $\|\alpha\|_p = \left(\sum_i \alpha_i^p\right)^{1/p}$ where $p = 2/L < 1$ (so we slightly abuse the norm notation as this is non-convex and thus not a norm) in the last layer, while the weights of the subnetworks are restricted to $\mathcal{S}$, the direct product of $L-1$ Euclidean unit spheres, each corresponding to the size of the matrix at its layer:

$$\mathcal{S} = \mathbb{S}^{d\cdot m} \times \mathbb{S}^{m^2} \times \ldots \times \mathbb{S}^{m^2} \times \mathbb{S}^{m} \,. \tag{45}$$

In other words, for each $\bar{\mathcal{W}} = \left(W^{(1)}, \ldots, W^{(L-1)}\right) \in \mathcal{S}$, the weight matrices are normalized, *i.e.,* $\forall l : \left\|W^{(l)}\right\|_F = 1$. It will be convenient to let $\boldsymbol{\alpha}$ denote the weights of the last layer (originally $\mathbf{w}^{(L)}$) in this setting.

Theorem C.1 *If (44) is attainable, then*

$$P_L(f) = \inf_{\theta=(k,\{\bar{\mathcal{W}}_i\}_{i=1}^k,\boldsymbol{\alpha})\in\Theta_L} \|\boldsymbol{\alpha}\|_{2/L}^{2/L} \quad \text{s.t.} \;\; h_\theta^L = f,\; \forall i\in[k] : \bar{\mathcal{W}}_i \in \mathcal{S} \tag{46}$$

where $\mathcal{S}$ is defined in (45), and for each solution of one problem (either (46) or (44)) we can find an equivalent solution of the other (with the same h_θ^L).

Proof The proof appears in Appendix C. ■

The proof of this Theorem has the immediate implication that

Corollary C.2 (Parameter alignment) *Consider any parallel architecture h_θ^L, with k parallel L-layer networks, which minimizes (44), i.e., $h_\theta^L = f$ and $C_L(\theta) = P_L(f)$. Then in each parallel sub-network, equality holds between the ℓ_2-norms of all its (inner) layers and the its corresponding coefficient in the last layer. That is, $\forall i \in [k]$:*

$$\left\|W_i^{*,(1)}\right\|_F^2 = \ldots = \left\|W_i^{*,(L-1)}\right\|_F^2 = \left|\mathbf{w}_i^{*(L)}\right|^2 \tag{47}$$

Proof During the proof of Theorem C.1 we used the inequality of arithmetic and geometric means in (53) to show that $\forall i \in [k]$:

$$\sqrt[L]{\left|\mathbf{w}_i^{*(L)}\right|^2 \prod_{l=1}^{L-1} \left\|W_i^{*,(l)}\right\|_F^2} \le \frac{1}{L}\left(\left|\mathbf{w}_i^{*(L)}\right|^2 + \sum_{l=1}^{L-1} \left\|W_i^{*,(l)}\right\|_F^2\right)$$

After finishing that proof, we know that it must hold with equality. But, equality holds if and only if all the numbers in the means are equal. ■

Next, we consider minimizing an ℓ_2-regularized loss over a finite set of N samples, *i.e.,*

$$\inf_{\theta\in\Theta_L} \sum_{n=1}^N \ell\left(h_\theta^L(\mathbf{x}_n), y_n\right) + \lambda \cdot C_L(\theta) \,, \tag{48}$$

where ℓ is a differentiable convex instantaneous loss function. From Theorem C.1, it follows that (48) is equivalent to minimizing an $\ell_{2/L}$-regularized loss over the same set [7], *i.e.,*

$$\inf_{\theta=\left(k,\{\bar{\mathcal{W}}_i\}_{i=1}^k,\boldsymbol{\alpha}\right)\in\Theta_L} \sum_{n=1}^N \ell\left(h_\theta^L(\mathbf{x}_n), y_n\right) + \lambda \cdot \|\boldsymbol{\alpha}\|_{2/L}^{2/L} \quad \text{s.t.} \; \forall i \in [k] : \bar{\mathcal{W}}_i \in \mathcal{S} \,. \tag{49}$$

[7] Take any minimizer θ^* of (48) and check that it must attain minimum $C_L(\theta)$ when constrained to perfectly fit $\left(\mathbf{x}_n, h_{\theta^*}(\mathbf{x}_n)\right)_{n=1}^N$. The claim then follows from Theorem C.1 and the definition of P_L.

When learning on finite sample sets, the optimal solutions can be shown to be supported on bounded vector sets. More specifically,

Theorem C.3 *For any lower semi-continuous loss ℓ, if (49) is attainable, then it has an optimal solution where $\|\boldsymbol{\alpha}\|_0 \le N$. Moreover, when $L \ge 3$,* ***all*** *optimal solutions of (49) have $\|\boldsymbol{\alpha}\|_0 \le N$.*

Proof The proof appears in Appendix C. ■

This helps us shed light on the behavior of loss minimization when regularizing all parameters, and not only the ones on the last layer.

As the networks become deeper, the regularization term in (49), $\|\boldsymbol{\alpha}\|_{2/L}^{2/L}$, converges to an ℓ_0-regularizer on $\boldsymbol{\alpha}$. That is, this regularizer essentially induces sparsity in the weights of the last linear layer. Moreover, when L is indeed large enough and the regularizer practically behaves like an ℓ_0-regularizer, all optimal solutions to (49) should have the same number of non-zero weights in the last layer. This implies that all optimal solutions of (48) also have the same number of non-zero weights in their last layer (otherwise our construction (55) in the proof of Theorem C.1 will yield a suboptimal solution which is impossible). This means the ℓ_2-regularized loss minimization problem (48) will implicitly zero out as many sub-networks in the parallel architecture as possible. This result is closely related to the result of Gunasekar et al. (2018), which found a similar inductive bias in certain *linear* convolutional neural nets. Here we show such a sparsity-inducing bias (which gets stronger with depth) also affects non-linear deep networks.

Proofs

Proof for Theorem C.1

Proof In the following proof we show that given an optimal solution of either of the two problems, one can construct a feasible solution to the second one, with an equal objective value. We follow a proof by Wei et al. (2018), who used a similar construction to bind the max margin of ℓ_2-regularized neural networks with one hidden layer, with the max margin of convex neural networks with one (infinite) hidden layer and an ℓ_1-regularization over the last layer.

- $C_L(\theta^*) \ge \|\boldsymbol{\alpha}^*\|_{2/L}^{2/L}$**:** Given a solution

$$\theta^* = \left(k^*, \mathcal{W}_1^*, \dots, \mathcal{W}_{k^*}^*, \mathbf{w}^{*(L)}\right)$$

to (44), we show how to construct a solution for (46). Start by setting the following solution, where $\forall i \in [k^*]$:

$$\begin{aligned} \bar{\mathcal{W}}_i &= \left(\bar{W}_i^{*,(1)}, \dots, \bar{W}_i^{*,(L-1)}\right) \\ \boldsymbol{\alpha}_i &= \mathbf{w}_i^{*(L)} \cdot \left\|W_i^{*,(1)}\right\|_F \cdots \left\|W_i^{*,(L-1)}\right\|_F \ . \end{aligned} \tag{50}$$

We use the 1-positive-homogeneity of the activation functions to show $\forall i \in [k^*]$ it holds that

$$\begin{aligned} v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) \prod_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F &= v\left(\left(\bar{W}_i^{*,(1)}, \dots, \bar{W}_i^{*,(L-1)}\right), \mathbf{x}\right) \cdot \prod_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F \\ &= v\left(\left(\left\| W_i^{*,(1)} \right\|_F \bar{W}_i^{*,(1)}, \dots, \left\| W_i^{*,(L-1)} \right\|_F \bar{W}_i^{*,(L-1)}\right), \mathbf{x}\right) \\ &= v\left(\mathcal{W}_i^*, \mathbf{x}\right) \ . \end{aligned} \tag{51}$$

Now note that the constructed solution is feasible, since $\bar{\mathcal{W}}_i \in \mathcal{S}, \forall i \in [k^*]$ and $\forall \mathbf{x} \in \mathbb{R}^d$ it holds

$$\begin{aligned} h^L_{\theta=\left(k^*, \left\{\bar{\mathcal{W}}_i\right\}_{i=1}^{k^*}, \boldsymbol{\alpha}\right)}\left(\mathbf{x}\right) &= \sum_{i=1}^{k^*} \boldsymbol{\alpha}_i v\left(\bar{\mathcal{W}}_i, \mathbf{x}\right) = \sum_{i=1}^{k^*} \mathbf{w}_i^{*(L)} \cdot \prod_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F \cdot v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) \\ &= \sum_{i=1}^{k^*} \mathbf{w}_i^{*(L)} \cdot v\left(\mathcal{W}_i^*, \mathbf{x}\right) = h^L_{\theta^*}\left(\mathbf{x}\right) = f\left(\mathbf{x}\right) \ . \end{aligned} \tag{52}$$

We now show that the value of the new solution, *i.e.,* its norm, equals

$$\begin{aligned} \left\|\boldsymbol{\alpha}\right\|_{2/L}^{2/L} &= \sum_{i=1}^{k^*} \left| \mathbf{w}_i^{*(L)} \cdot \prod_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F \right|^{2/L} = \sum_{i=1}^{k^*} \left(\left| \mathbf{w}_i^{*(L)} \right|^2 \cdot \prod_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F^2 \right)^{\frac{1}{L}} \\ &\leq \sum_{i=1}^{k^*} \frac{1}{L} \left(\left| \mathbf{w}_i^{*(L)} \right|^2 + \sum_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F^2 \right) \\ &= \frac{1}{L} \left(\left\| \mathbf{w}^{*(L)} \right\|_2^2 + \sum_{i=1}^{k^*} \sum_{l=1}^{L-1} \left\| W_i^{*,(l)} \right\|_F^2 \right) = C_L\left(\theta^*\right) \ , \end{aligned} \tag{53}$$

where we used the inequality of arithmetic and geometric means.

As a conclusion, we get the required inequality

$$\left\|\boldsymbol{\alpha}^*\right\|_{2/L}^{2/L} \leq \left\|\boldsymbol{\alpha}\right\|_{2/L}^{2/L} \leq C_L\left(\theta^*\right) \ . \tag{54}$$

- $C_L\left(\theta^*\right) \leq \left\|\boldsymbol{\alpha}^*\right\|_{2/L}^{2/L}$**:** When (46) is attainable, and given an optimal solution $\theta = \left(k^*, \boldsymbol{\alpha}^*, \left\{\bar{\mathcal{W}}_i^*\right\}_{i=1}^{k^*}\right)$, we show how to attain a solution to the implementation cost formulation (44).

  We construct a solution θ where $\forall i \in [k^*]$

$$\mathbf{w}_i^{(L)} = \mathrm{sign}\left(\boldsymbol{\alpha}_i^*\right) \left|\boldsymbol{\alpha}_i^*\right|^{1/L} \tag{55}$$

$$\mathcal{W}_i = \left|\boldsymbol{\alpha}_i^*\right|^{1/L} \cdot \bar{\mathcal{W}}_i^* \ . \tag{56}$$

  By using the 1-positive-homogeneity property of the activation functions (once for every matrix in $\bar{\mathcal{W}}_i$), we get that for all $i \in [k^*]$:

$$v\left(\mathcal{W}_i, \mathbf{x}\right) = v\left(\left|\boldsymbol{\alpha}_i^*\right|^{\frac{1}{L}} \bar{\mathcal{W}}_i^*, \mathbf{x}\right) = \left|\boldsymbol{\alpha}_i^*\right|^{\frac{L-1}{L}} v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) \tag{57}$$

Now, the constructed solution can be shown to be feasible, since $\forall \mathbf{x} \in \mathbb{R}^d$:

$$
\begin{aligned}
h_\theta^L(\mathbf{x}) &= \sum_{i=1}^{k^*} \mathbf{w}_i^{(L)} v(\mathcal{W}_i, \mathbf{x}) = \sum_{i=1}^{k^*} \operatorname{sign}(\boldsymbol{\alpha}_i^*) |\boldsymbol{\alpha}_i^*|^{\frac{1}{L}+\frac{L-1}{L}} v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) \\
&= \sum_{i=1}^{k^*} \operatorname{sign}(\boldsymbol{\alpha}_i^*) |\boldsymbol{\alpha}_i^*| \, v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) = \sum_{i=1}^{k^*} \boldsymbol{\alpha}_i^* v\left(\bar{\mathcal{W}}_i^*, \mathbf{x}\right) \\
&= h^L_{\theta=\left(k^*, \left\{\bar{\mathcal{W}}_i^*\right\}_{i=1}^{k^*, \boldsymbol{\alpha}^*}\right)}(\mathbf{x}) = f(\mathbf{x}) \ .
\end{aligned}
\tag{58}
$$

The value of this solution is

$$
\begin{aligned}
C_L(\theta) &= \frac{1}{L}\left(\left\| \mathbf{w}^{(L)} \right\|_2^2 + \sum_{i=1}^{k^*} \sum_{l=1}^{L-1} \left\| W_i^{(l)} \right\|_F^2 \right) \\
&= \frac{1}{L} \sum_{i=1}^{k^*} \left(|\boldsymbol{\alpha}_i^*|^{2/L} + |\boldsymbol{\alpha}_i^*|^{2/L} \cdot \sum_{l=1}^{L-1} \underbrace{\left\| \bar{W}_i^{(l)} \right\|_F^2}_{=1} \right) \\
&= \frac{1}{L} \sum_{i=1}^{k^*} \left[|\boldsymbol{\alpha}_i^*|^{2/L} (1 + (L-1)) \right] = \frac{L}{L} \sum_{i=1}^{k^*} |\boldsymbol{\alpha}_i^*|^{2/L} = \|\boldsymbol{\alpha}^*\|_{2/L}^{2/L} \ .
\end{aligned}
\tag{59}
$$

As a conclusion,

$$
C_L(\theta^*) \le C_L(\theta) = \|\boldsymbol{\alpha}^*\|_{2/L}^{2/L} \ . \tag{60}
$$

Overall we get the required equality:

$$
C_L(\theta^*) = \|\boldsymbol{\alpha}^*\|_{2/L}^{2/L} \ . \tag{61}
$$

■

Proof for Theorem C.3

Proof

When $L = 2 \Rightarrow \frac{2}{L} = 1$, our minimization problem is a standard ℓ_1-penalized problem. Rosset et al. (2007, Theorem 2) use Carathodory's theorem and prove that there exists an optimal solution with a finite support size, which is at most $N + 1$ (using Carathodory's theorem). Tibshirani et al. (2013, Lemma 14) demonstrate how an iterative procedure (using a technique similar to what we do below) can zero out an (attainable) optimal solution with a finite support, until an optimal solution with a support size at most N is attained.

We therefore narrow our proof to deeper networks where $L \ge 3 \Rightarrow \frac{2}{L} < 1$ and the norm in the objective function is no longer convex (but quasi-convex). Following a proof by Petrov (2019), we are able to show that when $L \ge 3$, not only there exists an optimal solution with a finite support of size at most N, but **all** optimal solution are such.

We start by assuming the contrary – there exists an optimal solution θ^*, with a set of unit vectors $\left\{\bar{\mathcal{W}}_1, \dots, \bar{\mathcal{W}}_{N+1}\right\} \subseteq \theta^*$ such that w.l.o.g. $\|\boldsymbol{\alpha}\|_0 = N+1$ (that is, $\boldsymbol{\alpha}_i^* \neq 0, \forall i \in [N+1]$). We wish to find a non-zero vector , $\boldsymbol{b} \in \mathbb{R}^{N+1}$ such that $\forall i \in [N+1]\,, n \in [N] : \boldsymbol{b}_i v\left(\bar{\mathcal{W}}_i, x_n\right) = 0$. Notice we have only N constraints and $N+1$ variables, meaning we get an homogeneous underdetermined system. We can thus always choose such a vector $\boldsymbol{\beta} \in \mathbb{R}^{k^*}$ such that

$$\boldsymbol{\beta}_i = \begin{cases} \boldsymbol{b}_i & i \in [N+1] \\ 0 & \text{otherwise} \end{cases} \tag{62}$$

and $\sum_{i=1}^{N+1} \boldsymbol{\beta}_i v\left(\bar{\mathcal{W}}_i, x_n\right) = 0, \forall n \in [N]$, which means the solutions $\boldsymbol{\alpha}^* + \rho\boldsymbol{\beta}, \forall \rho \in \mathbb{R}$ have the same network output as $\boldsymbol{\alpha}^*$. Now choose $\rho > 0$ small enough, such that $\forall i \in [N+1]$:

$$|\boldsymbol{\alpha}_i^*| - \rho\,|\boldsymbol{\beta}_i| = s_i \boldsymbol{\alpha}_i^* - \rho\,|\boldsymbol{\beta}_i| > 0\ , \tag{63}$$

where $s_i \triangleq \operatorname{sign} \boldsymbol{\alpha}_i^*$. The function $z^{\frac{2}{L}}$ is concave $\forall z \geq 0$, and so we apply the Jensen inequality to get

$$\begin{aligned} \|\boldsymbol{\alpha}^*\|_{2/L}^{2/L} \geq \sum_i^{N+1} |\boldsymbol{\alpha}_i^*|^{\frac{2}{L}} &= \sum_i^{N+1} \left(s_i \boldsymbol{\alpha}_i^*\right)^{\frac{2}{L}} \\ &= \sum_i^{N+1} \left(\frac{1}{2}\left(s_i\boldsymbol{\alpha}_i^* + \rho\boldsymbol{\beta}_i\right) + \frac{1}{2}\left(s_i\boldsymbol{\alpha}_i^* - \rho\boldsymbol{\beta}_i\right)\right)^{\frac{2}{L}} \\ &> \frac{1}{2}\sum_i^{N+1}\left(\underbrace{\left(s_i\boldsymbol{\alpha}_i^* + \rho\boldsymbol{\beta}_i\right)^{\frac{2}{L}}}_{>0} + \underbrace{\left(s_i\boldsymbol{\alpha}_i^* - \rho\boldsymbol{\beta}_i\right)^{\frac{2}{L}}}_{>0}\right) \\ &= \frac{1}{2}\sum_i^{N+1}\left(|\boldsymbol{\alpha}_i^* + s_i\rho\boldsymbol{\beta}_i|^{\frac{2}{L}} + |\boldsymbol{\alpha}_i^* - s_i\rho\boldsymbol{\beta}_i|^{\frac{2}{L}}\right) \\ &= \frac{1}{2}\sum_i^{N+1}\left(|\boldsymbol{\alpha}_i^* + \rho\boldsymbol{\beta}_i|^{\frac{2}{L}} + |\boldsymbol{\alpha}_i^* - \rho\boldsymbol{\beta}_i|^{\frac{2}{L}}\right) \\ &= \frac{1}{2}\left(\|\boldsymbol{\alpha}^* + \rho\boldsymbol{\beta}\|_{2/L}^{2/L} + \|\boldsymbol{\alpha}^* - \rho\boldsymbol{\beta}\|_{2/L}^{2/L}\right)\ . \end{aligned} \tag{64}$$

Notice we used the strict Jensen inequality, since neither the function $z^{\frac{2}{L}}$ is linear, nor are the two terms equal (following our choice of ρ in (63)).

Finally, we notice the above imply that one of the two solutions $\boldsymbol{\alpha}^* \pm \rho\boldsymbol{\beta}$ must have a strictly smaller norm than $\boldsymbol{\alpha}^*$. As a result of our choice of $\boldsymbol{\beta}$, all three solutions have the same loss, *i.e.,*

$$\sum_{n=1}^{N} \ell\left(h_{\boldsymbol{\alpha}^*}(x_n)\,, y_n\right) = \sum_{n=1}^{N} \ell\left(h_{\boldsymbol{\alpha}^*+\rho\boldsymbol{\beta}}(x_n), y_n\right) = \sum_{n=1}^{N} \ell\left(h_{\boldsymbol{\alpha}^*-\rho\boldsymbol{\beta}}(x_n)\,, y_n\right)\ . \tag{65}$$

The above necessarily mean that at least one of these two solutions we constructed has an objective value strictly smaller than the objective value of $\boldsymbol{\alpha}^*$, in contradiction to its optimality. ■

Appendix D. Proof of Claim 5.2

Proof If $\|\mathbf{x}\| > b$ then,

$$\begin{aligned}
&\int_{\mathbb{S}^{d-1}} d\mathbf{w} \left[\mathbf{w}^\top \mathbf{x} + b\right]_+ \\
&\overset{(1)}{=} \int_{\mathbb{S}^{d-1}} d\mathbf{w} \left[\|\mathbf{x}\| w_1 + b\right]_+ \\
&\overset{(2)}{=} \|\mathbf{x}\| \int_{-1}^{1} dw_1 \left[w_1 + \frac{b}{\|\mathbf{x}\|}\right]_+ \int_{\mathbf{w}' \in \mathbb{S}^{d-1}: w_1' = w_1} \prod_{i=2}^{d} dw_i' \\
&\overset{(3)}{=} \|\mathbf{x}\| \int_{\max\left[-1, -\frac{b}{\|\mathbf{x}\|}\right]}^{1} dw_1 \left(w_1 + \frac{b}{\|\mathbf{x}\|}\right) S_{d-1} \left(1 - w_1^2\right)^{\frac{d-1}{2}} \\
&\overset{(4)}{=} \|\mathbf{x}\| S_{d-1} \int_{-\frac{b}{\|\mathbf{x}\|}}^{1} dw_1 w_1 \left(1 - w_1^2\right)^{\frac{d-1}{2}} + S_{d-1} \int_{-\frac{b}{\|\mathbf{x}\|}}^{1} dw_1 \left(1 - w_1^2\right)^{\frac{d-1}{2}} \\
&\overset{(5)}{=} \|\mathbf{x}\| S_{d-1} \int_{-\frac{b}{\|\mathbf{x}\|}}^{1} dw_1 w_1 \left(1 - w_1^2\right)^{\frac{d-1}{2}} + C + f_{\text{odd}}\left(\frac{b}{\|\mathbf{x}\|}\right) \\
&= \|\mathbf{x}\| \frac{S_{d-1}}{d+1} \left(1 - \frac{b^2}{\|\mathbf{x}\|^2}\right)^{\frac{d+1}{2}} + C + f_{\text{odd}}\left(\frac{b}{\|\mathbf{x}\|}\right)
\end{aligned}$$

In (1) we assume $d > 1$ and, WLOG, that $\mathbf{x}$ is in direction $(1, 0, \ldots, 0)$, in (2) we integrate over the surface area of a $d-2$ dimensional sphere, in (3) we denoted S_d as the surface area of the d-sphere, in (4) we assume that $\|\mathbf{x}\| > b$, and in (5) we used the fact that the integral of an even function is equal to the sum of a constant C with and odd function f_{odd}. Therefore, for $\|\mathbf{x}\| > b$

$$\begin{aligned}
h(\mathbf{x}) &= \int_{\mathbb{S}^{d-1}} d\mathbf{w} \left(\left[\mathbf{w}^\top \mathbf{x} + 1\right]_+ - 2\left[\mathbf{w}^\top \mathbf{x}\right]_+ + \left[\mathbf{w}^\top \mathbf{x} - 1\right]_+\right) \\
&= 2\|\mathbf{x}\| \frac{S_{d-1}}{d+1} \left[\left(1 - \frac{1}{\|\mathbf{x}\|^2}\right)^{\frac{d+1}{2}} - 1\right] \\
&= S_{d-1} \frac{1}{\|\mathbf{x}\|} + O\left(\frac{1}{\|\mathbf{x}\|^3}\right).
\end{aligned}$$

From here it is straightforward to see that

$$\nabla h(\mathbf{x}) \sim -\frac{\mathbf{x}}{\|\mathbf{x}\|^3}$$

$$\nabla^2 h(\mathbf{x}) \sim \frac{1}{\|\mathbf{x}\|^3} \left(3 \frac{\mathbf{x}\mathbf{x}^\top}{\|\mathbf{x}\|^2} - \mathbf{I}\right)$$

Therefore, for some constants r_0, C, and any norm,

$$\begin{aligned}
\frac{1}{r^{d-1}} \int_{\|\mathbf{x}\| \le r} d\mathbf{x} \left\|\nabla^2 h(\mathbf{x})\right\| \le \frac{C}{r^{d-1}} \int_{r_0}^{r} u^{d-1} \frac{1}{u^3} du = \frac{C}{r^{d-1}} \int_{r_0}^{r} u^{d-4} du \\
= \frac{C}{r^{d-1}} r^{d-3} = \frac{C}{r^2}
\end{aligned}$$

Which vanishes as $r \to \infty$. ■