Benefits of depth, part 2

To recap, the last lecture ended with the following function $h : \mathbb{R} \to \mathbb{R}$, defined as

$$h(x) := \sigma\left(\sigma(2x) - \sigma(4x-2)\right) = \begin{cases} 2x & \text{when } x \in [0, 1/2], \\ 2(1-x) & \text{when } x \in (1/2, 1], \\ 0 & \text{when } x \notin [0,1], \end{cases}$$

where $\sigma(x)$ is the ReLU $x \mapsto \max\{0, x\}$ for this and the preceding lecture.

Taken by itself, h is just a bump *(picture drawn in class)*. Where h becomes special is when it is composed; for instance, h^t looks like 2^{t-1} copies of h laid side-by-side and compressed to fit in $[0, 1]$. This is captured in the following lemma, proved last time.

Lemma. For any integer $i \in \{0, \dots, 2^{t-1} - 1\}$ and any real $x \in [0, 1]$, h^t satisfies

$$h^t\left(\frac{1}{2^{t-1}}(x+i)\right) = h(x).$$

If we were to try to create h^t using linear combinations, we'd have to do

$$x \mapsto \sum_{i=0}^{2^{t-1}} h\left(\frac{1}{2^{t-1}}(x+i)\right);$$

in particular, we'd use $3 \cdot 2^{t-1}$, whereas h^t uses $3t$ nodes. So h, though humble, is able to demonstrate the power of composition and depth.

This argument just constructed one function with many bumps via a compact representation. It still needs to be argued that shallow networks don't have any way to approximate these functions well. The next key is the following definition and lemma.

Say that a univariate function $f : \mathbb{R} \to \mathbb{R}$ is *t-affine* if there exists a partition of $\mathbb{R}$ into t intervals such that for each interval, there exists an affine function that is equal to f along that interval.

The notion t-affine will be the way we track complexity and bumpiness of univariate functions. The following lemma is essential, and the key conclusion is that combining nodes in linear combinations slowly grows complexity, whereas composing them quickly grows complexity.

Lemma.

1. If f is t-affine and $c \in \mathbb{R}$, then cf and $f + c$ are t-affine.
2. If f and g are s- and t-affine, then $f + g$ is $(s + t - 1)$-affine.
3. If $(f_1, \dots, f_r)$ are t-affine and $(a_0, \dots, a_r)$ are real scalars, then $a + 0 + \sum_{i=1}^{r} a_i f_i$ is (tr)-affine.
4. If f and g are s- and t-affine, then $f \circ g$ is (st)-affine.
5. If $f : \mathbb{R} \to \mathbb{R}$ is a neural network with l layers, N nodes, and t-affine nonlinearities σ, then f is $(tN/l)^l$-affine.

Remark. *This* proof, out of all the steps to the final theorem, is what breaks for sigmoids.

Proof.

1. Given a partition $\mathcal{I}_f$ corresponding to f, for any interval $J \in \mathcal{I}_f$, the affine function h equal to f along J is still affine after the transformations ch and $c+h$. Thus $c+f$ and cf are t-affine, indeed with the same partition $\mathcal{I}_f$.

2. Let $(a_1, \ldots, a_s)$ and $(b_1, \ldots, b_t)$ be the sequence of right-endpoints of intervals defininining f and g, where $a_s = \infty = b_t$. Combining and sorting these endpoints gives a sequence $(c_1, \ldots, c_l)$, where $c_l = \infty$ and $l \leq s + t - 1$; considering the corresponding collection of l intervals defined by these endpoints, *both* f and g must be affine within each interval; said another way, this collection of l intervals is the set of all intersections of intervals defining f and g. Since the sum of two affine functions is affine, then $f+g$ is affine in each of these l pieces.

3. Define $g_1 := a_0 + a_1 f_1$, and $g_i := a_i f_i$ for $i \geq 2$, whereby $a_0 + \sum_i a_i f_i = \sum_i g_i$. By the first point of this lemma, g_i is t-affine for each i. Applying the second point inductively, $\sum_i g_i$ is (rt)-affine.

4. Let $\mathcal{I}_f$ and $\mathcal{I}_g$ denote the partitions defining f and g, respectively. Fix any $J \in \mathcal{I}_g$, and let h denote the fixed affine function computed by g along J. It will be shown that J can be partitioned into at most s pieces so that $f \circ g$ is affine along each; since $J \in \mathcal{I}_f$ was arbitrary, it follows that $\mathbb{R}$ can be partitioned into $\leq st$ intervals along which $f \circ g$ is affine.

   To this end, returning to $J \in \mathcal{I}_g$ and a corresponding affine h, consider two cases. If h has slope 0, then $h(J)$ is constant, thus $f(h(J))$ is also constant, meaning $f \circ h$ is affine along J. Otherwise, h has nonzero slope, and define

   $$S := \left\{ h(J) \cap J' : J' \in \mathcal{I}_f \right\}.$$

   Since h is affine, $h(J)$ is an interval, and thus S is a partition of $h(J)$ into intervals. Since h has nonzero slope, it is bijective and in particular has an inverse h^{-1}, so define

   $$S_J := \left\{ J \cap h^{-1}(J'') : J'' \in S \right\}.$$

   Since h is affine and bijective, h^{-1} is affine, thus $h^{-1}(J'')$ is an interval for each interval $J'' \in S$; since S partitions $h(J)$, then S_J must partition J. Lastly, $|S_J| = |S| = |\mathcal{I}_f| \leq s$.

5. Let $N_0, \ldots, N_l$ denote the numbers of nodes per layer, and let $g_i : \mathbb{R} \to \mathbb{R}^{N_i}$ denote the collection of all outputs from layer i. The proof will proceed by induction, showing each coordinate of g_i is $t^i \prod_{j=1}^{i-1} N_i$-affine. To see that this implies the final result, note by Jensen's inequality (which will be covered in the upcoming convexity lectures) that

   $$\begin{aligned} \prod_{j=1}^{i-1} N_j \leq \prod_{j=1}^{i} N_j &= \exp\left(\sum_{j=1}^{i} \ln(N_j) \right) = \exp\left(\frac{l}{l} \sum_{j=1}^{i} \ln(N_j) \right) \\ &\leq \exp\left(l \ln\left(\sum_{j=1}^{i} (N_j/l) \right) \right) = \exp\left(l \ln(N/l) \right) = \left(\frac{N}{l} \right)^l. \end{aligned}$$

   Returning to the induction, the base case being layer 1 (not layer 0). Viewing the output of layer 1 as $\vec{\sigma}(A_1 x + b_1)$, Note that $x \mapsto (A_1 x + b_1)_i$ for each coordinate i is 1-affine. By the earlier product rule, composing each coordinate with t-affine σ means each coordinate of $\vec{\sigma}(A_1 x + b_1)$ is t-affine.

   Now consider layer $i+1$ with $i \geq 1$. By the inductive hypothesis, each coordinate of g_i is $(t^i \prod_{j=1}^{i-1} N_j)$-affine, so by the earlier affine combination rule, the map $x \mapsto A_{i+1} g_i(x) + b_{i+1}$ is $(N_i t^i \prod_{j=1}^{i-1} N_j)$-affine in each coordinate, and so $g_{i+1} = \vec{\sigma}(A_{i+1} g_i(x) + b_{i+1})$ is $(t^{i+1} \prod_{j=1}^{i} N_j)$-affine in each coordinate. $\square$

So we've constructed a well-behaved, highly-bumpy function in many layers, and upper bounded the bumpiness of general functions. What remains is to show that these properties imply an approximation error.

Theorem (restated from last lecture)**.** Let any integer $k \geq 1$ be given, and let S_k denote those ReLU networks mapping $\mathbb{R}$ to $\mathbb{R}$ with $\leq k$ layers and $\leq 2^k$ nodes. Then there exists a function $g : \mathbb{R} \to \mathbb{R}$ representable as a network with $2(k^2+3)$ layers and $3(k^2+3)$ nodes so that

$$\inf_{f \in S_k} \|f - g\|_1 \geq \frac{1}{32}.$$

Proof. Define $g := h^{k^2+3}$, which by the earlier lemma consists of 2^{k^2+2} shrunken copies of h. Let any $f \in S_k$ be given, which by the earlier lemmas is $(2 \cdot 2^k/k)^k \leq 2^{k^2+1}$-affine. (by considering the cases $k = 1$ and $k > 1$ separately).

(Picture developed in class based on the following idea.) The idea of the proof is as follows. Consider functions mapping from $[0, 1]$ to $\mathbb{R}$, and draw the horizontal line through $1/2$. Keeping in mind that f is piecewise affine, it crosses the horizontal line at most 2^{k^2+1} times; let $\mathcal{C}_f$ denote the partition of $[0, 1]$ constructed from these crossing points, meaning $s := |\mathcal{C}_f| == 1 + 2^{k^2+1}$, and for each interval within $\mathcal{C}_f$, f resides on a single side of the horizontal line.

Similarly, $g = h^{k^2+3}$ crosses the line 2^{k^2+4} times (twice per copy of h). In turn, this gives a partition $\mathcal{C}_g$ of $[0, 1]$ into $t := |\mathcal{C}_g| = 1 + 2^{k^2+4}$ intervals. Moreover, for any interval within $\mathcal{C}_g$, the shape formed by g and the horizontal line is a triangle, and these triangles alternate between appearing above and below the horizontal line. Each triangle has width at least $1/(2t)$ and thus total area at least $1/(8t)$.

The argument is now as follows. Since the triangles given by g keep switching between below and above the horizontal lane, then any interval of $\mathcal{C}_f$ containing many intervals of $\mathcal{C}_g$ must have many triangles pointing the opposite way of f, which is on a single side of the horizontal line for a fixed interval of $\mathcal{C}_f$. More precisely, given any interval $J \in \mathcal{C}_f$, let $X_J := \{J' \in \mathcal{C}_g : J' \subseteq J\}$ denote all intervals of $\mathcal{C}_g$ complained contained within J. At least $(|X_J| - 1)/2$ of these intervals must have triangles pointing the wrong way; as such,

$$\int_J |f(x) - g(x)|dx \geq \sum_{I \in X_J} |f(x) - g(x)|dx \geq \frac{1}{8t}\left(\frac{|X_J| - 1}{2}\right).$$

Next, note that every interval in $\mathcal{C}_g$ lands in X_J for some J, or it contains an (internal!) endpoint of an interval in $\mathcal{C}_f$; thus

$$t \leq \sum_{J \in \mathcal{C}_f} |X_J| + s.$$

Putting these together,

$$\begin{aligned}
\int_{[0,1]} |f(x) - g(x)|dx = \sum_{J \in \mathcal{C}_f} \int_J |f(x) - g(x)|dx &\geq \sum_{J \in \mathcal{C}_f} \frac{1}{8t}\left(\frac{|X_J| - 1}{2}\right). = \frac{1}{16t}\left(\sum_{J \in \mathcal{C}_f} (|X_J| - 1)\right) \\
&\geq \frac{1}{16t}(t - s - s) = \frac{1}{16}\left(1 - \frac{2s}{t}\right) = \frac{1}{16}\left(1 - \frac{2 + 2^{k^2+2}}{1 + 2^{k^2+4}}\right) \\
&\geq \frac{1}{32}.
\end{aligned}$$

□

(To close the lecture, there was some discussion of convolutional networks, but the main material was not presented, and will be returned to.)