Is Deeper Better Only When Shallow Is Good?

Eran Malach and Shai Shalev-Shwartz

School of Computer Science, The Hebrew University, Israel

Depth Separation

There exist functions which can be expressed efficiently by a deep network but require an exponential width in order to be expressed by a shallow network

Motivation

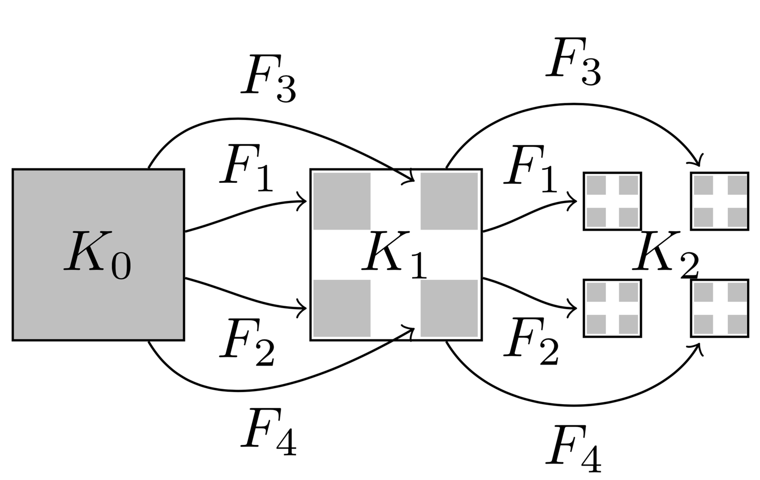
- Basic question: on which distributions deeper networks are much better than shallow ones?
- Various works show **depth separation** functions. (Telgarsky 2015, Safran and Shamir 2016, Cohen et al 2016, Daniely 2017, Poggio et al 2017).
- But, can such functions be learned efficiently using gradient-descent?
- We show that in some cases **strong** depth separation implies that gradient-descent **fails**.

IFS and Fractal Distributions

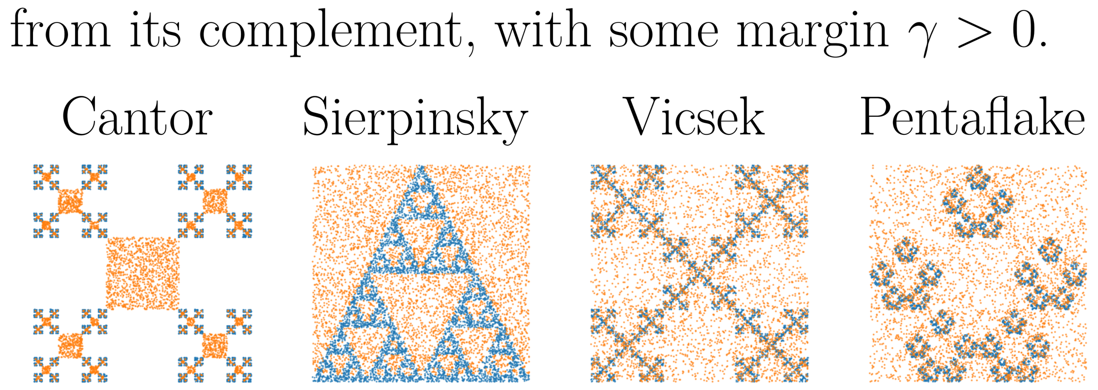
- Iterated Function System:

$$K_0 = [-1,1]^d$$
$$K_n = F_1(K_{n-1}) \cup \ldots \cup F_r(K_{n-1})$$

- We assume F_i are affine, invertible, contractive with disjoint images.
- The "depth" of the fractal is n
- Example: $F_i(x) = c_i + \frac{1}{4}(x - c_i)$ for $c_i \in \{\pm 1\}^2$



- We use the fractal set K_n to generate a binary classification problem.
- A **fractal distribution** D_n is a distribution in which positive examples sampled from the set K_n and negative examples are sampled from its complement, with some margin $\gamma > 0$.



Depth Separation of Fractals

We show that a network of depth $O(n)$ can express a depth n fractal, but a shallower network requires exponential width:

Theorem 1 *Consider an IFS over $[-1,1]^d$ with r generating functions and depth n. For any fractal distribution D_n there exists a ReLU network of depth $2n+1$ and width $5dr$ which realizes D_n.*

Proof by induction:

- A ReLU network can approximate $I_0(x) = 1_{x \in K_0}$
- Assume we expressed: $I_{n-1}(x) = 1_{x \in K_{n-1}}$
- Then, we can express: $1_{x \in K_n} = \vee_i I_{n-1}(F_i^{-1}(x))$

Theorem 2 *If D_n has non-zero probability in any area of K_n, then a network of depth t must have width $\geq \frac{d}{e} r^{n/td}$ to realize D_n.*

Proof idea:

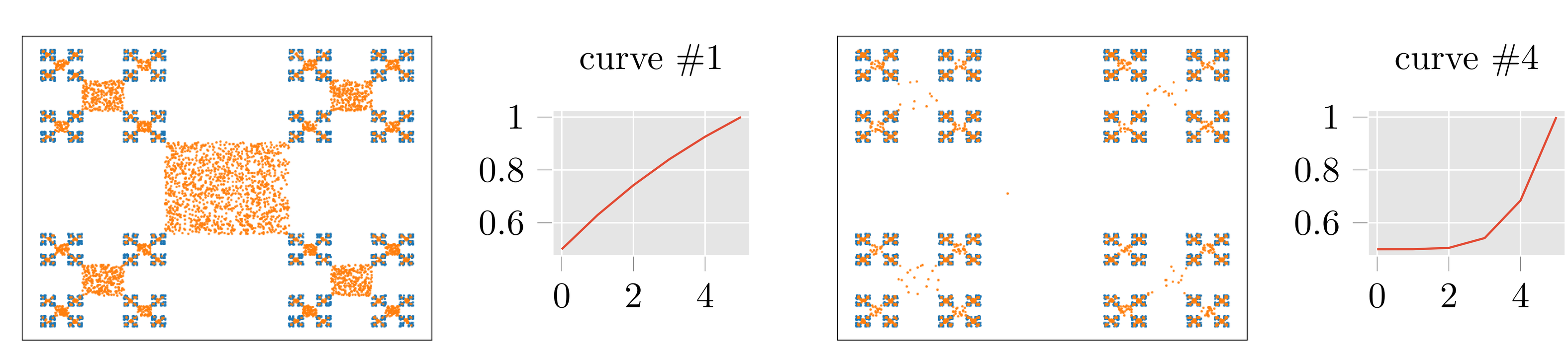
- A network of width k and depth t has at most $(ek/d)^{td}$ linear regions
- To realize D_n we need r^n linear regions

Approximation Curve

- **Approximation curve:** How much of the negative examples are on the fine details of the fractal:

$$P(j) := 1 - L_{D_n}(1_{x \in K_j}) = 1 - \mathbb{P}_{(x,y) \sim D_n}[x \in K_j \wedge y = -1]$$

- Note: $P(0) = 1/2$, $P(n) = 1$, and P is monotonically increasing



The following theorem shows that with reasonable width, the error of a depth $\Theta(j)$ network is roughly $1 - P(j)$:

Theorem 3 *Fix a depth n distribution with approx. curve P. Denote $L_{D_n}(H_{t,k})$ the loss of depth-t width-k nets. Then, for all j:*
1) For a depth $t = 2j+2$ and width $k = 5dr$ network we have

$$L_{D_n}(H_{t,k}) \leq (1 - P(j))$$

2) (for $d = 1$) For every s, if $k < r^s$ and $t < j/s$ then

$$(1 - r^{st-j})(1 - P(j)) \leq L_{D_n}(H_{t,k})$$

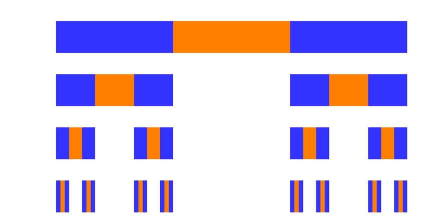
$\Rightarrow$ if $P(j) = \frac{1}{2}$ for every $j < d$ then every shallow network has loss $\gtrsim \frac{1}{2}$

Strong Depth Separation

There exists a distribution on which a small deep network gets loss 0, but a small shallow network gets loss $\gtrsim \frac{1}{2}$

Gradient Descent and Approximation Curve

Cantor distribution:
$F_1(x) = \frac{1}{3} - \frac{1}{3}x$ and $F_2(x) = \frac{2}{3} + \frac{1}{3}x$



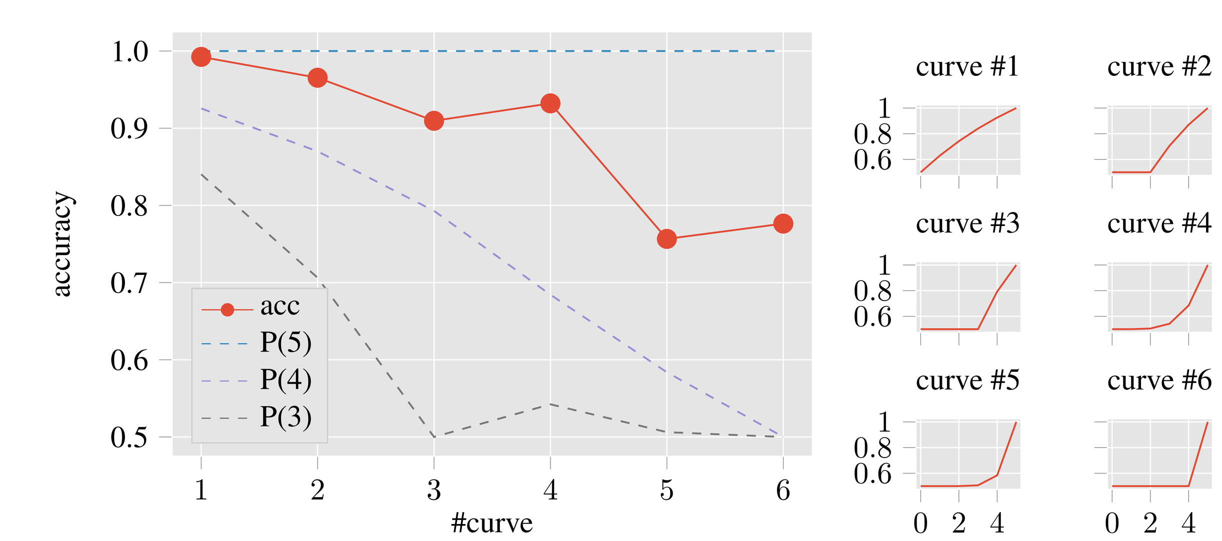
We show that the approximation curve controls the **optimization**:

Theorem 4 *Consider a depth t, width k, network, and suppose the weights, W, are initialized randomly in the "normal" way. Consider a depth n, one-dimensional Cantor fractal, and let $j = \lceil \log(tk^2/\delta) \rceil$. Then, with probability $> 1 - \delta$, all elements of the gradient at W are of magnitude $< 5(P(j) - \frac{1}{2})$.*

$\Rightarrow$ Gradient-descent is likely to fail on every cantor distribution with strong depth separation, even though the network is expressive enough.

Experiments

- Training depth 5 network on 2D Cantor distribution of depth 5, for different approximation curves:



- Training networks of various depth and width on 2D Cantor distribution with "good" curve vs. training on CIFAR-10: