Proceedings of Machine Learning Research vol 65:1–5, 2017

Nearly-tight VC-dimension bounds for piecewise linear neural networks

Nick Harvey NICKHAR@CS.UBC.CA
Christopher Liaw CVLIAW@CS.UBC.CA
Abbas Mehrabian ABBASMEHRABIAN@GMAIL.COM
Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Abstract

We prove new upper and lower bounds on the VC-dimension of deep neural networks with the ReLU activation function. These bounds are tight for almost the entire range of parameters. Letting W be the number of weights and L be the number of layers, we prove that the VC-dimension is $O(WL\log(W))$, and provide examples with VC-dimension $\Omega(WL\log(W/L))$. This improves both the previously known upper bounds and lower bounds. In terms of the number U of non-linear units, we prove a tight bound $\Theta(WU)$ on the VC-dimension. All of these results generalize to arbitrary piecewise linear activation functions.

Keywords: VC-dimension, neural networks, ReLU activation function

Extended abstract. The full version with all the proofs appears as [arXiv:1703.02930, v2].

1. Introduction

Deep neural networks underlie many of the recent breakthroughs in applied machine learning, particularly in image and speech recognition. These successes motivate a renewed study of these networks' theoretical properties.

Classification is one of the learning tasks in which deep neural networks have been particularly successful, e.g., for image recognition. A natural foundational question that arises is: what are the generalization guarantees of these networks in a statistical learning framework? An established way to address this question is by considering VC-dimension, as it is well known that this asymptotically determines the sample complexity of PAC learning with such classifiers (Vapnik and Chervonenkis, 1971; Blumer et al., 1989).

In this paper, we prove nearly-tight bounds on the VC-dimension of deep neural networks in which the non-linear activation function is a piecewise linear function with a constant number of pieces. For simplicity we will henceforth refer to such networks as "piecewise linear networks". The most common activation function used in practice is, by far, the *rectified linear unit*, also known as *ReLU* (Goodfellow et al., 2016; LeCun et al., 2015). The ReLU function is defined as $\sigma(x) = \max\{0, x\}$, so it is clearly piecewise linear.

It is particularly interesting to consider how the VC-dimension is affected by the various attributes of the network: the number W of parameters (i.e., weights and biases), the number U of non-linear units (i.e., nodes), and the number L of layers. Among all networks with the same size (number of weights), is it true that those with more layers have larger VC-dimension?

Such a statement is indeed true, and previously known; however, a tight characterization of how depth affects VC-dimension was unknown prior to this work.

Our results. For the definitions of VC-dimension, shattering etc., we refer the reader to Anthony and Bartlett (1999). Our first main result is a new VC-dimension lower bound that holds even for the restricted family of ReLU networks.

Theorem 1 (Main lower bound) *There exists a universal constant C such that the following holds. Given any W, L with $W > CL > C^2$, there exists a ReLU network with $\leq L$ layers and $\leq W$ parameters with VC-dimension $\geq WL\log_2(W/L)/C$.*

Remark. By a rough calculation, we can take $C = 640$ but we have not tried to optimize this constant.

Remark. Our construction can be augmented slightly to give a neural network with linear threshold and identity activation functions with the same guarantees.

Prior to our work, the best known lower bounds were $\Omega(WL)$ (Bartlett et al., 1998, Theorem 2) and $\Omega(W\log W)$ (Maass, 1994, Theorem 1). We strictly improve both bounds to $\Omega(WL\log(W/L))$.

Our proof of Theorem 1 uses the "bit extraction" technique that was developed by Bartlett et al. (1998) to prove an $\Omega(WL)$ lower bound. We refine their technique in a key way — we partition the input bits into blocks and extract multiple bits at a time instead of a single bit at a time. This yields a more efficient bit extraction network, which gains us the additional logarithmic factor that appears in Theorem 1.

Unfortunately there is a barrier to refining this technique any further. Our next theorem shows the hardness of computing the mod function, implying that the bit extraction technique cannot yield a stronger lower bound than Theorem 1. Further discussion of this connection may be found in the full version.

Theorem 2 *Assume there exists a piecewise linear network with W parameters and L layers that computes a function $f : \mathbb{R} \to \mathbb{R}$, with the property that $|f(x) - (x \bmod 2)| < 1/2$ for all $x \in \{0, 1, \ldots, 2^m - 1\}$. Then we have $m = O(L\log(W/L))$.*

One interesting aspect of the proof is that it does not use Warren's lemma (Warren, 1968), which is a mainstay of VC-dimension upper bounds, see Goldberg and Jerrum (1995); Bartlett et al. (1998); Anthony and Bartlett (1999).

Telgarsky (2016) showed how to construct a function f which satisfies $f(x) = (x \bmod 2)$ for $x \in \{0, 1, \ldots, 2^m - 1\}$ using a neural network with $O(m)$ layers and $O(m)$ parameters. By choosing $m = k^3$, Telgarsky showed that any function g computable by a neural network with $\Theta(k)$ layers and $O(2^k)$ nodes must necessarily have $\|f - g\|_1 > c$ for some constant $c > 0$. Our Theorem 2 implies a similar statement. In particular, if we choose $m = k^{1+\varepsilon}$ then for any function g computable by a neural network with $\Theta(k)$ layers and $O(2^{k^\varepsilon})$ parameters, there must exist $x \in \{0, 1, \ldots, 2^m - 1\}$ such that $|f(x) - g(x)| > 1/2$.

Our next main result is an upper bound on the VC-dimension of neural networks with any piecewise linear activation function with a constant number of pieces. Recall that ReLU is an example of a piecewise linear activation function.

Theorem 3 (Main upper bound) *Consider a piecewise linear neural network with W parameters arranged in L layers. Let $\mathcal{F}$ be the set of (real-valued) functions computed by this network. Then, if $m = \mathrm{VCDim}(\mathrm{sgn}(\mathcal{F}))$ and p is the number of pieces of the activation function, it holds that $m \leq 4W(L+1)\log_2(2eWmp)$. Asymptotically, this implies $\mathrm{VCDim}(\mathrm{sgn}(\mathcal{F})) = O(WL\log W)$.*

Remark. After the initial version of this paper was submitted, we have learned that a similar upper bound of $O(WL \log W)$ for the VC-dimension of piecewise linear neural networks was proved independently by Bartlett (2017).

Prior to our work, the best published upper bounds were $O(W^2)$ (Goldberg and Jerrum, 1995, Section 3.1) and $O(WL \log W + WL^2)$ (Bartlett et al., 1998, Theorem 1), both of which hold for piecewise polynomial activation functions; we strictly improve both bounds to $O(WL \log W)$ for the special case of piecewise linear functions.

The proof of Theorem 3 is very similar to the proof of the upper bound for piecewise polynomial networks in (Bartlett et al., 1998, Theorem 1) but optimized for piecewise linear networks. In particular, we use a result of Warren (1968).

To compare our upper and lower bounds, let $d(W, L)$ denote the largest VC-dimension of a piecewise linear network with W parameters and L layers. Theorems 1 and 3 imply there exist constants c, C such that

$$c \cdot WL \log(W/L) \;\le\; d(W, L) \;\le\; C \cdot WL \log W \,. \tag{1}$$

For neural networks arising in practice it would certainly be the case that L is significantly smaller than $W^{0.99}$, in which case our results determine the asymptotic bound $d(W, L) = \Theta(WL \log W)$. On the other hand, in the regime $L = \Theta(W)$, which is merely of theoretical interest, we also now have a tight bound $d(W, L) = \Theta(WL)$, obtained by combining Theorem 1 with results of Goldberg and Jerrum (1995). There is now only a very narrow regime, say $W^{0.99} \ll L \ll W$, in which the bounds of (1) are not asymptotically tight, and they differ only in the logarithmic factor.

Our final result is an upper bound for VC-dimension in terms of W and U (the number of non-linear units, or nodes). It follows from our proof of Theorem 1 that this bound is tight in the case $d = 1$.

Theorem 4 *Consider a neural network with W parameters and U units with activation functions that are piecewise polynomials of degree at most d. Let $\mathcal{F}$ be the set of (real-valued) functions computed by this network. Then* $\mathrm{VCDim}(\mathrm{sgn}(\mathcal{F})) = O(WU \log(d + 1))$.

The proof of this result appears in the full version. The best known upper bound before our work was $O(W^2)$, implicitly proven for fixed d in (Goldberg and Jerrum, 1995, Section 3.1). Our theorem improves this to the tight result $O(WU)$.

The idea of the proof of Theorem 4 is that the sign of the output of a neural network can be expressed as a Boolean formula where each predicate is a polynomial inequality. Then we apply (Goldberg and Jerrum, 1995, Theorem 2.2), which gives a bound for the VC-dimension of a class of Boolean functions that can be expressed using distinct polynomial inequalities.

The proofs of all our results appear in the full version of this paper, see Harvey et al. (2017).

Related Work. The VC-dimension of neural networks with *linear threshold functions* has been studied many years ago. Cover (1968) and later (Baum and Haussler, 1989, Corollary 2) proved that their VC-dimension is $O(W \log W)$. The $\Omega(W \log W)$ lower bound of Maass (1994) mentioned above actually apply to the restricted setting of linear threshold functions, hence giving a tight bound of $\Theta(W \log W)$ for such networks. For neural networks with linear threshold and identity activation functions, Bartlett et al. (1998) proved a lower bound of $\Omega(WL)$. The lower bounds of Maass (1994) and Bartlett et al. (1998) also hold for our scenario of ReLU networks since linear threshold

functions can easily be simulated using ReLU functions. We refer the reader to the excellent monograph by Anthony and Bartlett (1999) that covers these and many other theoretical results on neural networks.

Recently there have been several papers that study neural networks from an approximation theory point of view and aim to understand which functions can be expressed using a neural network of given a depth and size. There are technical similarities between our work and these. Last year, two striking papers considered the problem of approximating a deep neural network with a shallower network. Telgarsky (2016) shows that there is a ReLU network with L layers and $U = \Theta(L)$ units such that any network approximating it with only $O(L^{1/3})$ layers must have $\Omega(2^{L^{1/3}})$ units; this phenomenon holds even for real-valued functions. Eldan and Shamir (2016) show an analogous result for a high-dimensional 3-layer network that cannot be approximated by a 2-layer network except with an exponential blow-up in the number of nodes.

Very recently, several authors have shown that deep neural networks are capable of approximating broad classes of functions. Safran and Shamir (2017) show that a sufficiently non-linear C^2 function on $[0,1]^d$ can be approximated with ϵ error in L_2 by a ReLU network with $O(\text{polylog}(1/\epsilon))$ layers and weights, but any such approximation with $O(1)$ layers requires $\Omega(1/\epsilon)$ weights. Yarotsky (2017) shows that any C^n-function on $[0,1]^d$ can be approximated with ϵ error in L_∞ by a ReLU network with $O(\log(1/\epsilon))$ layers and $O((\frac{1}{\epsilon})^{d/n}\log(1/\epsilon))$ weights. Liang and Srikant (2017) show that a sufficiently smooth univariate function can be approximated with ϵ error in L_∞ by a network with ReLU and threshold gates with $\Theta(\log(1/\epsilon))$ layers and $O(\text{polylog}(1/\epsilon))$ weights, but that $\Omega(\text{poly}(1/\epsilon))$ weights would be required if there were only $o(\log(1/\epsilon))$ layers; they also prove analogous results for multivariate functions. Lastly, Cohen et al. (2016) draw a connection to tensor factorizations to show that, for non-ReLU networks, the set of functions computable by a shallow network have measure zero among those computable by a deep networks.

Acknowledgments

CL is supported by an NSERC graduate scholarship. AM is supported by an NSERC Postdoctoral Fellowship and a Simons-Berkeley Research Fellowship. Part of this work was done while he was visiting the Simons Institute for the Theory of Computing at UC Berkeley.

References

Martin Anthony and Peter Bartlett. *Neural network learning: theoretical foundations*. Cambridge University Press, 1999.

Peter Bartlett. The impact of the nonlinearity on the VC-dimension of a deep network, 2017. unpublished manuscript.

Peter Bartlett, Vitaly Maiorov, and Ron Meir. Almost linear VC-dimension bounds for piecewise polynomial networks. *Neural Computation*, 10(8):2159–2173, Nov 1998.

Eric B. Baum and David Haussler. What size net gives valid generalization? *Neural Computation*, 1(1):151–160, 1989.

A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *JACM*, 36(4), 1989. (Conference version in STOC'86).

N. Cohen, O. Sharir, and A. Shashua. On the expressive power of deep learning: A tensor analysis. In *COLT*, 2016.

Thomas M. Cover. Capacity problems for linear machines. In L. Kanal, editor, *Pattern Recognition*, pages 283–289. Thompson Book Co., 1968.

Ronen Eldan and Ohad Shamir. The power of depth for feedforward neural networks. In *COLT*, 2016.

Paul W. Goldberg and Mark R. Jerrum. Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers. *Machine Learning*, 18(2):131–148, 1995. (Conference version in COLT'93).

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Nick Harvey, Christopher Liaw, and Abbas Mehrabian. Nearly-tight VC-dimension bounds for piecewise linear neural networks, 2017. URL https://arxiv.org/abs/1703.02930.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

Shyu Liang and R. Srikant. Why deep neural networks for function approximation?, 2017. arXiv:1610.04161.

Wolfgang Maass. Neural nets with superlinear VC-dimension. *Neural Computation*, 6(5):877–884, Sept 1994.

I. Safran and O. Shamir. Depth-width tradeoffs in approximating natural functions with neural networks, 2017. arXiv:1610.09887.

Matus Telgarsky. Benefits of depth in neural networks. In *COLT*, 2016.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971. doi: 10.1137/1116025.

Hugh E. Warren. Lower bounds for approximation by nonlinear manifolds. *Transactions of the American Mathematical Society*, 133(1):167–178, 1968.

Dmitry Yarotsky. Error bounds for approximations with deep ReLU networks, 2017. arXiv:1610.01145.