Lecture 10: architectural benefits

Announcements:
* Next 5 lectures on zoom.
* Hw2 probably out Oct 2.

original material, now "open questions"

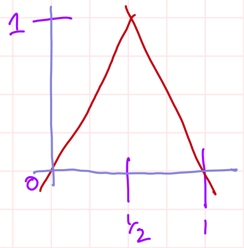
1. Architectural benefits of convolution & attention layers.
2. benefits of activation choice
3. Modeling distributions
4. low norm approximation
5. effect of depth [non-algorithmic]
   (i) polynomial networks (sum-product networks)
   (ii) many layers to fewer layers
   (iii) depth 3 vs depth 2

Many-to-few layers

We'll construct efficient approximators with many layers; warning: algorithmic relevance unknown.

also: poor scaling with dimension

All based on "triangle mapping $\Delta : \mathbb{R} \to \mathbb{R}$"

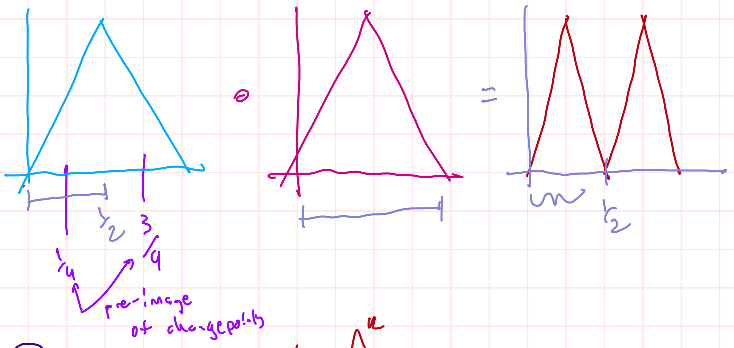


$$\Delta(x) = \begin{cases} 2x & x \in [0, \tfrac12) \\ 2(1-x) & x \in [\tfrac12, 1) \\ 0 & \text{otherwise} \end{cases}$$

$$= 2\sigma(x) - 4\sigma(x - \tfrac12) + \sigma(x)$$

[2-layer, 3-node ReLU network.]

$(\Delta \circ \Delta)(x)$
$= \Delta(\Delta(x))$



$\Delta^{k+1}(x) =$
$= \Delta^k(\Delta(x))$

2^k copies of Δ^k

Remark: Approximating this with shallow network needs exponentially many nodes, and we used 3k nodes.

One attempt to capture fractal structure:

Proposition: "fractal property" $\forall x \in [0,1]$
$$\Delta^L(x) = \Delta\left(2^{L-1}x - \lfloor 2^{L-1}x \rfloor\right) = \Delta(\langle 2^{L-1}x \rangle)$$
"fractional part"

(2^{L-1} side by side then squeezed.)

Proof. Induction on #layers i: $\left(\forall x \in [0,1]\ \Delta^i(x) = \Delta(\langle 2^{i-1}x \rangle)\right) \Rightarrow \left(\forall x \in [0,1]\ \Delta^{i+1}(x) = \Delta(\langle 2^i x \rangle)\right)$

not on outside

Base case $\Delta^1(x) = \Delta(\langle 2^{1-1}x \rangle)$ ✓.

IH: Let $x \in [0,1]$ & target depth $i+1$ be given, assume IH

$x \in [0, \tfrac12)$
$$\Delta^{i+1}(x) = \Delta^i(\Delta(x)) = \Delta^i(2x) = \Delta(\langle 2^{i-1} \cdot 2x \rangle)$$
$$= \Delta(\langle 2^i x \rangle).$$

$x \in (\tfrac12, 1]$. if $i=1$, $\Delta^0(x) = x$
$$\Delta^{i+1}(x) = \Delta^{i-1}(\Delta(\Delta(x))) = \Delta^{i-1}(\Delta(2(1-x)))$$
(using $\Delta(y) = \Delta(1-y)$) $= \Delta^{i-1}(\Delta(1 - 2(1-x))) = \Delta^{i-1}(\Delta(2x-1))$
$$= \Delta^i(2x-1) = \Delta(\langle 2^{i-1}(2x-1) \rangle) = \Delta(\langle 2^i x \rangle).$$
//

Applications of Δ^k.

① Approximate x^2 and xy up to accuracy ε with $\ln(1/\varepsilon)^2$ nodes & layers.
$\Rightarrow$ polynomials $\Rightarrow$ smooth functions.

② Parity function on d bits: $x \in \{\pm 1\}^d$, $\prod_{i=1}^{d} x_i = \Delta^{L-1}\left(\frac{d + \sum x_i}{2d}\right)$. [$d = 2^L$.]

③ If $f(z) = f(1-z)$ $\forall z \in [0,1]$, then $f(\Delta^{L+1}(x)) = f(\Delta(\langle 2^L x \rangle))$, 2^{L+1} copies of f ! } "viral fractal property"

④ In particular, can use ③ to extract bits

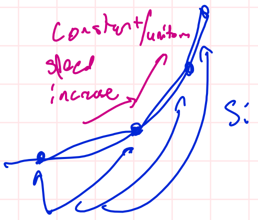
Approximating x^2 via Δ

$\forall x \geq 0$ $$x^2 = \int_0^\infty 2\,\underbrace{\sigma(x-b)}_{\text{ReLU}}\,db$$: places ReLUs uniformly;

hope of enough uniformity for Δ^L.

Consider progressive affine approximations of x^2:

$\forall i$ $S_i = \left\{ \frac{0}{2^i}, \frac{1}{2^i}, \frac{2}{2^i}, \dots, \frac{2^i}{2^i} \right\}$, $h_i :=$ affine interpolation of x^2 along S_i



$=$ given $x \in [0,1]$, $\exists\, \tau \in [0,1]$ & $k \in 0, \dots, 2^i - 1$ s.t. $x = \frac{k+\tau}{2^i}$,

$$h_i(x) = (1-\tau)\left(\frac{k}{2^i}\right)^2 + \tau\left(\frac{k+1}{2^i}\right)^2$$

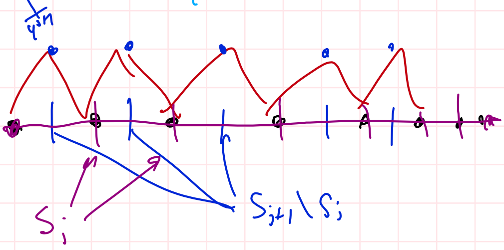
Note $$h_i(x) = h_0(x) + \sum_{j<i} \left(h_{j+1}(x) - h_j(x) \right)$$
$$= x + \sum_{j<i} \left(h_{j+1}(x) - h_j(x) \right).$$

Consider $h_{j+1}(x) - h_j(x)$

* case 1 $x \in S_j$: Since $S_{j+1} \supseteq S_j$, $h_{j+1}(x) - h_j(x) = x^2 - x^2 = 0$.
* case 2 $x \in S_{j+1} \setminus S_j$: pick k integer s.t. $x = \frac{2k+1}{2^{j+1}}$,

$$h_{j+1}(x) - h_j(x) = \left(\frac{2k+1}{2^{j+1}}\right)^2 - \frac{1}{2}\left(\left(\frac{2k}{2^{j+1}}\right)^2 + \left(\frac{2k+2}{2^{j+1}}\right)^2 \right)$$
$$= \frac{1}{4^{j+1}} \left(4k^2 + 4k + 1 - \frac{1}{2}\left(4k^2 + 4k^2 + 8k + 4 \right) \right)$$
$$= -\frac{1}{4^{j+1}}$$

* case 3 $x \in [0,1] \setminus S_{j+1}$:



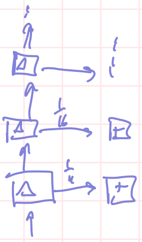
$$h_{j+1}(x) - h_j(x) = \frac{-\Delta^j(x)}{4^{j+1}}$$

$$\Rightarrow \quad h_i(x) = x - \sum_{j=0}^{i-1} \frac{\Delta^{j+1}(x)}{4^{j+1}}.$$

Theorem. Let h_i be the piecewise affine interpolation of x^2 along S_i.

1. h_i can be written as ReLU network with $3i$ nodes in $2i$ layers.
2. $\sup_{x \in [0,1]} \left| h_i(x) - x^2 \right| = \frac{1}{4^{i+1}}$.

Proof 1. what we had before, but reuse Δ:



2. Basically same proof as $h_i(x) - h_{i+1}(x)$, see notes.

//

Remark: 1. need $O(\ln(1/\varepsilon))$ layers & nodes to get ε accuracy

2. Multiplication via polarization identity: $xy = \frac{1}{2}\left((x+y)^2 - x^2 - y^2 \right)$.

$\rightarrow$ polynomials $\rightarrow \dots$