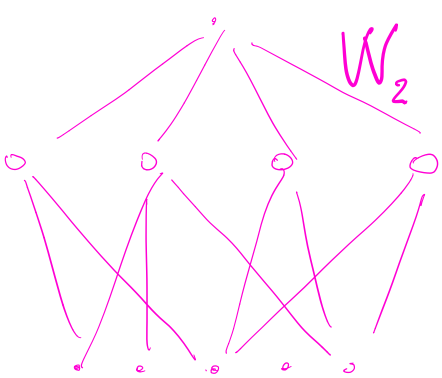
Why deep nets?

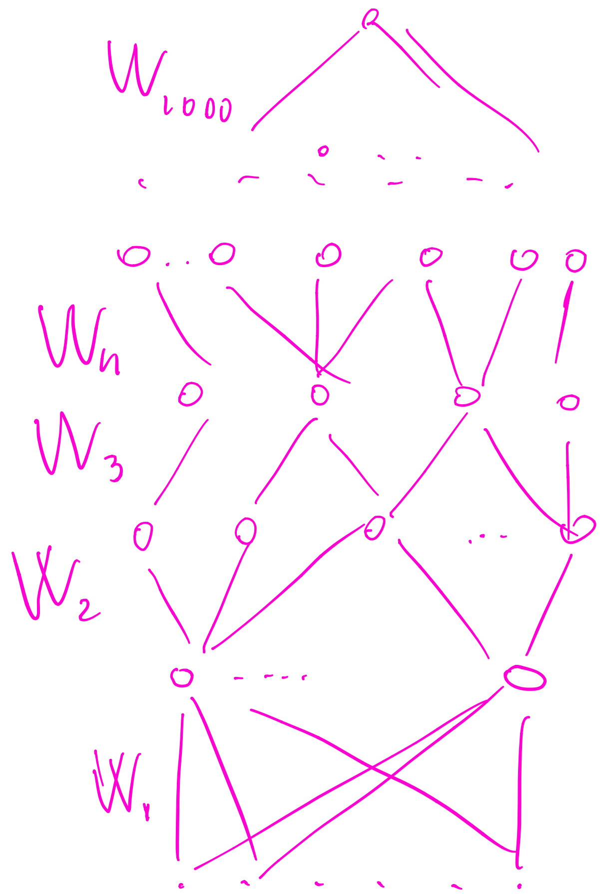
Is deep better than shallow?

When?

Why?



Shallow



Deep

Some history

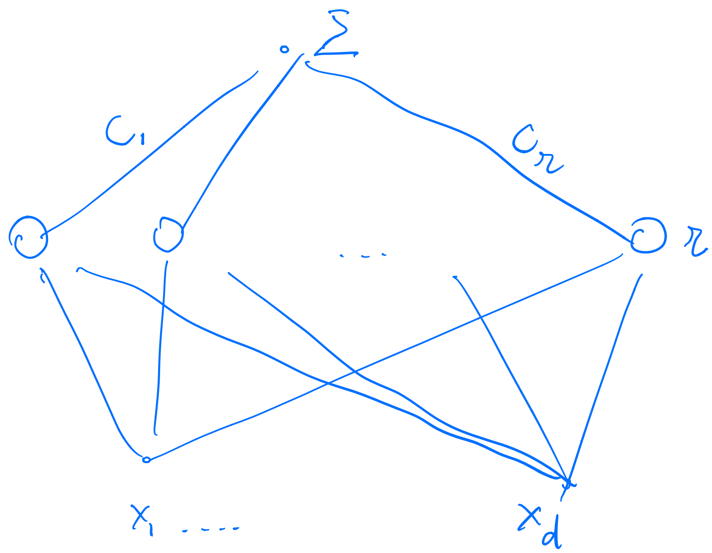
80': S.N.N.

2018:

Approximation Theory

why is depth better?

Shallow networks



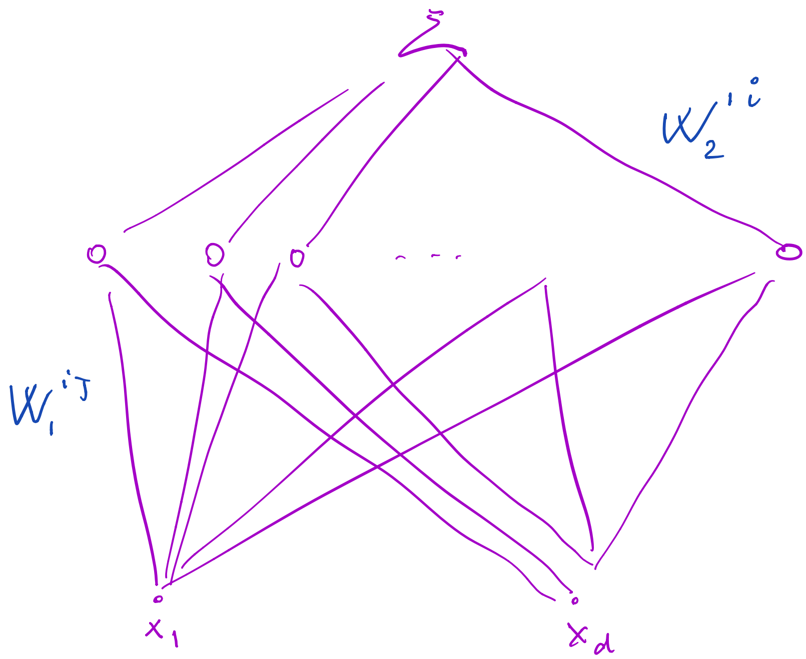
$W_2^{1,5}$

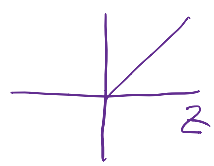
Main example: Kernel machine

$$y = \sum_{i}^{i=r} c_i K(x, x_i)$$

$$\min_{f \in H_n} \frac{1}{\cancel{n}} \sum^{n} \left(f(x_i) - y_i \right)^2 + \lambda \|f\|_H^2$$

Another example

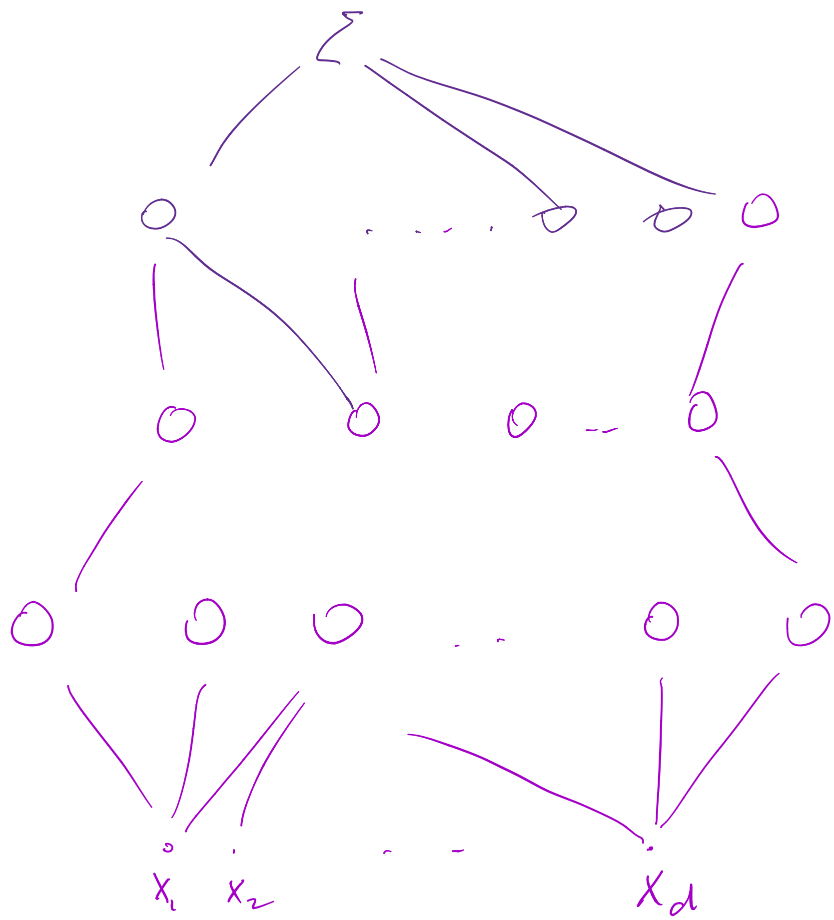


$\sigma(z) = [z]_+ = \max(0, z) =$ 

$$y = \sum_i W_2^{1i} \, \sigma\left(\sum_j W_1^{ij} x_j\right)$$

usually $\sigma(\langle w, x\rangle + b)$ but I eliminate b by assuming one of the components is $x_d = 1$ so $W^{ij} = b_i$

Deep Networks



$$y = W_L^{J} \sigma\left(W_{L-1}^{Jk} \cdots \sigma\left(W^{kp} x_p\right)\right)$$

E. summation convention

Networks to approximate/represent functions

- Are deep nets better than shallow ones?
- The answer in the 80' was: no!

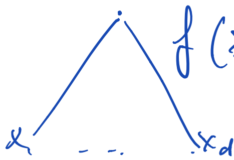
We will see

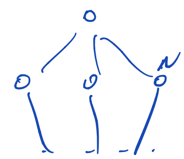
① proof of above

② a new answer: deep nets can be much better for certain f.

Key ideas in approximation theory

Functions and approximators





$g \in V_N$

Example $\quad \forall \varepsilon\ \exists \delta$ s.t. $|f(x+\delta) - f(x)| < \varepsilon$

$$f \in C(\mathbb{R}^d)$$

$$g \in V_n \longrightarrow \operatorname*{span}_N \sigma(W^{ij} x^T)$$

Density

$\forall f \in C(\mathbb{R}^d)$ $\forall$ compact K, $K \subset \mathbb{R}^d$ $\forall \varepsilon > 0$

$\exists g \in V$ s.t. $\sup_{x \in K} |f(x) - g(x)| < \varepsilon$

set of networks

networks

Degree of approximation

$\forall f \in C(\mathbb{R}^d)$

$\inf_{g \in V_n} \| f - g_\alpha \| < \varepsilon \implies$

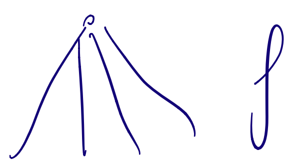
how large is α ?

Shallow nets:

density + degree approximation

Consider target functions

$$f \in W_m^d \qquad \|f\| = \sum_{i \le |k| \le m} \|D^k f\| \le 1$$



no structure assumed

$$g(x) = \sum^N c_i \langle \underline{w}^i, \underline{x} \rangle$$

Theorem

$\forall f \in W_m^d$, $\exists\, g \in V_N$ s.t.

$|g(x) - f(x)| < \varepsilon$ with

$$N = O(\varepsilon^{-d/m})$$

<u>Curse of dimensionality</u>

Bellman's term:

$\Longrightarrow$

- ☐ optimization
  cannot be done by
  RS
- ☐ function approximation
  requires $\left(\frac{1}{\varepsilon}\right)^d$ evaluations
  for f Lipschitz order ε
- ☐ integration ...

Blessings of

1) Smoothness

Barron's Gorosi

2) compositionality

Examples:

- P_k^d has $\binom{d+k}{k} = \dfrac{d+k\,!}{d!\,k!} =$

- $\approx k^d$ monomials

- A function of 10 variables corresponds to a 10 D table. If each dimension is discretized in just 10 partitions I have table with 10^{10} entries. If $d = 100$ pixels then 10^{100} entries

- If $f \in W_m^d \longrightarrow N = O\left(\varepsilon^{-d/m}\right)$

  For $\varepsilon = 10^{-1}$ $d = 100$ $N = O\left(10^{100}\right)$

Summary proof

① $\sum_i^K c_i \sigma(a_i x + b_i) \sim p(x) \in \Pi_{K-1}$

② $\sum_i^r P_i^K(\langle W, x \rangle) \in P_K$ (pol up degree K in d variables)

$r \sim K^d \Rightarrow K \sim r^{1/d}$

③ Sobolev $E(W_d^m, R_r, L_p) \leq C K^{-m}$

space of ridge function with r elements

$R_r = \left\{ \sum_i^r g_i(a^i \cdot x) : a^i \in \mathbb{R}^n,\ g_i \in C(\mathbb{R}) \right\}_{i=1 \ldots r}$

② + ③ $\Longrightarrow E \leq C r^{-m/d}$

Logic of ① + ② + ③

- Networks approximate univariate poly
- Univariate pol in $\langle w, x \rangle$ represent multivariate pol
- Multivariate pol approximate Sobolev functions

Thus theorem.

① Any univariate $p(x)$ can be represented as linear combination of smooth ReLU

<u>proof</u>

$$\lim_{h \to 0} \frac{\sigma((a+h)x + b) - \sigma(ax+b)}{h} =$$

$$= \frac{d}{da} \sigma(ax+b)\Big|_{a=0} = x\,\sigma'(b)$$

<u>Theorem</u>

If σ is not a polynomial, $\sigma \in C^\infty$, the closure of N_r (span of r σ) contains the linear space of Π^{r-1}

Second derivative which needs 3 terms gives x^2

...

Thus N_2 is dense in $C(K)$ because of Weierstrass theorem.

From 1D to dD

$$P(x) = \sum_i^r p_i(\langle \underline{w}^i, \underline{x} \rangle) \quad \text{// } p_i \text{ univariate} \quad \in \mathbb{R}^d$$

$\in H_k^d$ (homogeneous pol in d variables of degree k)

$$r = \dim H_k = \binom{d+k-1}{k} = \frac{(d+k-1)!}{(d-1)!\,k!} = \sim k^d$$

- thus H_k^d pol. can be represented by a network with $r \sim k^d$ units

We want to show that

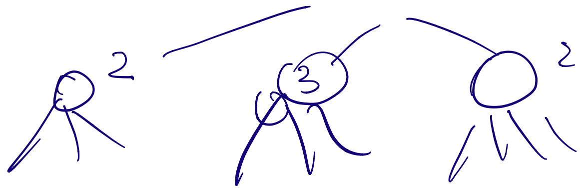
if $P(x)$ pol on $\mathbb{R}^d$ <u>then</u>

$$P(x) = \sum_i^r p_i(\langle w_i, \underline{x} \rangle)$$

for some choice of r w_i and p_i

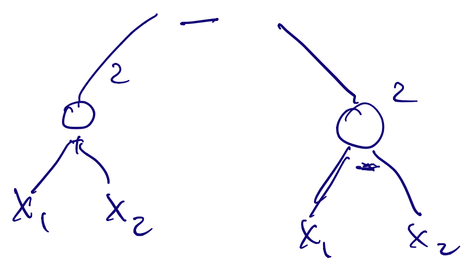
No general proof but consider following:

assume network with units s.t.



Can I synthesize $\forall$ $P(x)$ of

[illegible]
d variables? I can get x_1^2, x_2^2 . . .
how do I get $x_1 x_2$? Well



$= 4 x_1 x_2$

Remark

H_k^d hom pol degree k in $\mathbb{R}^d$

P_k^d " " " " "

$$\dim H_k^d = \binom{n-1+k}{k} = R$$

$$\dim P_k = \binom{n+k}{k}$$

③ Define $E(B, X, L_p) =$

$$= \inf_{p \in X} |p - f|_{L_p} \quad \forall f \in B$$

Theorem

$$E(W_d^m, N_r, L_p) \leq C r^{-m/d-1}$$

$$N_r = \sum_{i=1}^{r} \sigma(\langle \underline{w}^i, \underline{x} \rangle)$$

proof

Classically $E(W^m, P_k, L_p) \leq C k^{-m}$

Since $r \sim k^{d-1} \Rightarrow k \sim r^{1/d-1} \Rightarrow$

$E(W^m, N_r, L_p) \sim E(W^m, P_k, L_p) \sim$

$\sim C r^{-\frac{m}{d-1}}$

Remarks

1) Even without (B) a shallow net can represent arb. well polynomials in P_k^d with r units $r \sim k^d$

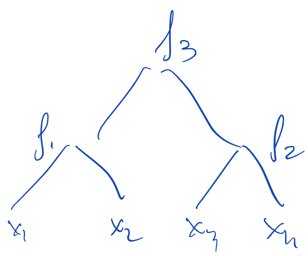
Depth and curse

For general functions
shallow and deep nets
suffer curse of dimensionality

But... for Local (Hierarchical)
Compositional functions
deep nets - unlike shallow
ones - do not have curse

LHC functions

Simplest example



$$f(x_1, x_2, x_3, x_4) =$$

$$= f_3\left(f_1(x_1, x_2), f_2(x_3, x_4)\right)$$

$f \in W_m^{d,2}$ with $f_i \in W_m^2$

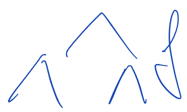
Another example

$$f(x_1, x_2) = \left(A x_1 x_2 + B x_1 + C x_2\right)^{2^{10}}$$

shallow net require $\sim 2^{11}$ units

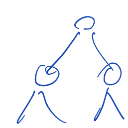
a deep net $\sim 3 + 3 \cdot 10$

Intuition



shallow net $\left(\frac{1}{\varepsilon}\right)^4$ units

deep net $\left(\frac{1}{\varepsilon}\right)^2$ for each node



for total $\sim 3\left(\frac{1}{\varepsilon}\right)^2$ units

Another example

$$y = \sqrt{\sin(x_1 + x_2) \cdot (x_3 \cdot x_4)^2} =$$

$$= h_6\Big(h_5\big(h_2(x_1, x_2), h_3(x_3, x_4)\big)\Big)$$

Theorem

Deep nets with same graph approximate functions in d variables in $W_m^{d,2}$ with # units per node $\sim O\left(\frac{1}{\varepsilon}\right)^{2/m}$ for total units $O\left((d-1)\left(\frac{1}{\varepsilon}\right)^{2/m}\right)$

Proof

Each h can be approximated with $O\left(\varepsilon^{-2/m}\right)$ units. We assume each h is Lipschitz continuous

that is $|h(x) - h(x+\varepsilon)| \leq L\varepsilon$

By hypothesis $\|h - P\| < \varepsilon$ $|h_1 - p_1| < \varepsilon$

$|h_2 - p_2| < \varepsilon$. Then

$|h(h_1, h_2) - P(p_1, p_2)| =$

$|h(h_1, h_2) - h(p_1, p_2) + h(p_1, p_2) - P(p_1, p_2)|$

$\leq |h - h| + |h - P| \leq \varepsilon + \varepsilon \sim O(\varepsilon)$

Minkowski

Lipschitz hypothesis

$|f(x_1) - f(x_2)| \leq K |x_1 - x_2|$

- More general theorems for DAGs

This theorem may explain why deep nets are successful and why all the really good ones are CNNs

Locality is key, not weight sharing! weight sharing helps but not exponentially