mathematics

Article

Universal Function Approximation by Deep Neural Nets with Bounded Width and ReLU Activations

Boris Hanin

Department of Mathematics, Texas A&M, College Station, TX 77843, USA; bhanin@math .tamu.edu

Received: 29 September 2019; Accepted: 16 October 2019; Published: 18 October 2019

Abstract: This article concerns the expressive power of depth in neural nets with ReLU activations and a bounded width. We are particularly interested in the following questions: What is the minimal width $w_{\min}(d)$ so that ReLU nets of width $w_{\min}(d)$ (and arbitrary depth) can approximate any continuous function on the unit cube $[0,1]^d$ arbitrarily well? For ReLU nets near this minimal width, what can one say about the depth necessary to approximate a given function? We obtain an essentially complete answer to these questions for convex functions. Our approach is based on the observation that, due to the convexity of the ReLU activation, ReLU nets are particularly well suited to represent convex functions. In particular, we prove that ReLU nets with width $d+1$ can approximate any continuous convex function of d variables arbitrarily well. These results then give quantitative depth estimates for the rate of approximation of any continuous scalar function on the d-dimensional cube $[0,1]^d$ by ReLU nets with width $d+3$.

Keywords: Deep Neural Nets; ReLU Networks; Approximation Theory

1. Introduction

Over the past several years, neural nets, particularly deep nets, have become the state-of-the-art in a remarkable number of machine learning problems, from mastering go to image recognition/segmentation and machine translation (see the review article [1] for more background). Despite all their practical successes, a robust theory of why they work so well is in its infancy. Much of the work to date has focused on the problem of explaining and quantifying the expressivity (the ability to approximate a rich class of functions) of deep neural nets [2–11]. Expressivity can be seen both as an effect of both depth and width. It has been known since at least the work of Cybenko [12] and Hornik-Stinchcombe-White [13] that if no constraint is placed on the width of a hidden layer, then a single hidden layer is enough to approximate essentially any function. The purpose of this article, in contrast, is to investigate the "effect of depth without the aid of width." More precisely, for each $d \geq 1$, we would like to estimate:

$$w_{\min}(d) := \min\left\{ w \in \mathbb{N} \;\middle|\; \begin{array}{c}\text{ReLU nets of width } w \text{ can approximate any}\\ \text{positive continuous function on } [0,1]^d \text{ arbitrarily well}\end{array} \right\}. \qquad (1)$$

Here, $\mathbb{N} = \{0, 1, 2, \ldots\}$ are the natural numbers and ReLU is the so-called "rectified linear unit," $\mathrm{ReLU}(t) = \max\{0, t\}$, which is the most popular non-linearity used in practice (see (4) for the exact definition). In Theorem 1, we prove that $\omega_{\min}(d) \leq d+2$. This raises two questions:

Q1. Is the estimate in the previous line sharp?

Q2. How efficiently can ReLU nets of a given width $w \geq w_{\min}(d)$ approximate a given continuous function of d variables?

A priori, it is not clear how to estimate $\omega_{min}(d)$ and whether it is even finite. One of the contributions of this article is to provide reasonable bounds on $\omega_{min}(d)$ (see Theorem 1). Moreover, we also provide quantitative estimates on the corresponding rate of approximation. On the subject of Q1, we will prove in forthcoming work with M.Sellke [14] that in fact, $\omega_{\min}(d) = d+1$. When $d=1$, the lower bound is simple to check, and the upper bound follows for example from Theorem 3.1 in [5]. The main results in this article, however, concern Q1 and Q2 for convex functions. For instance, we prove in Theorem 1 that:

$$w_{\min}^{\text{conv}}(d) \leq d+1, \tag{2}$$

where:

$$w_{\min}^{\text{conv}}(d) := \min \left\{ w \in \mathbb{N} \;\middle|\; \begin{array}{c} \text{ReLU nets of width } w \text{ can approximate any} \\ \text{positive convex function on } [0,1]^d \text{ arbitrarily well} \end{array} \right\}. \tag{3}$$

This illustrates a central point of the present paper: the convexity of the ReLU activation makes ReLU nets well-adapted to representing convex functions on $[0,1]^d$.

Theorem 1 also addresses Q2 by providing quantitative estimates on the depth of a ReLU net with width $d+1$ that approximates a given convex function. We provide similar depth estimates for arbitrary continuous functions on $[0,1]^d$, but this time for nets of width $d+3$. Several of our depth estimates are based on the work of Balázs-György-Szepesvári [15] on max-affine estimators in convex regression.

In order to prove Theorem 1, we must understand what functions can be exactly computed by a ReLU net. Such functions are always piecewise affine, and we prove in Theorem 2 the converse: every piecewise affine function on $[0,1]^d$ can be exactly represented by a ReLU net with hidden layer width at most $d+3$. Moreover, we prove that the depth of the network that computes such a function is bounded by the number affine pieces it contains. This extends the results of Arora-Basu-Mianjy-Mukherjee (e.g., Theorem 2.1 and Corollary 2.2 in [2]).

Convex functions again play a special role. We show that every convex function on $[0,1]^d$ that is piecewise affine with N pieces can be represented exactly by a ReLU net with width $d+1$ and depth N.

2. Statement of Results

To state our results precisely, we set notation and recall several definitions. For $d \geq 1$ and a continuous function $f : [0,1]^d \to \mathbb{R}$, write:

$$\|f\|_{C^0} := \sup_{x \in [0,1]^d} |f(x)|.$$

Further, denote by:

$$\omega_f(\varepsilon) := \sup\{|f(x) - f(y)| \mid |x-y| \leq \varepsilon\}$$

the modulus of continuity of f, whose value at ε is the maximum that f can change when its argument moves by at most ε. Note that by the definition of a continuous function, $\omega_f(\varepsilon) \to 0$ as $\varepsilon \to 0$. Next, given $d_{\text{in}}, d_{\text{out}}$, and $w \geq 1$, we define a feed-forward neural net with ReLU activations, input dimension d_{in}, hidden layer width w, depth n, and output dimension d_{out} to be any member of the finite-dimensional family of functions:

$$\text{ReLU} \circ A_n \circ \cdots \circ \text{ReLU} \circ A_1 \circ \text{ReLU} \circ A_1 \tag{4}$$

that map $\mathbb{R}^d$ to $\mathbb{R}_+^{d_{\text{out}}} = \{x = (x_1, \ldots, x_{d_{\text{out}}}) \in \mathbb{R}^{d_{\text{out}}} \mid x_i \geq 0\}$. In (4),

$$A_j : \mathbb{R}^w \to \mathbb{R}^w,\; j = 2, \ldots, n-1, \qquad A_1 : \mathbb{R}^{d_{\text{in}}} \to \mathbb{R}^w,\; A_n : \mathbb{R}^w \to \mathbb{R}^{d_{\text{out}}}$$

are affine transformations, and for every $m \geq 1$:

$$\operatorname{ReLU}(x_1, \ldots, x_m) = (\max\{0, x_1\}, \ldots, \max\{0, x_m\}) .$$

We often denote such a net by $\mathcal{N}$ and write:

$$f_{\mathcal{N}}(x) := \operatorname{ReLU} \circ A_n \circ \cdots \circ \operatorname{ReLU} \circ A_1 \circ \operatorname{ReLU} \circ A_1(x)$$

for the function it computes. Our first result contrasts both the width and depth required to approximate continuous, convex, and smooth functions by ReLU nets.

Theorem 1. *Let $d \geq 1$ and $f : [0,1]^d \to \mathbb{R}_+$ be a positive function with $\|f\|_{C^0} = 1$. We have the following three cases:*

1. (f is continuous) *There exists a sequence of feed-forward neural nets $\mathcal{N}_k$ with ReLU activations, input dimension d, hidden layer width $d+2$, and output dimension 1, such that:*

$$\lim_{k\to\infty} \|f - f_{\mathcal{N}_k}\|_{C^0} = 0. \tag{5}$$

In particular, $w_{min}(d) \leq d+2$. Moreover, write ω_f for the modulus of continuity of f, and fix $\varepsilon > 0$. There exists a feed-forward neural net $\mathcal{N}_\varepsilon$ with ReLU activations, input dimension d, hidden layer width $d+3$, output dimension 1, and:

$$depth\,(\mathcal{N}_\varepsilon) = \frac{2 \cdot d!}{\omega_f(\varepsilon)^d} \tag{6}$$

such that:

$$\|f - f_{\mathcal{N}_\varepsilon}\|_{C^0} \leq \varepsilon. \tag{7}$$

2. (f is convex) *There exists a sequence of feed-forward neural nets $\mathcal{N}_k$ with ReLU activations, input dimension d, hidden layer width $d+1$, and output dimension 1, such that:*

$$\lim_{k\to\infty} \|f - f_{\mathcal{N}_k}\|_{C^0} = 0. \tag{8}$$

Hence, $\omega_{min}^{conv}(d) \leq d+1$. Further, there exists $C > 0$ such that if f is both convex and Lipschitz with Lipschitz constant L, then the nets $\mathcal{N}_k$ in (8) can be taken to satisfy:

$$depth\,(\mathcal{N}_k) = k+1, \qquad \|f - f_{\mathcal{N}_k}\|_{C^0} \leq CLd^{3/2}k^{-2/d}. \tag{9}$$

3. (f is smooth) *There exists a constant K depending only on d and a constant C depending only on the maximum of the first K derivative of f such that for every $k \geq 3$, the width $d+2$ nets $\mathcal{N}_k$ in (5) can be chosen so that:*

$$depth(\mathcal{N}_k) = k, \qquad \|f - f_{\mathcal{N}_k}\|_{C^0} \leq C\,(k-2)^{-1/d}. \tag{10}$$

The main novelty of Theorem 1 is the width estimate $w_{\mathrm{min}}^{\mathrm{conv}}(d) \leq d+1$ and the quantitative depth estimates (9) for convex functions, as well as the analogous estimates (6) and (7) for continuous functions. Let us briefly explain the origin of the other estimates. The relation (5) and the corresponding estimate $w_{\mathrm{min}}(d) \leq d+2$ are a combination of the well-known fact that ReLU nets with one hidden layer can approximate any continuous function and a simple procedure by which a ReLU net with input dimension d and a single hidden layer of width n can be replaced by another ReLU net that computes the same function, but has depth $n+2$ and width $d+2$. For these width $d+2$ nets, we are unaware of how to obtain quantitative estimates on the depth required to approximate a fixed continuous function to a given precision. At the expense of changing the width of our ReLU nets from $d+2$ to $d+3$, however, we furnish the estimates (6) and (7). On the other hand, using Theorem 3.1 in [5], when f is sufficiently smooth, we obtain the depth estimates (10) for width $d+2$ ReLU nets. Indeed, since we are working on a compact set $[0,1]^d$, the smoothness classes $W_{w,q,\gamma}$ from [5] reduce to classes of functions that have sufficiently many bounded derivatives.

Our next result concerns the exact representation of piecewise affine functions by ReLU nets. Instead of measuring the complexity of such a function by its Lipschitz constant or modulus of continuity, the complexity of a piecewise affine function can be thought of as the minimal number of affine pieces needed to define it.

Theorem 2. *Let $d \geq 1$ and $f : [0,1]^d \to \mathbb{R}_+$ be the function computed by some* ReLU *net with input dimension d, output dimension 1, and arbitrary width. There exist affine functions $g_\alpha, h_\beta : [0,1]^d \to \mathbb{R}$ such that f can be written as the difference of positive convex functions:*

$$f = g - h, \qquad g := \max_{1\leq\alpha\leq N} g_\alpha, \qquad h := \max_{1\leq\beta\leq M} h_\beta. \tag{11}$$

Moreover, there exists a feed-forward neural net $\mathcal{N}$ with ReLU activations, input dimension d, hidden layer width $d+3$, output dimension 1, and:

$$\mathit{depth}\,(\mathcal{N}) = 2(M+N) \tag{12}$$

that computes f exactly. Finally, if f is convex (and hence, h vanishes), then the width of $\mathcal{N}$ can be taken to be $d+1$, and the depth can be taken to be N.

The fact that the function computed by a ReLU net can be written as (11) follows from Theorem 2.1 in [2]. The novelty in Theorem 2 is therefore the uniform width estimate $d+3$ in the representation on any function computed by a ReLU net and the $d+1$ width estimate for convex functions. Theorem 2 will be used in the proof of Theorem 1.

3. Relation to Previous Work

This article is related to several strands of prior work:

1. Theorems 1 and 2 are "deep and narrow" analogs of the well-known "shallow and wide" universal approximation results (e.g., Cybenko [12] and Hornik-Stinchcombe-White [13]) for feed-forward neural nets. Those articles show that essentially any scalar function $f : [0,1]^d \to \mathbb{R}$ on the d-dimensional unit cube can be arbitrarily well approximated by a feed-forward neural net with a single hidden layer with arbitrary width. Such results hold for a wide class of nonlinear activations, but are not particularly illuminating from the point of understanding the expressive advantages of depth in neural nets.
2. The results in this article complement the work of Liao-Mhaskar-Poggio [3] and Mhaskar-Poggio [5], who considered the advantages of depth for representing certain hierarchical or compositional functions by neural nets with both ReLU and non-ReLU activations. Their results (e.g., Theorem 1 in [3] and Theorem 3.1 in [5]) give bounds on the width for approximation both for shallow and certain deep hierarchical nets.
3. Theorems 1 and 2 are also quantitative analogs of Corollary 2.2 and Theorem 2.4 in the work of Arora-Basu-Mianjy-Mukerjee [2]. Their results give bounds on the depth of a ReLU net needed to compute exactly a piecewise linear function of d variables. However, except when $d=1$, they do not obtain an estimate on the number of neurons in such a network and hence cannot bound the width of the hidden layers.
4. Our results are related to Theorems II.1 and II.4 of Rolnick-Tegmark [16], which are themselves extensions of Lin-Rolnick-Tegmark [4]. Their results give lower bounds on the total size (number of neurons) of a neural net (with non-ReLU activations) that approximates sparse multivariable polynomials. Their bounds do not imply a control on the width of such networks that depends only on the number of variables, however.

5. This work was inspired in part by questions raised in the work of Telgarsky [8–10]. In particular, in Theorems 1.1 and 1.2 of [8], Telgarsky constructed interesting examples of sawtooth functions that can be computed efficiently by deep width 2 ReLU nets that cannot be well approximated by shallower networks with a similar number of parameters.
6. Theorems 1 and 2 are quantitative statements about the expressive power of depth without the aid of width. This topic, usually without considering bounds on the width, has been taken up by many authors. We refer the reader to [6,7] for several interesting quantitative measures of the complexity of functions computed by deep neural nets.
7. Finally, we refer the reader to the interesting work of Yarofsky [11], which provides bounds on the total number of parameters in a ReLU net needed to approximate a given class of functions (mainly balls in various Sobolev spaces).

4. Proof of Theorem 2

Proof of Theorem 2. We first treat the case:

$$f = \sup_{1\leq \alpha \leq N} g_\alpha, \qquad g_\alpha : [0,1]^d \to \mathbb{R} \quad \text{affine}$$

when f is convex. We seek to show that f can be exactly represented by a ReLU net with input dimension d, hidden layer width $d+1$, and depth N. Our proof relies on the following observation.

Lemma 1. *Fix $d \geq 1$, and let $T : \mathbb{R}_+^d \to \mathbb{R}$ be an arbitrary function and $L : \mathbb{R}^d \to \mathbb{R}$ be affine. Define an invertible affine transformation $A : \mathbb{R}^{d+1} \to \mathbb{R}^{d+1}$ by:*

$$A(x, y) = (x, L(x) + y)\,.$$

Then, the image of the graph of T under:

$$A \circ \mathrm{ReLU} \circ A^{-1}$$

is the graph of $x \mapsto \max\{T(x), L(x)\}$, viewed as a function on $\mathbb{R}_+^d$.

Proof. We have $A^{-1}(x, y) = (x, -L(x) + y)$. Hence, for each $x \in \mathbb{R}_+^d$, we have:

$$\begin{aligned} A \circ \mathrm{ReLU} \circ A^{-1}(x, T(x)) &= \Big(x, (T(x) - L(x))\, \mathbf{1}_{\{T(x)-L(x)>0\}} + L(x)\Big) \\ &= (x, \max\{T(x), L(x)\})\,. \end{aligned}$$

□

We now construct a neural net that computes f. We note that the construction is potentially applicable to the study of avoiding sets (see the work of Shang [17]). Define invertible affine functions $A_\alpha : \mathbb{R}^{d+1} \to \mathbb{R}^{d+1}$ by:

$$A_\alpha(x, x_{d+1}) := (x, g_\alpha(x) + x_{d+1})\,, \qquad x = (x_1, \ldots, x_d),$$

and set:

$$H_\alpha := A_\alpha \circ \mathrm{ReLU} \circ A_\alpha^{-1}.$$

Further, define:

$$H_{\mathrm{out}} := \mathrm{ReLU} \circ \langle \vec{e}_{d+1}, \cdot \rangle \tag{13}$$

where $\vec{e}_{d+1}$ is the $(d+1)$th standard basis vector so that $\langle \vec{e}_{d+1}, \cdot \rangle$ is the linear map from $\mathbb{R}^{d+1}$ to $\mathbb{R}$ that maps $(x_1, \ldots, x_{d+1})$ to x_{d+1}. Finally, set:

$$H_{\text{in}} := \text{ReLU} \circ (\text{id}, 0)\,,$$

where $(\text{id}, 0)\,(x) = (x, 0)$ maps $[0,1]^d$ to the graph of the zero function. Note that the ReLU in this initial layer is linear. With this notation, repeatedly using Lemma 1, we find that:

$$H_{\text{out}} \circ H_N \circ \cdots \circ H_1 \circ H_{\text{in}}$$

therefore has input dimension d, hidden layer width $d+1$, depth N, and computes f exactly.

Next, consider the general case when f is given by:

$$f = g - h, \qquad g = \sup_{1 \le \alpha \le N} g_\alpha, \qquad h = \sup_{1 \le \beta \le M} h_\beta$$

as in (11). For this situation, we use a different way of computing the maximum using ReLU nets.

Lemma 2. *There exists a* ReLU *net* $\mathcal{M}$ *with input dimension* 2, *hidden layer width* 2, *output dimension* 1, *and depth* 2 *such that:*

$$\mathcal{M}\,(x, y) = \max\{x, y\}, \qquad x \in \mathbb{R}, y \in \mathbb{R}_+.$$

Proof. Set $A_1(x, y) := (x - y, y)$, $A_2(z, w) = z + w$, and define:

$$\mathcal{M} = \text{ReLU} \circ A_2 \circ \text{ReLU} \circ A_1.$$

We have for each $y \ge 0, x \in \mathbb{R}$:

$$f_{\mathcal{M}}(x, y) = \text{ReLU}((x - y)\mathbf{1}_{\{x - y > 0\}} + y) = \max\{x, y\},$$

as desired. □

We now describe how to construct a ReLU net $\mathcal{N}$ with input dimension d, hidden layer width $d+3$, output dimension 1, and depth $2(M+N)$ that exactly computes f. We use width d to copy the input x, width 2 to compute successive maximums of the positive affine functions g_α, h_β using the net $\mathcal{M}$ from Lemma 2 above, and width 1 as memory in which we store $g = \sup_\alpha g_\alpha$ while computing $h = \sup_\beta h_\beta$. The final layer computes the difference $f = g - h$. □

5. Proof of Theorem 1

Proof of Theorem 1. We begin by showing (8) and (9). Suppose $f : [0,1]^d \to \mathbb{R}_+$ is convex, and fix $\varepsilon > 0$. A simple discretization argument shows that there exists a piecewise affine convex function $g : [0,1]^d \to \mathbb{R}_+$ such that $\|f - g\|_{C^0} \le \varepsilon$. By Theorem 2, g can be exactly represented by a ReLU net with hidden layer width $d+1$. This proves (8). In the case that f is Lipschitz, we use the following, a special case of Lemma 4.1 in [15].

Proposition 1. *Suppose* $f : [0,1]^d \to \mathbb{R}$ *is convex and Lipschitz with Lipschitz constant L. Then, for every* $k \ge 1$, *there exist k affine maps* $A_j : [0,1]^d \to \mathbb{R}$ *such that:*

$$\left\| f - \sup_{1 \le j \le k} A_j \right\|_{C^0} \le 72L\, d^{3/2} k^{-2/d}.$$

Combining this result with Theorem 2 proves (9). We turn to checking (5) and (10). We need the following observations, which seems to be well known, but not written down in the literature.

Lemma 3. *Let $\mathcal{N}$ be a* ReLU *net with input dimension d, a single hidden layer of width n, and output dimension* 1. *There exists another* ReLU *net $\widetilde{\mathcal{N}}$ that computes the same function as $\mathcal{N}$, but has input dimension d and $n+2$ hidden layers with width $d+2$.*

Proof. Denote by $\{A_j\}_{j=1}^n$ the affine functions computed by each neuron in the hidden layer of $\mathcal{N}$ so that:

$$f_{\mathcal{N}}(x) = \mathrm{ReLU}\left(b + \sum_{j=1}^{n} c_j\, \mathrm{ReLU}(A_j(x))\right).$$

Let $T > 0$ be sufficiently large so that:

$$T + \sum_{j=1}^{k} c_j\, \mathrm{ReLU}(A_j(x)) > 0, \qquad \forall 1 \le k \le n, \;\; x \in [0,1]^d.$$

The affine transformations $\widetilde{A}_j$ computed by the jth hidden layer of $\widetilde{\mathcal{N}}$ are then:

$$\widetilde{A}_1(x) := (x, A_j(x), T) \qquad \text{and} \qquad \widetilde{A}_{n+2}(x,y,z) = z - T + b, \qquad x \in \mathbb{R}^d,\, y,z \in \mathbb{R}$$

and:

$$\widetilde{A}_j(x,y,z) = \left(x, A_j(x), z + c_{j-1}y\right), \qquad j = 2,\ldots,n+1.$$

We are essentially using width d to copy in the input variable, width 1 to compute each A_j, and width 1 to store the output. □

Recall that positive continuous functions can be arbitrarily well approximated by smooth functions and hence by ReLU nets with a single hidden layer (see, e.g., Theorem 3.1 [5]). The relation (5) therefore follows from Lemma 3. Similarly, by Theorem 3.1 in [5], if f is smooth, then there exists $K = K(d) > 0$ and a constant C_f depending only on the maximum value of the first K derivatives of f such that:

$$\inf_{\mathcal{N}} \|f - f_{\mathcal{N}}\| \le C_f n^{-1/d},$$

where the infimum is over ReLU nets $\mathcal{N}$ with a single hidden layer of width n. Combining this with Lemma 3 proves (10).

It remains to prove (6) and (7). To do this, fix a positive continuous function $f : [0,1]^d \to \mathbb{R}_+$ with modulus of continuity ω_f. Recall that the volume of the unit d-simplex is $1/d!$, and fix $\varepsilon > 0$. Consider the partition:

$$[0,1]^d = \bigcup_{j=1}^{d!/\omega_f(\varepsilon)^d} \mathcal{P}_j$$

of $[0,1]^d$ into $d!/\omega_f(\varepsilon)^d$ copies of $\omega_f(\varepsilon)$ times the standard d-simplex. Here, each $\mathcal{P}_j$ denotes a single scaled copy of the unit simplex. To create this partition, we first sub-divide $[0,1]^d$ into at most $\omega_f(\varepsilon)^{-d}$ cubes of side length at most $\omega_f(\varepsilon)$. Then, we subdivide each such smaller cube into $d!$ copies of the standard simplex (which has volume $1/d!$) rescaled to have side length $\omega_f(\varepsilon)$. Define f_ε to be a piecewise linear approximation to f obtained by setting f_ε equal to f on the vertices of the $\mathcal{P}_j$'s and taking f_ε to be affine on their interiors. Since the diameter of each $\mathcal{P}_j$ is $\omega_f(\varepsilon)$, we have:

$$\|f - f_\varepsilon\|_{C^0} \le \varepsilon.$$

Next, since f_ε is a piecewise affine function, by Theorem 2.1 in [2] (see Theorem 2), we may write:

$$f_\varepsilon = g_\varepsilon - h_\varepsilon,$$

where $g_\varepsilon, h_\varepsilon$ are convex, positive, and piecewise affine. Applying Theorem 2 completes the proof of (6) and (7). □

6. Conclusions

We considered in this article the expressive power of ReLU networks with bounded hidden layer widths. In particular, we showed that ReLU networks of width $d + 3$ and arbitrary depth are capable of arbitrarily good approximations of any scalar continuous function of d variables. We showed further that this bound could be reduced to $d + 1$ in the case of convex functions and gave quantitative rates of approximation in all cases. Our results show that deep ReLU networks, even at a moderate width, are universal function approximators. Our work leaves open the question of whether such function representations can be learned by (stochastic) gradient descent from a random initialization. We will take up this topic in future work.

Funding: This research was funded by NSF Grants DMS-1855684 and CCF-1934904.

Acknowledgments: It is a pleasure to thank Elchanan Mossel and Leonid Hanin for many helpful discussions. This paper originated while I attended EM's class on deep learning [18]. In particular, I would like to thank him for suggesting proving quantitative bounds in Theorem 2 and for suggesting that a lower bound can be obtained by taking piece-wise linear functions with many different directions. He also pointed out that the width estimates for the continuous function in Theorem 1 were sub-optimal in a previous draft. I would also like to thank Leonid Hanin for detailed comments on several previous drafts and for useful references to the results in approximation theory. I am also grateful to Brandon Rule and Matus Telgarsky for comments on an earlier version of this article. I am also grateful to BR for the original suggestion to investigate the expressivity of neural nets of width two. I also would like to thank Max Kleiman-Weiner for useful comments and discussion. Finally, I thank Zhou Lu for pointing out a serious error what used to be Theorem 3 in a previous version of this article. I have removed that result.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bengio, Y.; Hinton, G.; LeCun, Y. Deep learning. *Nature* **2015**, *521*, 436–444.
2. Arora, R.; Basu, A.; Mianjy, P.; Mukherjee, A. Understanding deep neural networks with Rectified Linear Units. In Proceedings of the International Conference on Representation Learning, Vancouver, BC, Canada, 30 April 30–3 May 2018.
3. Liao, Q.; Mhaskar, H.; Poggio, T. Learning functions: When is deep better than shallow. *arXiv* **2016**, arXiv:1603.00988v4.
4. Lin, H.; Rolnick, D.; Tegmark, M. Why does deep and cheap learning work so well? *arXiv* **2016**, arXiv:1608.08225v3.
5. Mhaskar, H.; Poggio, T. Deep vs. shallow networks: An approximation theory perspective *Anal. Appl.* **2016**, *14*, 829–848. [CrossRef]
6. Poole, B.; Lahiri, S.; Raghu, M.; Sohl-Dickstein, J.; Ganguli, S. Exponential expressivity in deep neural networks through transient chaos. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3360–3368.
7. Raghu, M.; Poole, B.; Kleinberg, J.; Ganguli, S.; Dickstein, J. On the expressive power of deep neural nets. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 2847–2854.
8. Telgrasky, M. Representation benefits of deep feedforward networks. *arXiv* **2015**, arXiv:1509.08101.
9. Telgrasky, M. Benefits of depth in neural nets. In Proceedings of the JMLR: Workshop and Conference Proceedings, New York, NY, USA, 19 June 2016; Volume 49, pp. 1–23.
10. Telgrasky, M. Neural networks and rational functions. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 3387–3393.

11. Yarotsky, D. Error bounds for approximations with deep ReLU network. *Neural Netw.* **2017**, *94*, 103–114. [CrossRef] [PubMed]
12. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control. Signals Syst. (MCSS)* **1989**, *2*, 303–314. [CrossRef]
13. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *J. Neural Netw.* **1989**, *2*, 359–366 [CrossRef]
14. Hanin, B.; Sellke, M. Approximating Continuous Functions by ReLU Nets of Minimal Width. *arXiv* **2017**, arXiv:1710.11278.
15. Balázs, G.; György, A.; Szepesvári, C. Near-optimal max-affine estimators for convex regression. In Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, San Diego, CA, USA, 9–12 May 2015; Volume 38, pp. 56–64.
16. Rolnick, D.; Tegmark, M. The power of deeper networks for expressing natural functions. In Proceedings of International Conference on Representation Learning, Vancouver, BC, Canada, 30 April–3 May 2018.
17. Shang, Y. A combinatorial necessary and sufficient condition for cluster consensus. *Neurocomputing* **2016**, *216*, 611–616. [CrossRef]
18. Mossel, E. Mathematical Aspects of Deep Learning. Available online: http://elmos.scripts.mit.edu/mathofdeeplearning/mathematical-aspects-of-deep-learning-intro/ (accessed on 10 September 2019)