[DLT 1-2]
Approximation power of deep networks

Matus Telgarsky <mjt@illinois.edu>

(with help from many friends!)

Deep learning *theory*?

- Deep learning research is fueled by practical success.

Deep learning *theory*?

- Deep learning research is fueled by practical success.
- Deep learning *theory* tries to be practically relevant.

Deep learning *theory*?

- Deep learning research is fueled by practical success.
- Deep learning *theory* tries to be practically relevant.
- Many core ML questions revisited, e.g.:
  - ReLU VC bounds are "tight";
    but this is not satisfactory.
  - "Generalization" should include covariate shift,
    adversarial examples, . . . ?

Why deep learning theory?

- (One) Eventual goal: contribute to algorithms.
  (Hopeful in difficult domains like RL?)

Why deep learning theory?

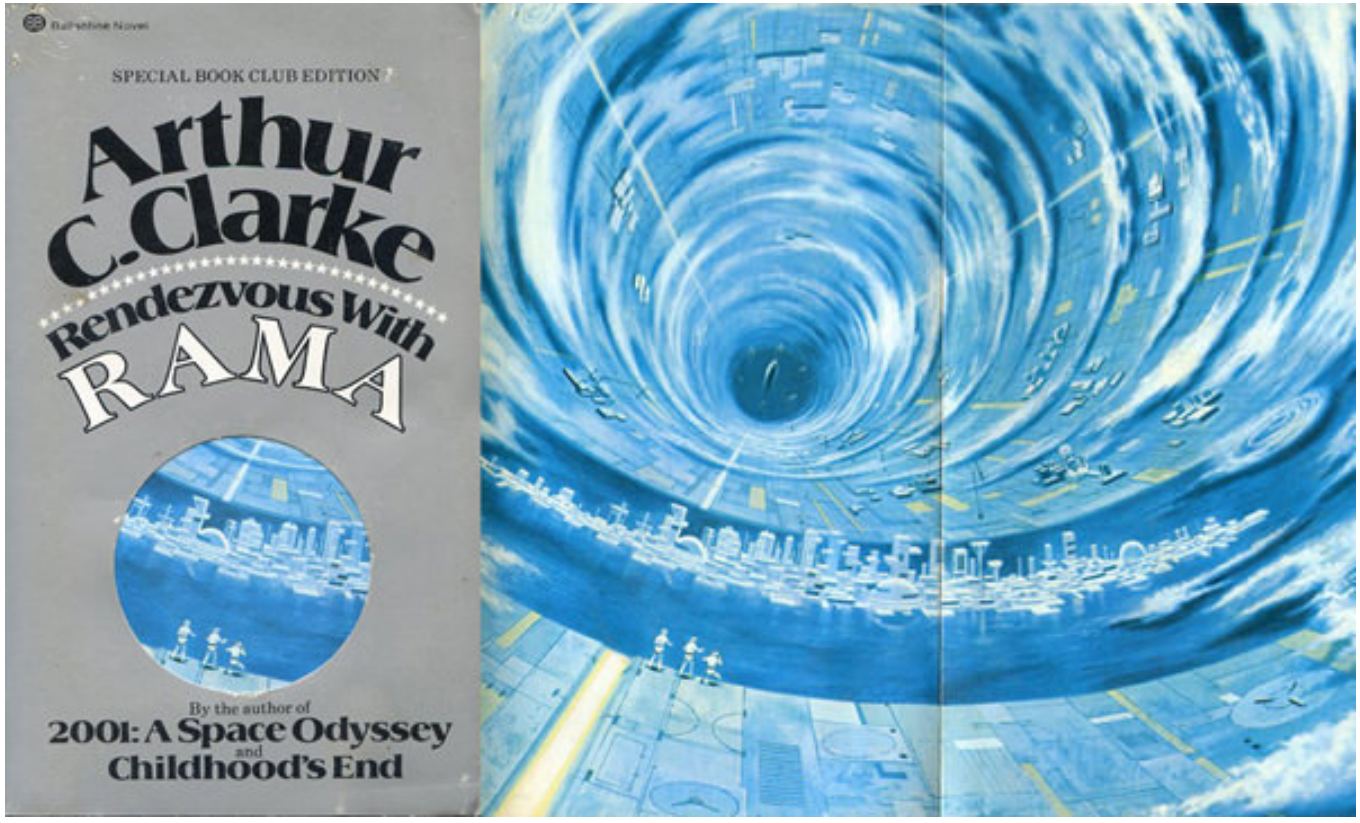
- (One) Eventual goal: contribute to algorithms.
  (Hopeful in difficult domains like RL?)
- Primary goal for now: analyze existing methods.

Why deep learning theory?

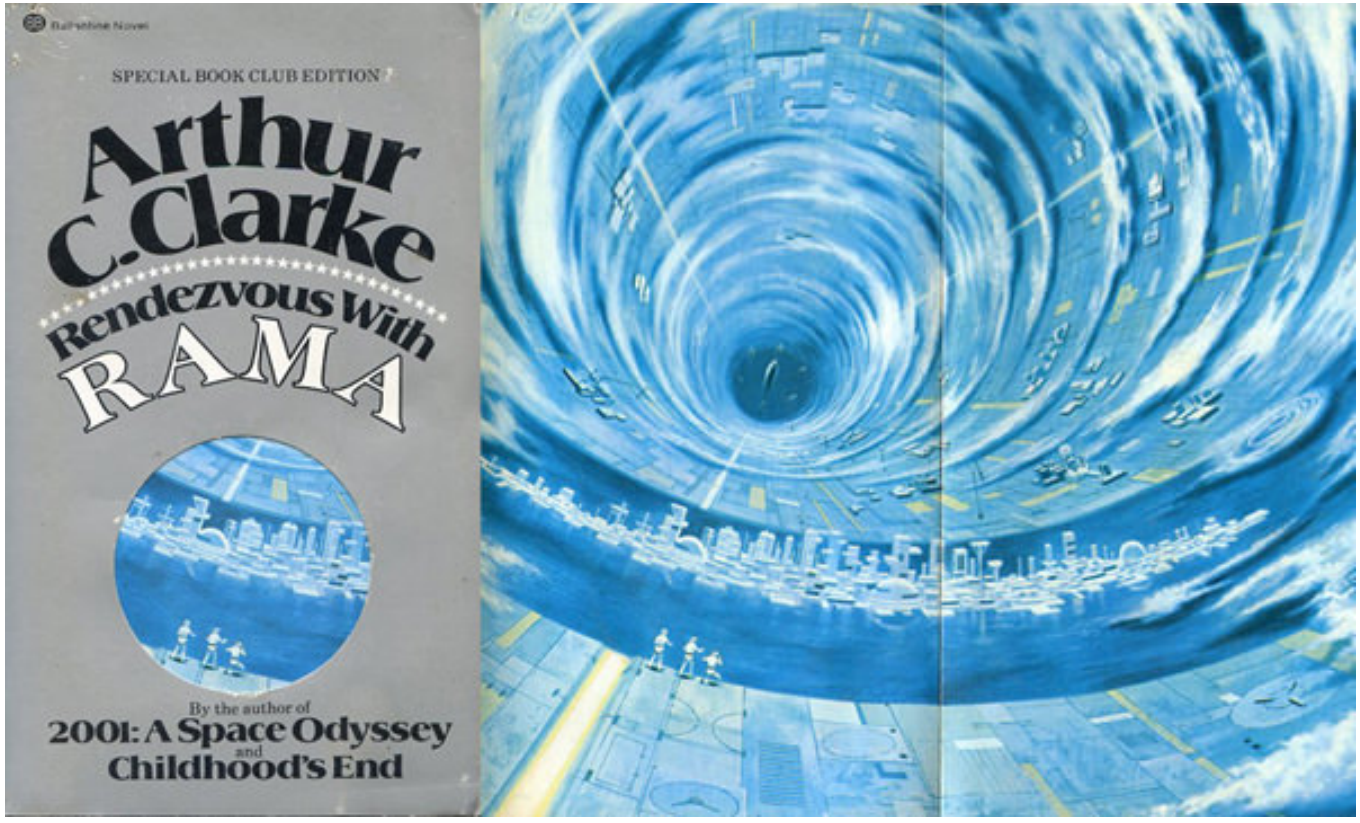
- (One) Eventual goal: contribute to algorithms.
  (Hopeful in difficult domains like RL?)
- Primary goal for now: analyze existing methods.
- Some people say DL theory is useless. . .

. . . one perspective on DL theory. . .

. . . one perspective on DL theory. . .



. . . one perspective on DL theory. . .



Probe an alien technology with mathematical tools!
Which tools are best?

A few DLT questions:

- What natural phenomena can deep networks approximate?
- Why does gradient descent work?
- Why do deep networks generalize?
- How does GAN training work? How can we defend against adversarial examples? How can we interpret deep network predictions? ...

A few DLT questions:

- What natural phenomena can deep networks approximate?
- Why does gradient descent work?
- Why do deep networks generalize?
- How does GAN training work? How can we defend against adversarial examples? How can we interpret deep network predictions? ...

Scope of these 4 lectures:

- 1-2: approximation theory.
- 3-4: margin perspective on optimization and generalization. *This part is highly biased!*

Goal (in DLT 1-2): in some prediction problem,

replace $f : \mathbb{R}^d \to \mathbb{R}$ with neural network $g : \mathbb{R}^d \to \mathbb{R}$.

Goal (in DLT 1-2): in some prediction problem,

replace $f : \mathbb{R}^d \to \mathbb{R}$ with neural network $g : \mathbb{R}^d \to \mathbb{R}$.

Primary setting: statistical learning theory, thus

$$\int \ell(f(x), y) \, \mathrm{d}P(x, y) \qquad \text{vs.} \qquad \int \ell(g(x), y) \, \mathrm{d}P(x, y).$$

Goal (in DLT 1-2): in some prediction problem,

replace $f : \mathbb{R}^d \to \mathbb{R}$ with neural network $g : \mathbb{R}^d \to \mathbb{R}$.

Primary setting: statistical learning theory, thus

$$\int \ell(f(x), y)\, \mathrm{d}P(x, y) \qquad \text{vs.} \qquad \int \ell(g(x), y)\, \mathrm{d}P(x, y).$$

- **Upper bounds:** If $\ell(\cdot, y)$ is 1-Lipschitz,

$$\int \left[\ell(g(x), y) - \ell(f(x), y)\right] \mathrm{d}P(x, y) \leq \left|g(x) - f(x)\right| \mathrm{d}P(x, y);$$

we will aim to make this small everywhere
(universal/uniform/$L_\infty(P)$ apx), or in $L_1(P)$.

Goal (in DLT 1-2): in some prediction problem,

replace $f : \mathbb{R}^d \to \mathbb{R}$ with neural network $g : \mathbb{R}^d \to \mathbb{R}$.

Primary setting: statistical learning theory, thus

$$\int \ell(f(x), y)\, \mathrm{d}P(x, y) \qquad \text{vs.} \qquad \int \ell(g(x), y)\, \mathrm{d}P(x, y).$$

- **Upper bounds:** If $\ell(\cdot, y)$ is 1-Lipschitz,

  $$\int \big[\ell(g(x), y) - \ell(f(x), y)\big]\, \mathrm{d}P(x, y) \le \big|g(x) - f(x)\big|\, \mathrm{d}P(x, y);$$

  we will aim to make this small everywhere (**universal/uniform/$L_\infty(P)$ apx**), or in $L_1(P)$.
- **Lower bounds:** we want large error on a large set; as a surrogate, $|g - f|$ large in $L_1(P)$ or $L_1(\mathrm{Unif})$.

By **deep networks** we mostly mean

$$x \mapsto A_L \sigma_{L-1} \left(\cdots \sigma_1 (A_1 x + b_1) \cdots \right) + b_L,$$

where nonlinearity/activation/transfer σ_i
is applied coordinate-wise.

By **deep networks** we mostly mean

$$x \mapsto A_L \sigma_{L-1} \left(\cdots \sigma_1 (A_1 x + b_1) \cdots \right) + b_L,$$

where nonlinearity/activation/transfer σ_i
is applied coordinate-wise.

There are many conventions;
we will briefly discuss others.

By **deep networks** we mostly mean

$$x \mapsto A_L \sigma_{L-1} \left(\cdots \sigma_1 (A_1 x + b_1) \cdots \right) + b_L,$$

where nonlinearity/activation/transfer σ_i
is applied coordinate-wise.

There are many conventions;
we will briefly discuss others.

We'll mostly stick to the ReLU $z \mapsto \max\{0, z\}$ (Fukushima '80);
it's easy to convert.

By **deep networks** we mostly mean

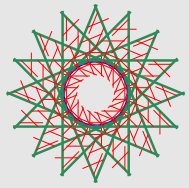
$$x \mapsto A_L \sigma_{L-1} \left(\cdots \sigma_1 (A_1 x + b_1) \cdots \right) + b_L,$$

where nonlinearity/activation/transfer σ_i
is applied coordinate-wise.

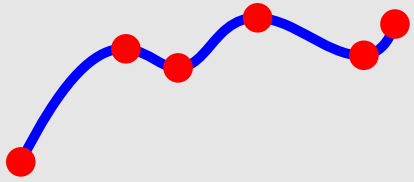
There are many conventions;
we will briefly discuss others.

We'll mostly stick to the ReLU $z \mapsto \max\{0, z\}$ (Fukushima '80);
it's easy to convert.

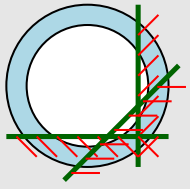
Many constructions today do not control Lipschitz constants
and seem impractical;
see (BJTX '19) for more discussion.



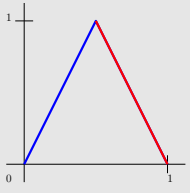
Elementary universal approximation.



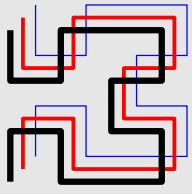
Classical universal approximation.



Benefits of depth.

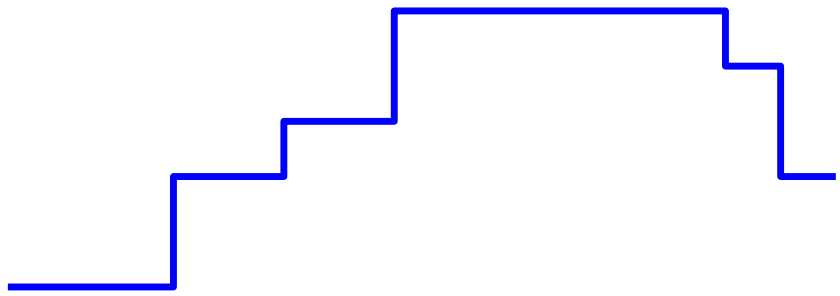


Sobolev spaces.

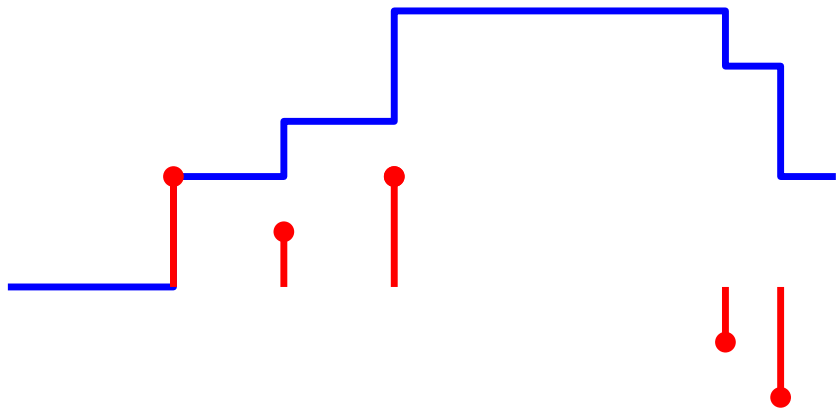


Odds & ends.

Univariate functions via step activations

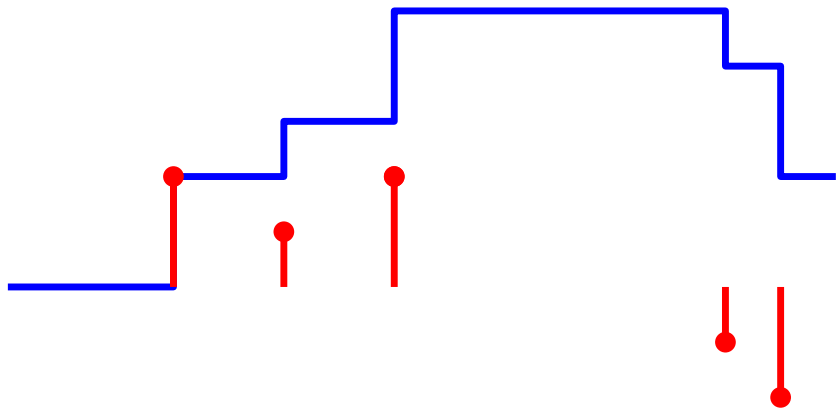


Univariate functions via step activations



$$x \mapsto 2 \cdot \mathbb{1}[x - 3 \geq 0] + \mathbb{1}[x - 5 \geq 0] + 2 \cdot \mathbb{1}[x - 7 \geq 0] - \mathbb{1}[x - 13 \geq 0] \cdots$$

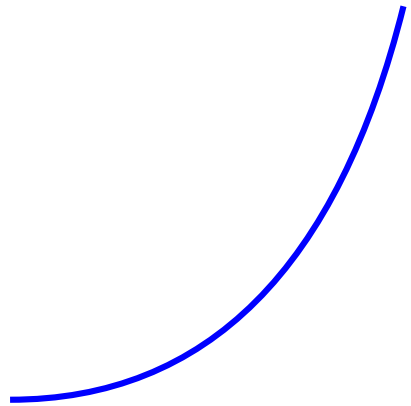
Univariate functions via step activations



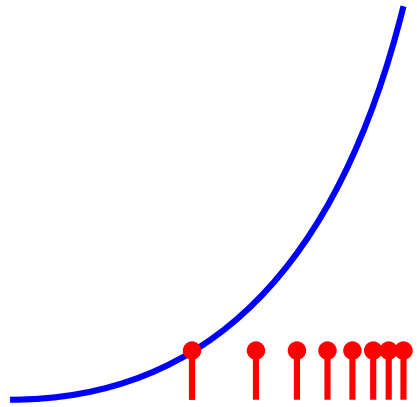
$$x \mapsto 2 \cdot \mathbb{1}[x-3 \geq 0] + \mathbb{1}[x-5 \geq 0] + 2 \cdot \mathbb{1}[x-7 \geq 0] - \mathbb{1}[x-13 \geq 0] \cdots$$

Remark. By contrast, polynomials struggle with flat regions.

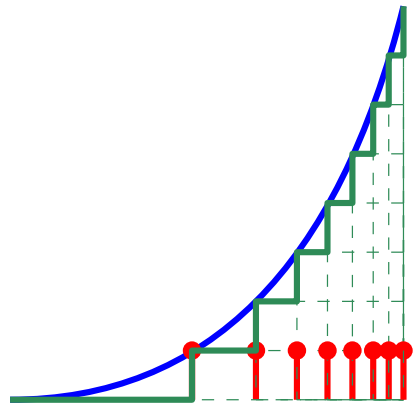
Smooth univariate functions via step activations



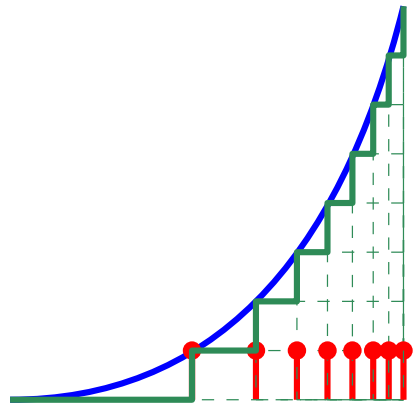
Smooth univariate functions via step activations



Smooth univariate functions via step activations

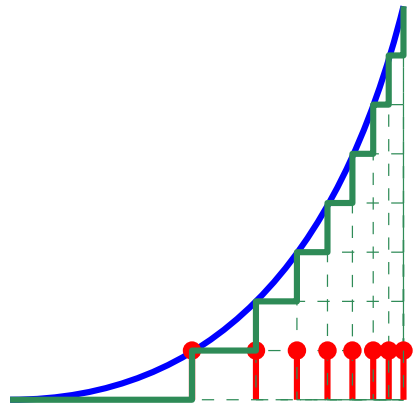


Smooth univariate functions via step activations



Approach #1: subdivide range, $^{\text{Lip}}/_{\epsilon}$ steps.

Smooth univariate functions via step activations

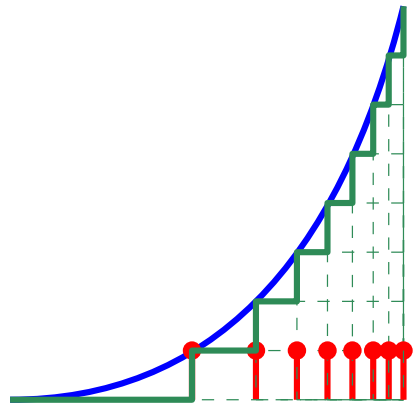


Approach #1: subdivide range, Lip/ϵ steps.
Approach #2: by FTC, for $x \geq 0$,

$$f(x) = f(0) + \int_0^x f'(b)\,\mathrm{d}b = f(0) + \int_0^\infty \mathbb{1}[x - b \geq 0] f'(b)\,\mathrm{d}b.$$

This is a density over infinitely many steps/nodes!
Sample $\int |f'|/\epsilon^2$ steps.

Smooth univariate functions via step activations



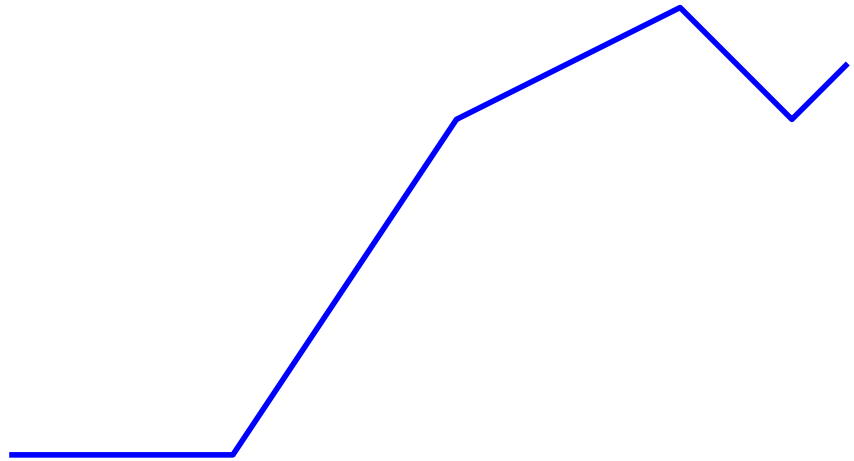
Approach #1: subdivide range, Lip/ϵ steps.
Approach #2: by FTC, for $x \geq 0$,

$$f(x) = f(0) + \int_0^x f'(b)\,\mathrm{d}b = f(0) + \int_0^\infty \mathbb{1}[x - b \geq 0] f'(b)\,\mathrm{d}b.$$

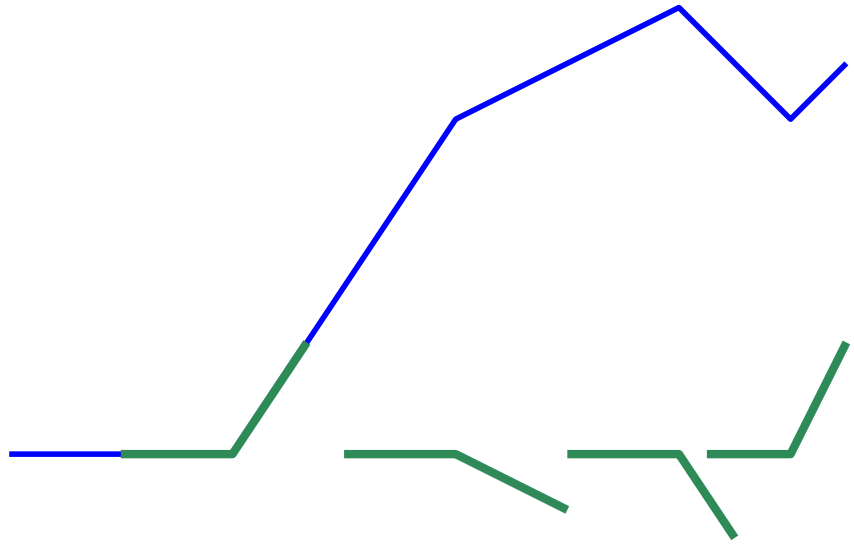
This is a density over infinitely many steps/nodes!
Sample $\int |f'|/\epsilon^2$ steps.
Remarks.

- Infinite width network!
- Refined average-case estimate! (Captures flat regions.)

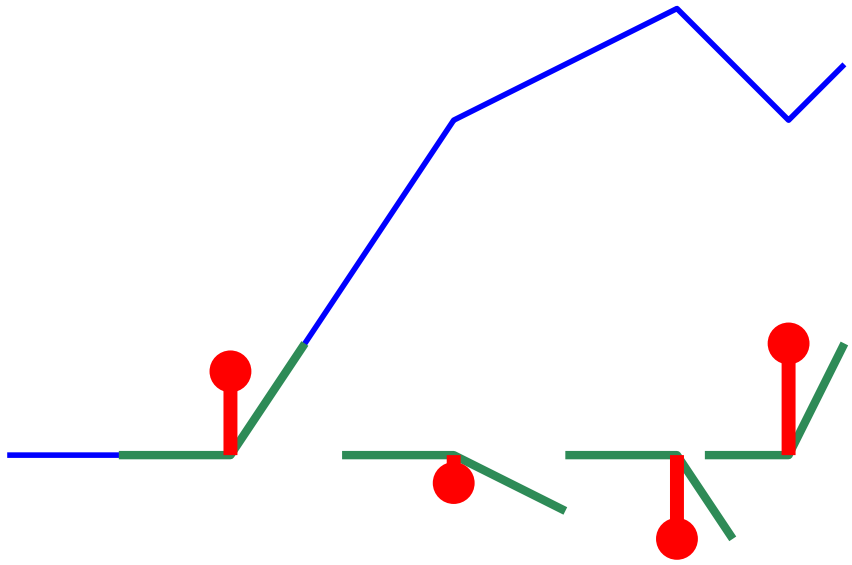
Univariate functions via ReLU activations



Univariate functions via ReLU activations



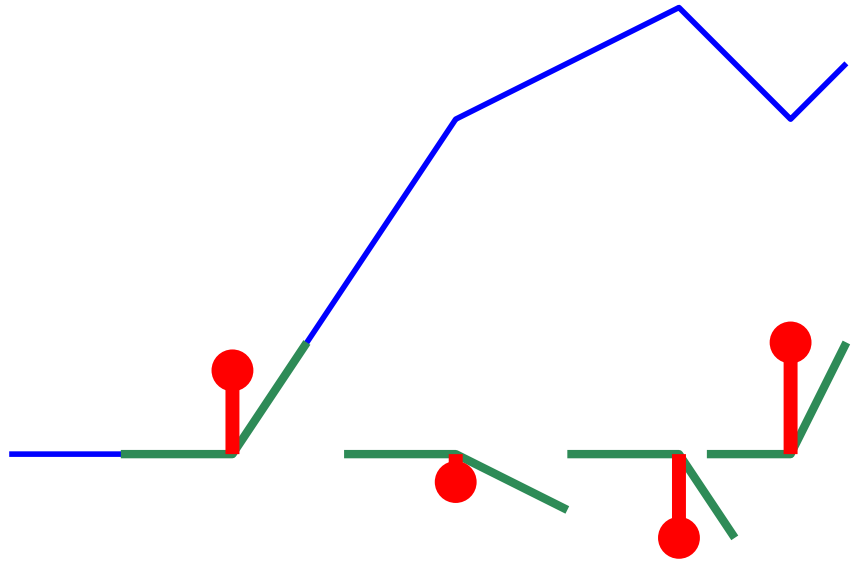
Univariate functions via ReLU activations



Place ReLU $z \mapsto \max\{0, z\}$ on change of slope.

How about smooth functions?

Univariate functions via ReLU activations



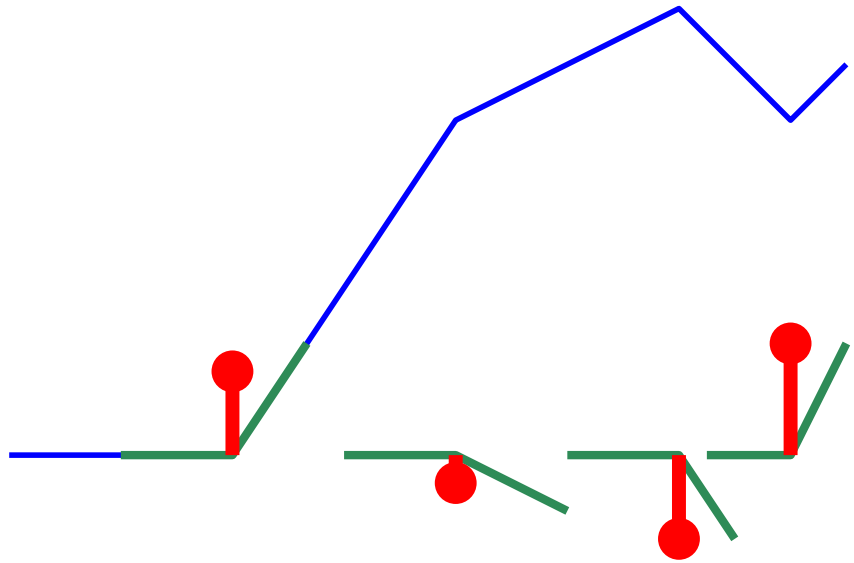
Place ReLU $z \mapsto \max\{0, z\}$ on change of slope.

How about smooth functions? For $x \geq 0$,

$$f(x) = f(0) + \sigma_{\mathrm{r}}(x) f'(0) + \int_0^\infty \sigma_{\mathrm{r}}(x - b) f''(b) \, \mathrm{d}(b).$$

Need to sample $\int |f''|/\epsilon^2$ ReLU!

Univariate functions via ReLU activations



Place ReLU $z \mapsto \max\{0, z\}$ on change of slope.

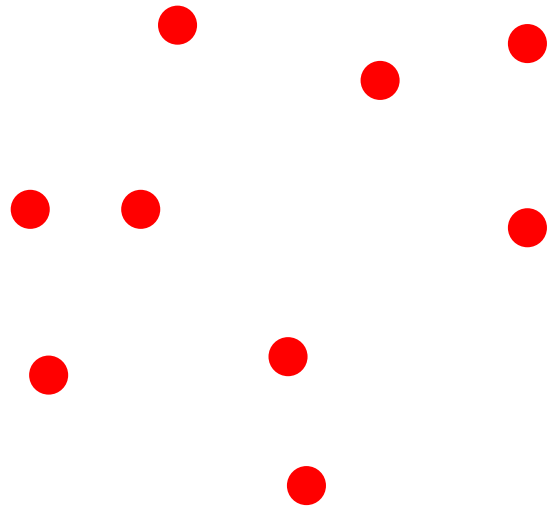
How about smooth functions? For $x \geq 0$,

$$f(x) = f(0) + \sigma_{\mathrm{r}}(x) f'(0) + \int_0^\infty \sigma_{\mathrm{r}}(x - b) f''(b) \, \mathrm{d}(b).$$

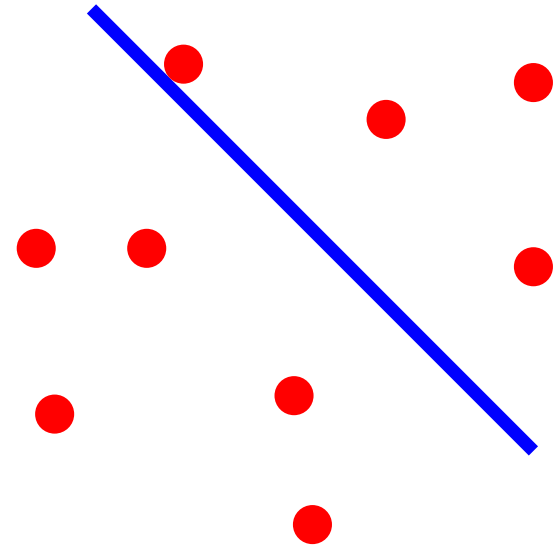
Need to sample $\int |f''|/\epsilon^2$ ReLU!

(In some sense optimal (Savarese-Evron-Soudry-Srebro '19).)

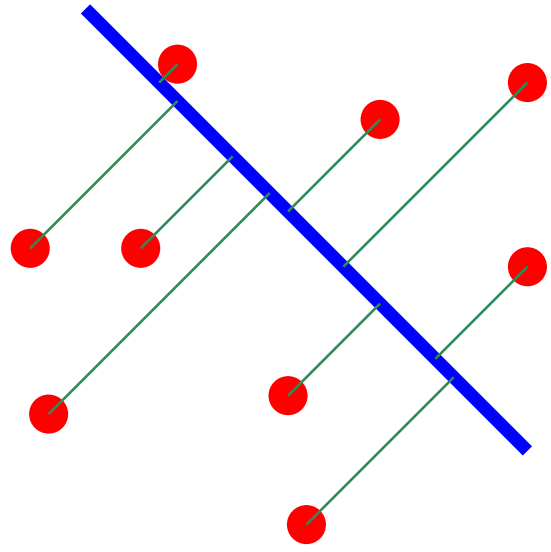
Multivariate, but finitely many points



Multivariate, but finitely many points

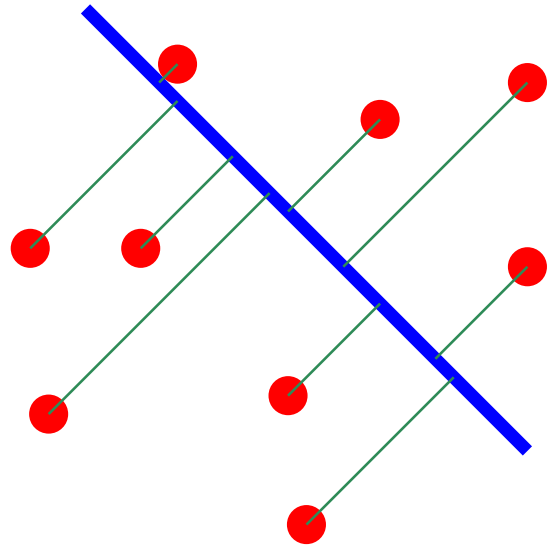


Multivariate, but finitely many points



With probability 1, a random line has unique projections.
We've reduced to the univariate case.

Multivariate, but finitely many points

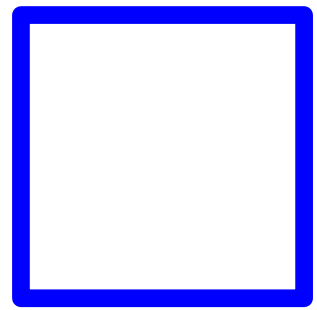


With probability 1, a random line has unique projections.
We've reduced to the univariate case.

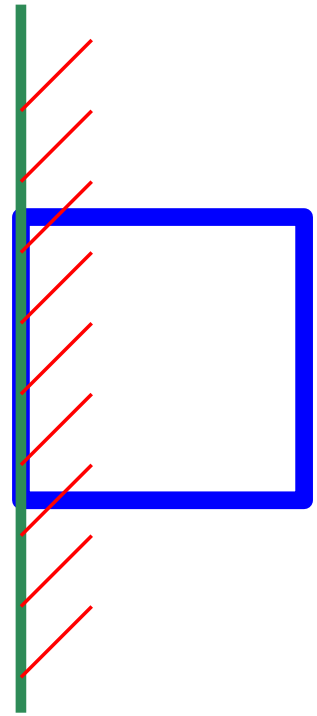
Caveats:

- Representation size may have blown up.
- Finite sets were not our original goal.

Approximate a multivariate box

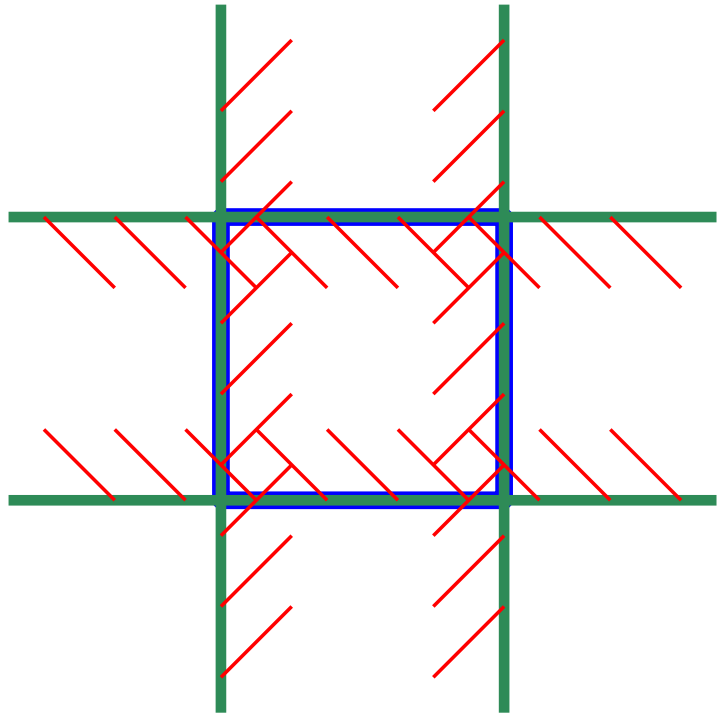


Approximate a multivariate box



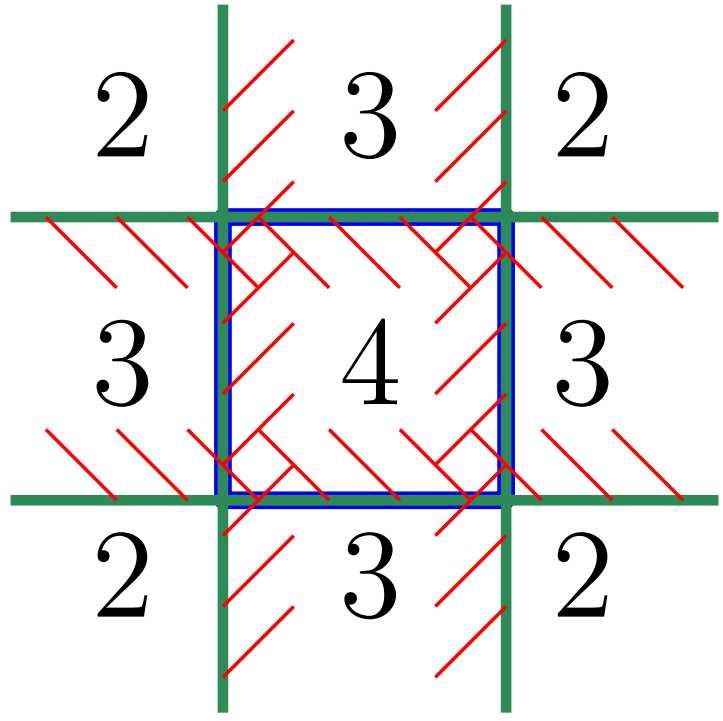
Supporting hyperplanes!

Approximate a multivariate box



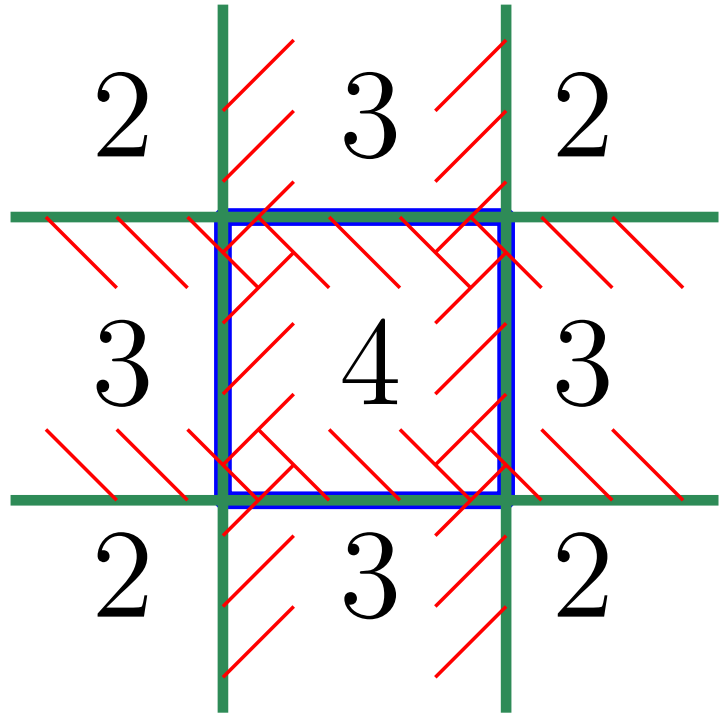
Supporting hyperplanes!

Approximate a multivariate box



Supporting hyperplanes! **. . . oops.**

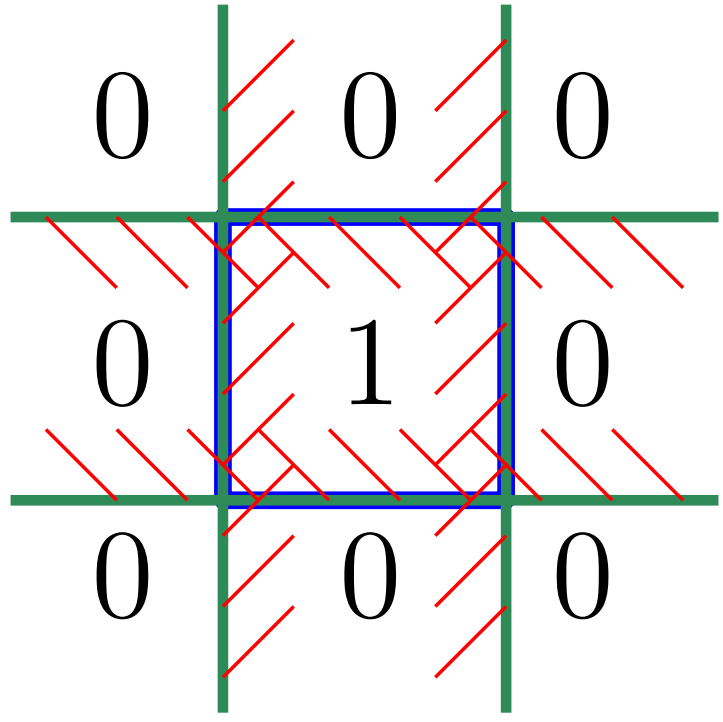
Approximate a multivariate box



Supporting hyperplanes! **. . . oops.**

Fix #1: product halfspaces together! (we'll return to this...)

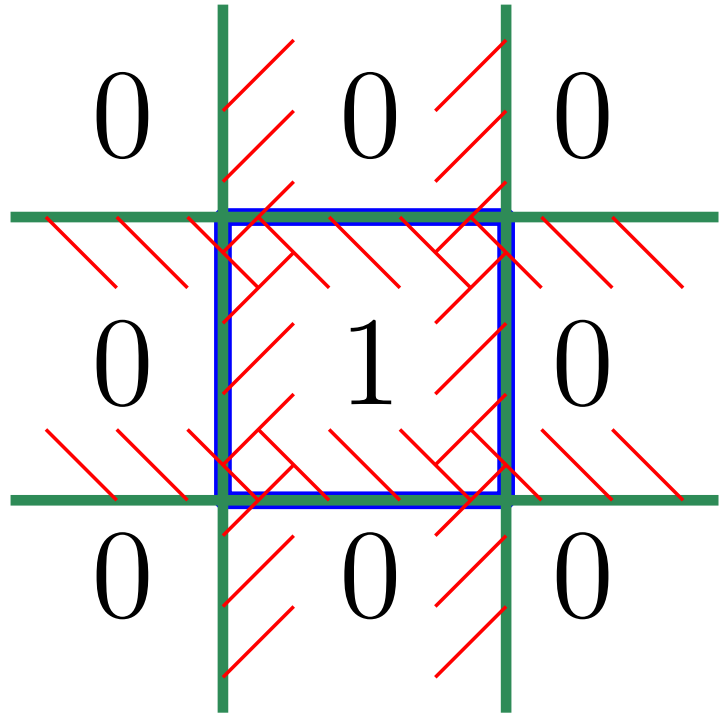
Approximate a multivariate box



Supporting hyperplanes! **. . . oops.**

Fix #1: product halfspaces together! (we'll return to this...)
Fix #2: add a layer, thresholding at 3.5!

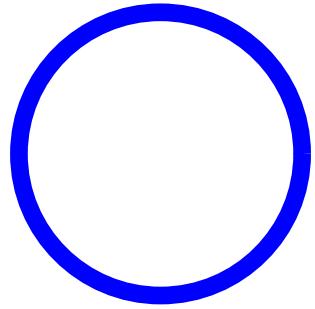
Approximate a multivariate box



Supporting hyperplanes! **. . . oops.**

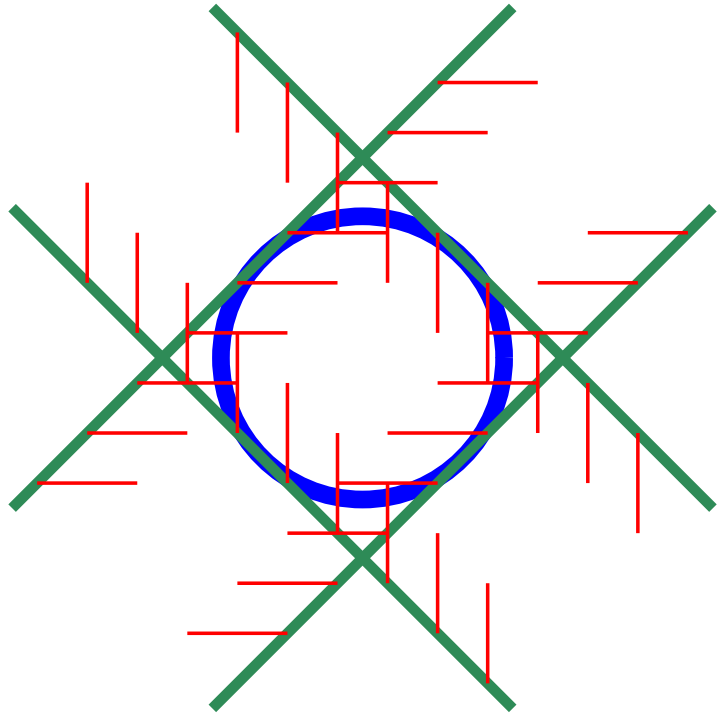
Fix #1: product halfspaces together! (we'll return to this...)
Fix #2: add a layer, thresholding at 3.5!
...how about one ReLU/hidden layer?

Approximate a multivariate ball



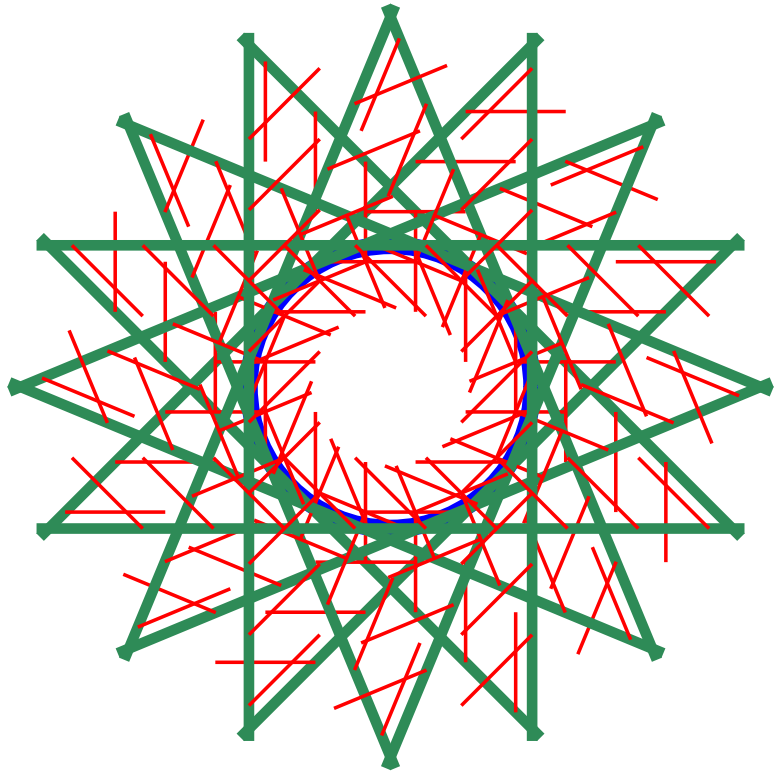
Fix #3: add all the hyperplanes!

Approximate a multivariate ball



Fix #3: add all the hyperplanes!

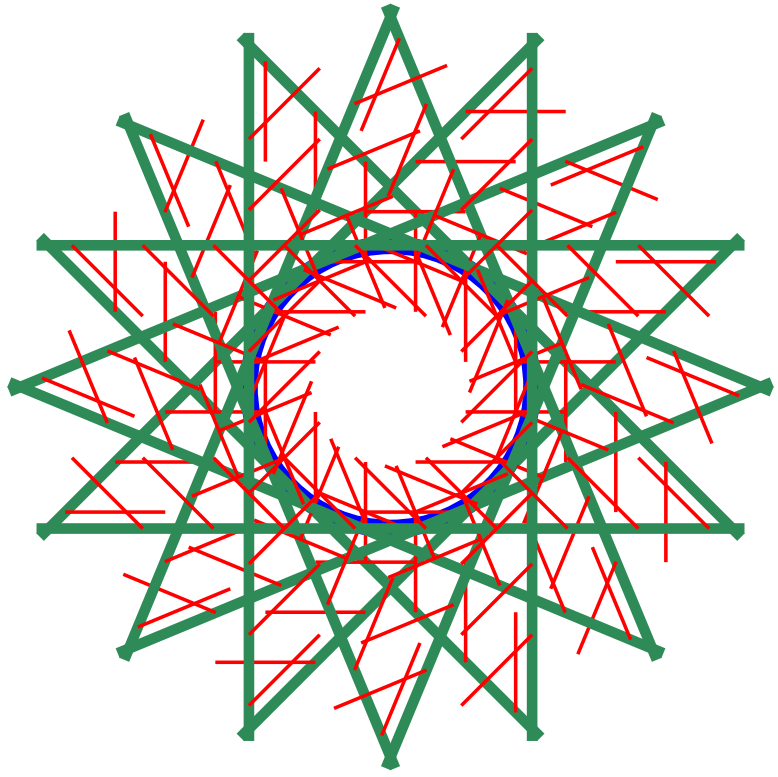
Approximate a multivariate ball



Fix #3: add all the hyperplanes!

Resulting radial function is constant within ball, attenuates away from it.

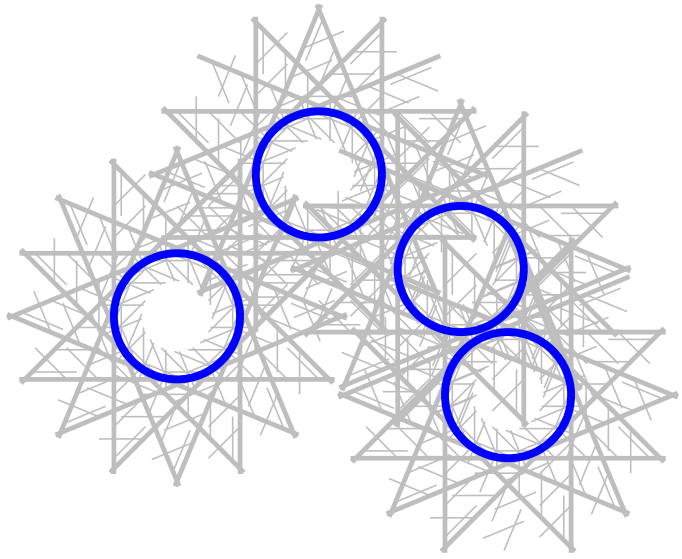
Approximate a multivariate ball



Fix #3: add all the hyperplanes!
Resulting radial function is constant within ball, attenuates away from it.

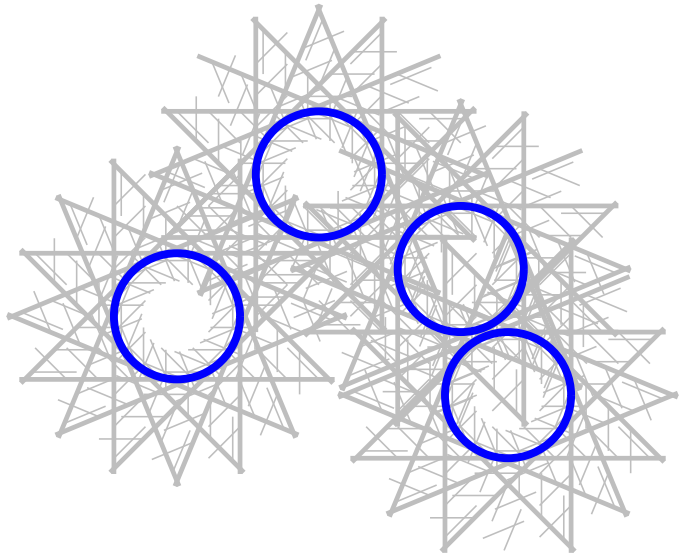
Bad news: good apx seems to require 2^d nodes...
(We'll come back to this.)

Combinations of radial bumps



Normalize bumps/RBFs into density p; convolve with f.

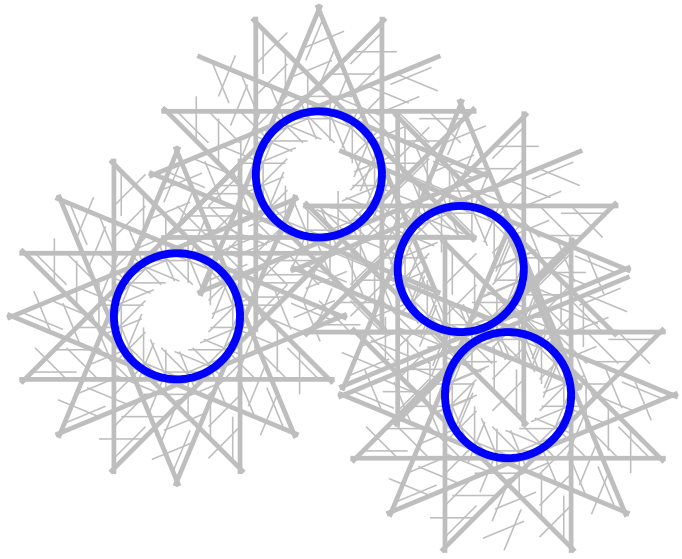
Combinations of radial bumps



Normalize bumps/RBFs into density p; convolve with f.

$$\left| f(x) - \int f(z)p(x-z)\,\mathrm{d}z \right|$$

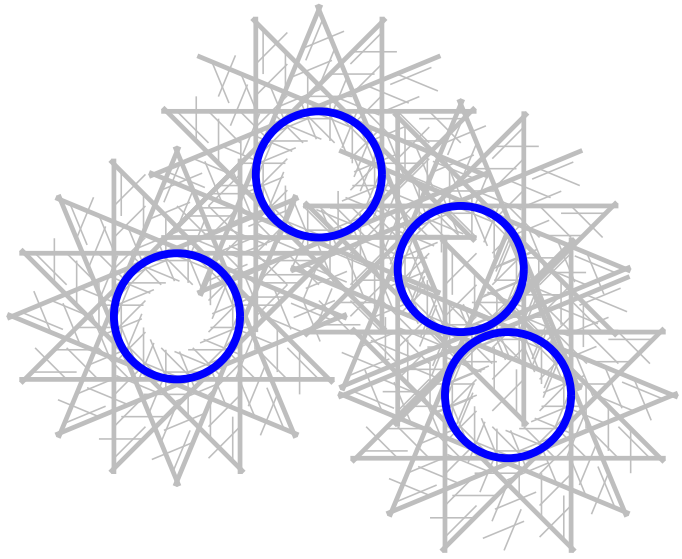
Combinations of radial bumps



Normalize bumps/RBFs into density p; convolve with f.

$$\left| f(x) - \int f(z)p(x-z)\,\mathrm{d}z \right| = \left| f(x) - \int f(x-z)p(z)\,\mathrm{d}z \right|$$

Combinations of radial bumps

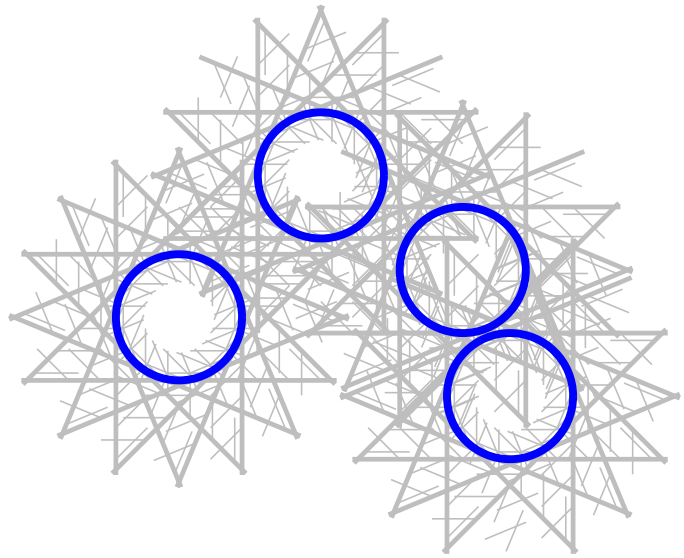


Normalize bumps/RBFs into density p; convolve with f.

$$\left| f(x) - \int f(z)p(x-z)\,\mathrm{d}z \right| = \left| f(x) - \int f(x-z)p(z)\,\mathrm{d}z \right|$$

$$= \left| \int f(x)p(z)\,\mathrm{d}z - \int f(x-z)p(z)\,\mathrm{d}z \right|$$

Combinations of radial bumps

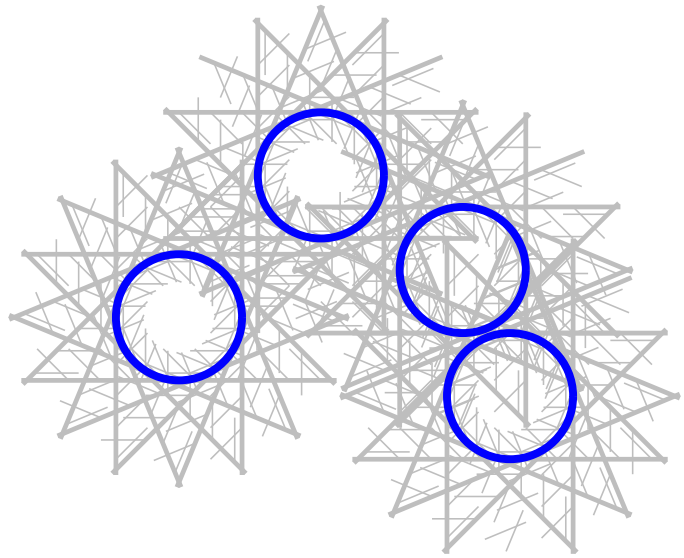


Normalize bumps/RBFs into density p; convolve with f.

$$\left| f(x) - \int f(z)p(x-z)\,\mathrm{d}z \right| = \left| f(x) - \int f(x-z)p(z)\,\mathrm{d}z \right|$$

$$= \left| \int f(x)p(z)\,\mathrm{d}z - \int f(x-z)p(z)\,\mathrm{d}z \right| \leq \int \left| f(x) - f(x-z) \right| p(z)\,\mathrm{d}z,$$

Combinations of radial bumps



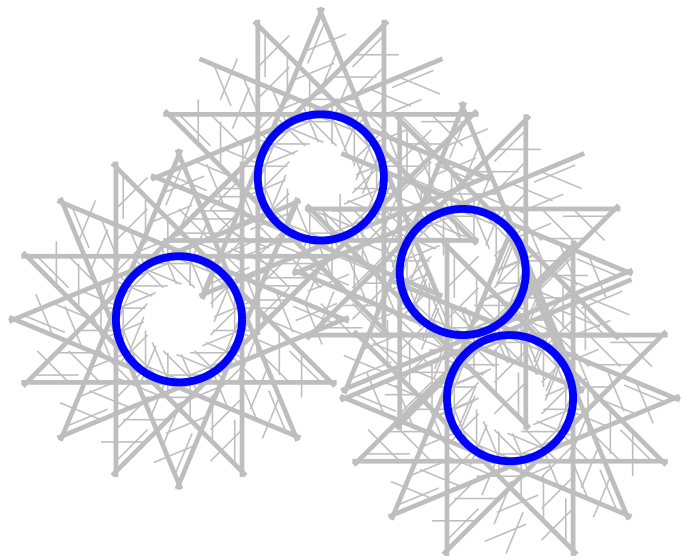
Normalize bumps/RBFs into density p; convolve with f.

$$\left| f(x) - \int f(z)p(x-z)\,\mathrm{d}z \right| = \left| f(x) - \int f(x-z)p(z)\,\mathrm{d}z \right|$$

$$= \left| \int f(x)p(z)\,\mathrm{d}z - \int f(x-z)p(z)\,\mathrm{d}z \right| \leq \int \left| f(x) - f(x-z) \right| p(z)\,\mathrm{d}z,$$

which is small if $p(z) \approx 0$ for large $\|z\|$.

Combinations of radial bumps



Normalize bumps/RBFs into density p; convolve with f.

$$\left|f(x) - \int f(z)p(x-z)\,\mathrm{d}z\right| = \left|f(x) - \int f(x-z)p(z)\,\mathrm{d}z\right|$$

$$= \left|\int f(x)p(z)\,\mathrm{d}z - \int f(x-z)p(z)\,\mathrm{d}z\right| \leq \int \left|f(x) - f(x-z)\right| p(z)\,\mathrm{d}z,$$

which is small if $p(z) \approx 0$ for large $\|z\|$.

Size estimate: $\left(d \cdot \mathrm{Lip}/\epsilon\right)^{\mathcal{O}(d)}$.

(Mhaskar-Michelli '92, BJTX '19.)

So far:

- Easy univariate constructions.
- 3-layer box constructions over $\mathbb{R}^d$: size $\left(\mathrm{Lip}/\epsilon\right)^{\mathcal{O}(d)}$.
- 2-layer RBF convolutions over $\mathbb{R}^d$: size $\left(d\cdot\mathrm{Lip}/\epsilon\right)^{\mathcal{O}(d)}$.

Remarks.

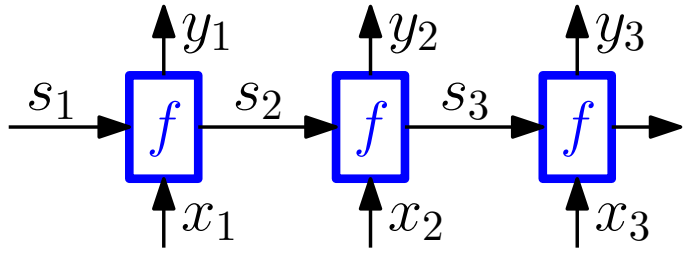
- Impractical constructions! Bad Lipschitz constants.
- Contrast with polynomials: flat pieces.
- Usefuleness of infinite width! Note also:

$$\mathbb{E}\sigma_{\mathrm{r}}(a^{\mathsf{T}}x) = \frac{1}{2}\mathbb{E}|a^{\mathsf{T}}x| = \frac{\|x\|}{\sqrt{2\pi}}.$$

- Poor complexity measures outside univariate!

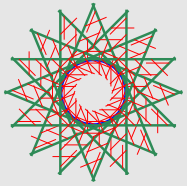
Interlude: three questions

1. Are fixed DN architectures closed under addition?
2. Can RNNs model Turing Machines?

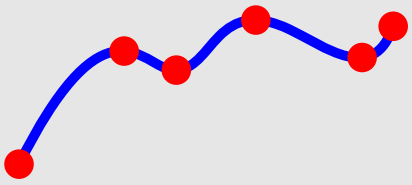


3. Given continuous $g : \mathbb{R}^d \to \mathbb{R}$,
   can we construct custom univariate activations so that

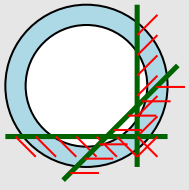
$$g(x) \stackrel{!}{=} \sum_{i=0}^{2d} f_i \left(\sum_{j=1}^{d} h_{i,j}(x_j) \right) ?$$



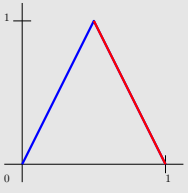
Elementary universal approximation.



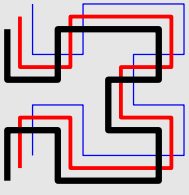
Classical universal approximation.



Benefits of depth.

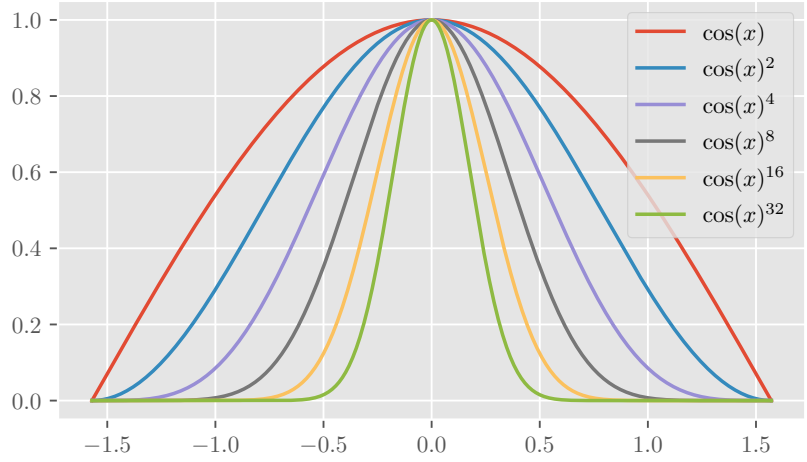


Sobolev spaces.

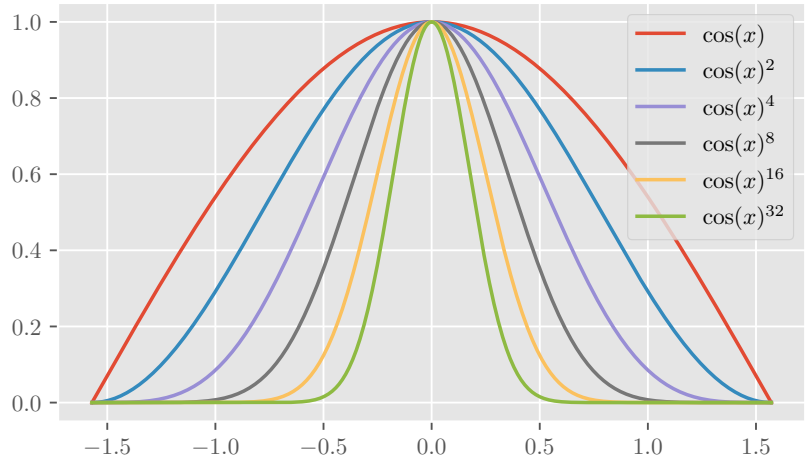


Odds & ends.

Bumps via multiplication

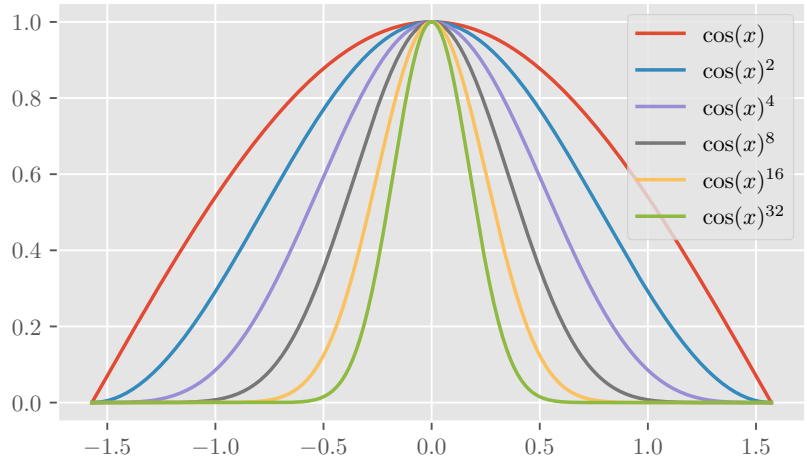


Bumps via multiplication



Univariate bump: $\cos(x)^p$ for large p.

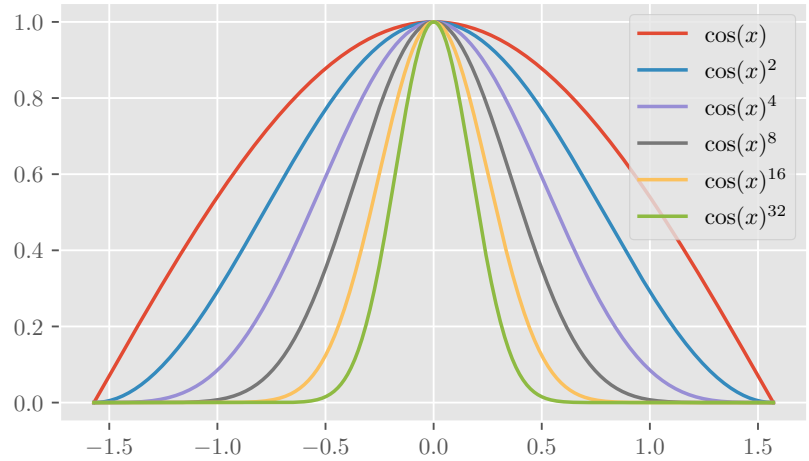
Bumps via multiplication



Univariate bump: $\cos(x)^p$ for large p. Multivariate bump:

$$\mathbb{1}\left[\|x\|_\infty \leq 1\right] = \prod_{i=1}^{d} \mathbb{1}\left[|x_i| \leq 1\right] \qquad \text{and} \qquad \prod_{i=1}^{d} \cos(x_i)^p.$$

Bumps via multiplication



Univariate bump: $\cos(x)^p$ for large p. Multivariate bump:

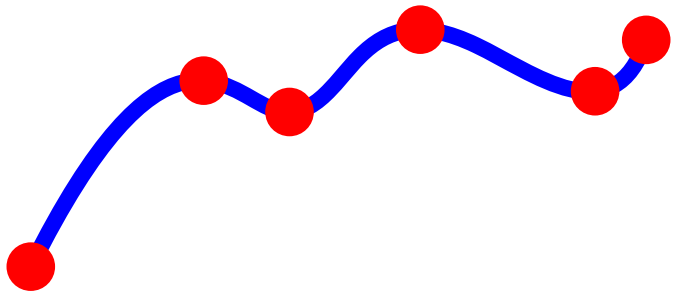
$$\mathbb{1}\left[\|x\|_\infty \le 1\right] = \prod_{i=1}^{d} \mathbb{1}\left[|x_i| \le 1\right] \qquad \text{and} \qquad \prod_{i=1}^{d} \cos(x_i)^p.$$

To remove the product:

$$\cos(x)\cos(x) = \frac{1}{2}\left(\cos(2x) + 1\right),$$

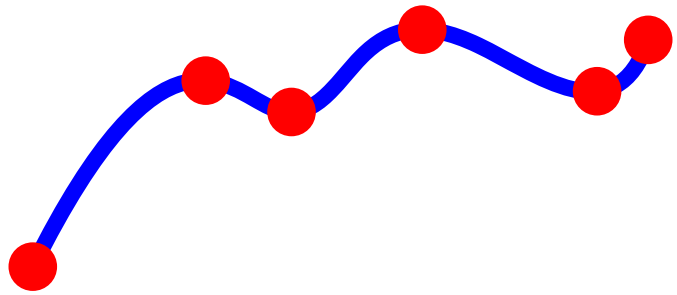
$$2\cos(x_1)\cos(x_2) = \cos(x_1 + x_2) + \cos(x_1 - x_2).$$

Weierstrass approximation theorem



Theorem (Weierstrass, 1885). Polynomials can uniformly approximate continuous functions over compact sets.

Weierstrass approximation theorem



Theorem (Weierstrass, 1885). Polynomials can uniformly approximate continuous functions over compact sets.

Remarks.

- Not a consequence of interpolation:
  must control behavior between interpolants.
- Proofs are interesting; e.g., Bernstein (Bernstein polynomials and tail bounds), Weierstrass (Gaussian smoothing gives analytic functions). ...
- **Stone-Weierstrass theorem:** Polynomial-like function families (e.g., closed under multiplication) also approximate continuous function.

Theorem (Hornik-Stinchcombe-White '89).
Let $\sigma : \mathbb{R} \to \mathbb{R}$ be given with

$$\lim_{z \to -\infty} \sigma(z) = 0, \qquad \lim_{z \to +\infty} \sigma(z) = 1,$$

and define $\mathcal{H}_\sigma := \left\{ x \mapsto \sigma(a^\mathsf{T} x - b) : (a, b) \in \mathbb{R}^{d+1} \right\}$.
Then $\operatorname{span}(\mathcal{H}_\sigma)$ uniformly approximates
continuous functions on compact sets.

Theorem (Hornik-Stinchcombe-White '89).
Let $\sigma : \mathbb{R} \to \mathbb{R}$ be given with

$$\lim_{z \to -\infty} \sigma(z) = 0, \qquad \lim_{z \to +\infty} \sigma(z) = 1,$$

and define $\mathcal{H}_\sigma := \left\{ x \mapsto \sigma(a^\mathsf{T} x - b) : (a, b) \in \mathbb{R}^{d+1} \right\}$.
Then $\operatorname{span}(\mathcal{H}_\sigma)$ uniformly approximates
continuous functions on compact sets.

Proof #1. $\mathcal{H}_{\cos}$ is closed under products since

$$2\cos(a)\cos(b) = \cos(a+b) + \cos(a-b).$$

Now uniformly approximate fixed $\mathcal{H}_{\cos}$ with $\operatorname{span}(\mathcal{H}_\sigma)$.
(Univariate fitting.)

Proof #2. $\mathcal{H}_{\exp}$ is closed under products since $e^a e^b = e^{a+b}$.
Now uniformly approximate fixed $\mathcal{H}_{\exp}$ with $\operatorname{span}(\mathcal{H}_\sigma)$.
(Univariate fitting.)

Theorem (Hornik-Stinchcombe-White '89).
Let $\sigma : \mathbb{R} \to \mathbb{R}$ be given with

$$\lim_{z \to -\infty} \sigma(z) = 0, \qquad \lim_{z \to +\infty} \sigma(z) = 1,$$

and define $\mathcal{H}_\sigma := \left\{ x \mapsto \sigma(a^\mathsf{T} x - b) : (a, b) \in \mathbb{R}^{d+1} \right\}$.
Then $\operatorname{span}(\mathcal{H}_\sigma)$ uniformly approximates
continuous functions on compact sets.

Theorem (Hornik-Stinchcombe-White '89).
Let $\sigma : \mathbb{R} \to \mathbb{R}$ be given with

$$\lim_{z \to -\infty} \sigma(z) = 0, \qquad \lim_{z \to +\infty} \sigma(z) = 1,$$

and define $\mathcal{H}_\sigma := \left\{ x \mapsto \sigma(a^\mathsf{T} x - b) : (a, b) \in \mathbb{R}^{d+1} \right\}$.
Then $\operatorname{span}(\mathcal{H}_\sigma)$ uniformly approximates
continuous functions on compact sets.

Remarks.

- ReLU is fine: use $\sigma(z) := \sigma_\mathrm{r}(z) - \sigma_\mathrm{r}(z-1)$.
- Size estimate: expanding terms, seem to get $\left(\mathrm{Lip}/\epsilon\right)^{\Omega(d)}$.
- Best conditions on σ (Leshno-Lin-Pinkus-Schocken '93): theorem holds **iff** σ not a polynomial.
- Inner hint about DN:
  no need for explicit multiplication?

Other proofs

- (Cybenko '89.)
  Assume contradictorily you miss some functions.
  By duality, $0 = \int \sigma(a^\mathsf{T} x - b)\,\mathrm{d}\mu(x)$
  for some signed measure μ, all (a, b).
  Using Fourier, can show this implies $\mu = 0 \ldots$
- (Leshno-Lin-Pinkus-Schocken '93.)
  If σ a polynomial, $\ldots$;
  else can (roughly) get derivatives of all orders,
  polynomials of all orders.
- (Barron '93.)
  Consider activation $x \mapsto \exp(ia^\mathsf{T} x)$,
  infinite width form $\displaystyle\int \exp(ia^\mathsf{T} x)\widetilde{f}(a)\,\mathrm{d}a.$
  Take real part and sample (Maurey) to get $g \in \mathrm{span}(\mathcal{H}_{\cos})$;
  convert to $\mathrm{span}(\mathcal{H}_\sigma)$ as before.
- (Funahashi '89.) Also Fourier, measure-theoretic.

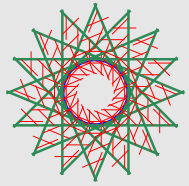
"Universal approximation"

(Uniform approximation of cont. functions on compact sets).

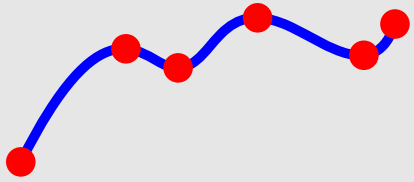
- Elementary proof: RBF (Mhaskar-Michelli '92; BJTX '19).
- Slick proof: Stone-Weierstrass and $\mathcal{H}_{\cos}$ or $\mathcal{H}_{\exp}$ (Hornik-Stinchcombe-White, '89).
- Proof with size estimates beating $\left(\mathrm{Lip}/\epsilon\right)^d$, indeed norm of Fourier transform of gradient, related to "sampling measure": (Barron '93).

Remarks.

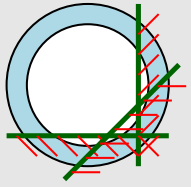
- Exhibits nothing special about DN; indeed, same proofs work for boosting, RBF SVM, ...
- Size estimates huge (soon we'll see $d^{\Omega(d)}$).
- Proofs use nice representation "tricks"; (e.g., Leshno et al "iff not polynomial").



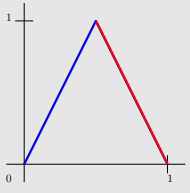
Elementary universal approximation.



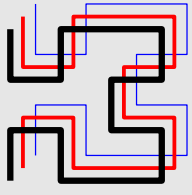
Classical universal approximation.



Benefits of depth.



Sobolev spaces.



Odds & ends.

Radial functions are easy with two ReLU layers

Consider $f(\|x\|^2)$ with Lipschitz constant Lip.

- Pick $h(x) \approx_\epsilon \|x\|_2^2 = \sum_i x_i^2$ with $d \cdot \mathrm{Lip}/\epsilon$ ReLU.

Radial functions are easy with two ReLU layers

Consider $f(\|x\|^2)$ with Lipschitz constant Lip.

- Pick $h(x) \approx_\epsilon \|x\|_2^2 = \sum_i x_i^2$ with $d \cdot \mathrm{Lip}/\epsilon$ ReLU.
- Pick $g \approx_\epsilon f$ with Lip/ϵ ReLU; then

$$\begin{aligned}\Big|f(\|x\|^2) - g(h(x))\Big| &\le \Big|f(\|x\|^2) - f(h(x))\Big| + \Big|f(h(x)) - g(h(x))\Big| \\ &\le \mathrm{Lip}\Big|\|x\|^2 - h(x)\Big| + \epsilon \le 2\epsilon.\end{aligned}$$

Radial functions are easy with two ReLU layers

Consider $f(\|x\|^2)$ with Lipschitz constant Lip.

- Pick $h(x) \approx_\epsilon \|x\|_2^2 = \sum_i x_i^2$ with $d \cdot \mathrm{Lip}/\epsilon$ ReLU.
- Pick $g \approx_\epsilon f$ with Lip/ϵ ReLU; then

$$\begin{aligned}\Big|f(\|x\|^2) - g(h(x))\Big| &\le \Big|f(\|x\|^2) - f(h(x))\Big| + \Big|f(h(x)) - g(h(x))\Big| \\ &\le \mathrm{Lip}\Big|\|x\|^2 - h(x)\Big| + \epsilon \le 2\epsilon.\end{aligned}$$

Remarks.

- Final size of $g \circ h$ is $\mathrm{poly}(\mathrm{Lip}, d, 1/\epsilon)$.
- Proof style is "typical"/lazy;
  (problematically) pays with Lipschitz constant.
- That was easy/intuitive; how about 1 ReLU layer?...

Radial functions are ***not*** easy with only one ReLU layer (I)

Theorem (Eldan-Shamir, 2015).
There exists a radial function f,
expressible with two ReLU layers of width $\mathrm{poly}(d)$,
and a probability measure P
so that every g with a single ReLU layer of width $2^{\mathcal{O}(d)}$ satisfies

$$\int \big(f(x) - g(x)\big)^2 \,\mathrm{d}P(x) \geq \Omega(1).$$

Radial functions are ***not*** easy with only one ReLU layer (I)

Theorem (Eldan-Shamir, 2015).
There exists a radial function f,
expressible with two ReLU layers of width $\mathrm{poly}(d)$,
and a probability measure P
so that every g with a single ReLU layer of width $2^{\mathcal{O}(d)}$ satisfies

$$\int \big(f(x) - g(x)\big)^2 \, \mathrm{d}P(x) \geq \Omega(1).$$

Proof hints.
Apply Fourier isometry and consider the transforms.
Transform of g is supported on a small set of tubes;
transform of f has large mass they can't reach.

Radial functions are ***not*** easy with only one ReLU layer (II)

Theorem (Daniely, 2017).

Let $(x, x') \sim P$ be uniform on two sphere surfaces,
define $h(x, x') = \sin(\pi d^3 x^\top x')$.

For any g with a single ReLU layer
of width $d^{\mathcal{O}(d)}$ and weight magnitude $\mathcal{O}(2^d)$,

$$\int \big(h(x, x') - g(x, x')\big)^2 \, \mathrm{d}P(x, x') \geq \Omega(1),$$

and h can be approximated to accuracy ϵ
by f with two ReLU layers of size $\mathrm{poly}(d, 1/\epsilon)$.

Radial functions are ***not*** easy with only one ReLU layer (II)

Theorem (Daniely, 2017).

Let $(x, x') \sim P$ be uniform on two sphere surfaces,
define $h(x, x') = \sin(\pi d^3 x^\top x')$.

For any g with a single ReLU layer
of width $d^{\mathcal{O}(d)}$ and weight magnitude $\mathcal{O}(2^d)$,

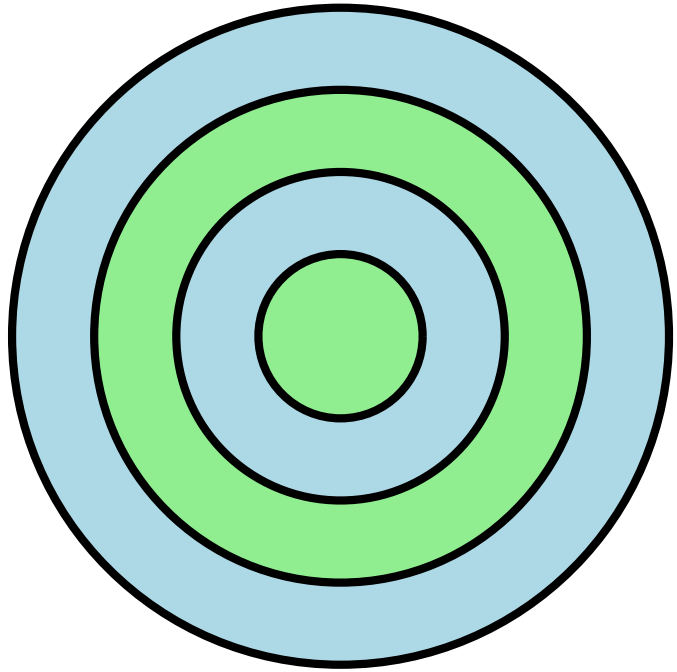
$$\int \big(h(x, x') - g(x, x')\big)^2 \, \mathrm{d}P(x, x') \geq \Omega(1),$$

and h can be approximated to accuracy ϵ
by f with two ReLU layers of size $\mathrm{poly}(d, 1/\epsilon)$.

Proof hints.

Spherical harmonics reduce this to a univariate problem;
apply region counting.

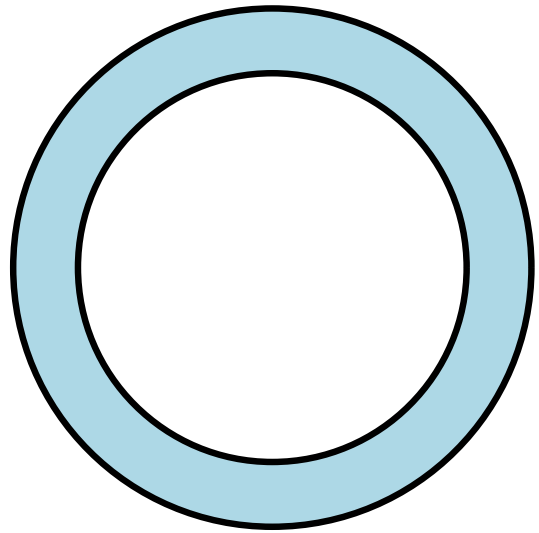
Approximation of high-dimensional radial functions



(A radial function contour plot.)

If we can approximate each shell,
we can approximate the overall function.

Approximation of high-dimensional radial shell

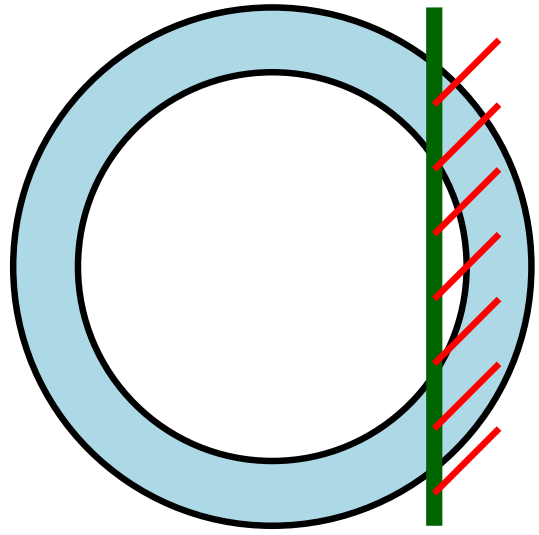


Let's approximate a single shell; consider

$$x \mapsto \mathbb{1}\left[\|x\| \in [1 - 1/d\ ,\ 1]\right],$$

which has a constant fraction of sphere volume.

Approximation of high-dimensional radial shell



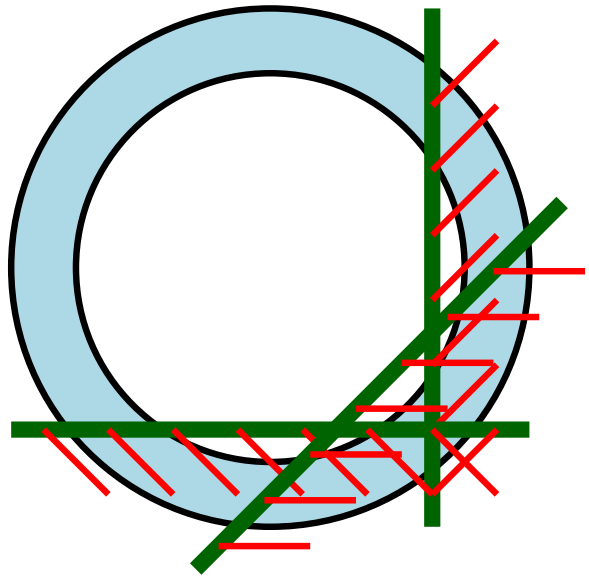
Let's approximate a single shell; consider

$$x \mapsto \mathbb{1}\left[\|x\| \in [1 - 1/d\ ,\ 1]\right],$$

which has a constant fraction of sphere volume.

Can't cut too deeply; get bad error on inner zero part...

Approximation of high-dimensional radial shell



Let's approximate a single shell; consider

$$x \mapsto \mathbb{1}\left[\|x\| \in [1 - 1/d\ ,\ 1]\right],$$

which has a constant fraction of sphere volume.

Can't cut too deeply; get bad error on inner zero part...

...but then we need to cover exponentially many caps.

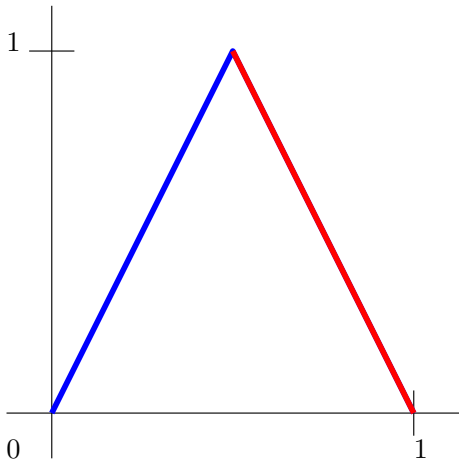
Let's go back to the drawing board;
what do shallow representations do exceptionally badly?

Let's go back to the drawing board;
what do shallow representations do exceptionally badly?

One weakness: their complexity scales with #bumps.

Consider the **tent map**

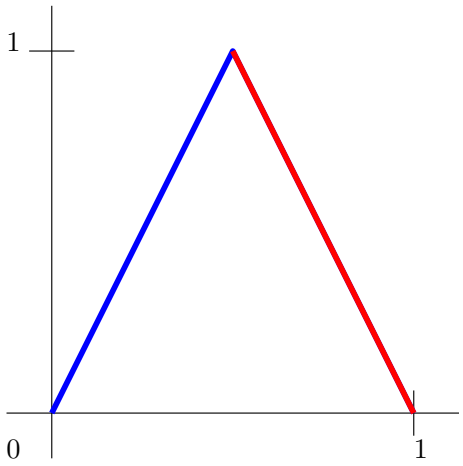
$$\Delta(x) := \sigma_{\mathrm{r}}(2x) - \sigma_{\mathrm{r}}(4x-2) = \begin{cases} 2x & x \in [0, 1/2), \\ 2(1-x) & x \in [1/2, 1]. \end{cases}$$



Δ.

Consider the **tent map**

$$\Delta(x) := \sigma_{\mathrm{r}}(2x) - \sigma_{\mathrm{r}}(4x-2) = \begin{cases} 2x & x \in [0, 1/2), \\ 2(1-x) & x \in [1/2, 1]. \end{cases}$$



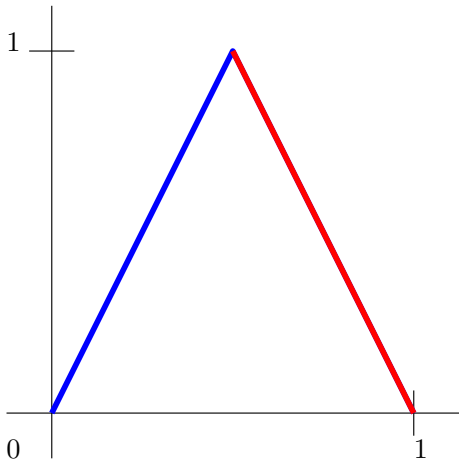
Δ.

What is the effect of composition?

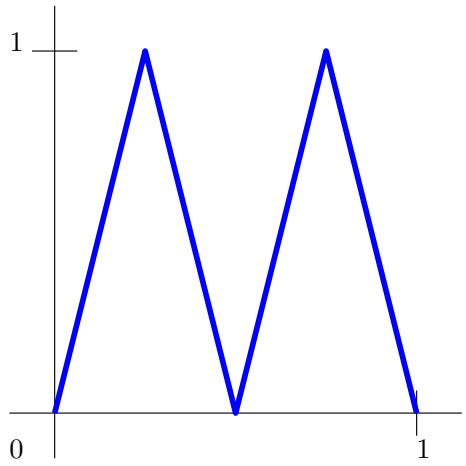
$$f(\Delta(x)) = \begin{cases} x \in [0, 1/2) & \Longrightarrow \quad f(2x) = f \text{ squeezed into } [0, 1/2], \\ x \in [1/2, 1] & \Longrightarrow \quad f(2(1-x)) = f \text{ reversed, squeezed.} \end{cases}$$

Consider the **tent map**

$$\Delta(x) := \sigma_{\mathrm{r}}(2x) - \sigma_{\mathrm{r}}(4x-2) = \begin{cases} 2x & x \in [0, 1/2), \\ 2(1-x) & x \in [1/2, 1]. \end{cases}$$



Δ.



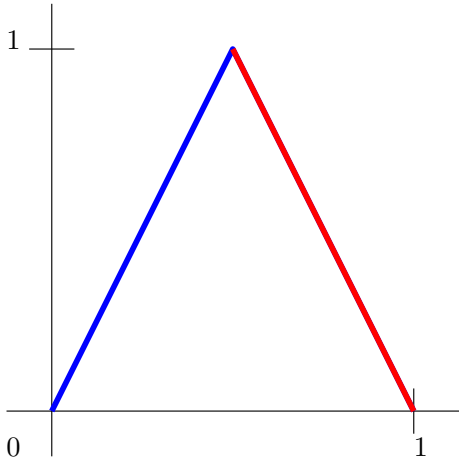
$\Delta^2 = \Delta \circ \Delta$.

What is the effect of composition?

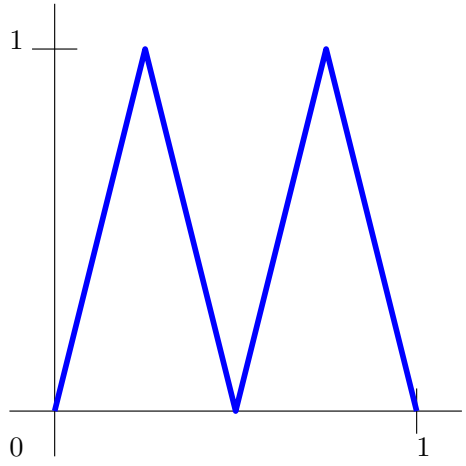
$$f(\Delta(x)) = \begin{cases} x \in [0, 1/2) & \Longrightarrow \quad f(2x) = f \text{ squeezed into } [0, 1/2], \\ x \in [1/2, 1] & \Longrightarrow \quad f(2(1-x)) = f \text{ reversed, squeezed.} \end{cases}$$

Consider the **tent map**

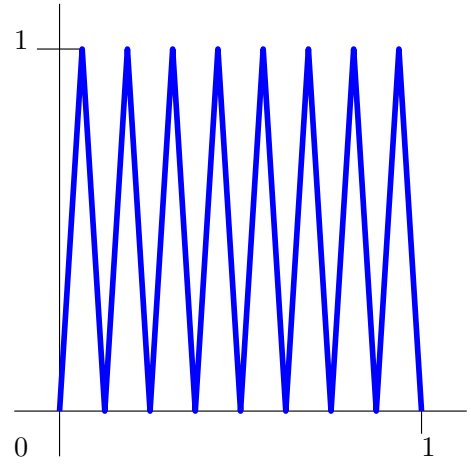
$$\Delta(x) := \sigma_{\mathrm{r}}(2x) - \sigma_{\mathrm{r}}(4x-2) = \begin{cases} 2x & x \in [0, 1/2), \\ 2(1-x) & x \in [1/2, 1]. \end{cases}$$



Δ.



$\Delta^2 = \Delta \circ \Delta$.



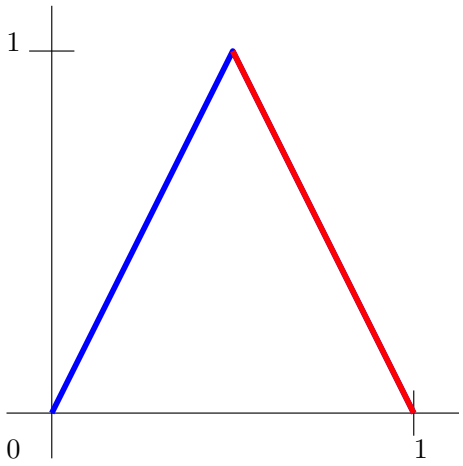
Δ^k.

What is the effect of composition?

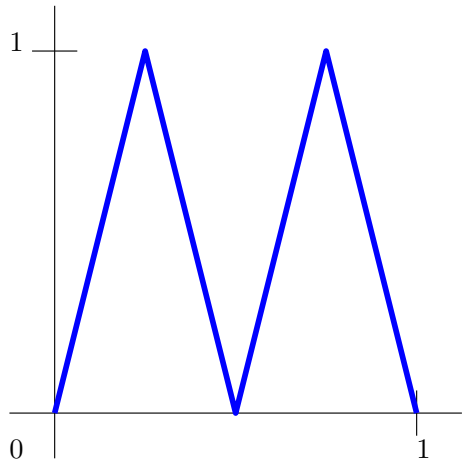
$$f(\Delta(x)) = \begin{cases} x \in [0, 1/2) & \Longrightarrow \quad f(2x) = f \text{ squeezed into } [0, 1/2], \\ x \in [1/2, 1] & \Longrightarrow \quad f(2(1-x)) = f \text{ reversed, squeezed.} \end{cases}$$

Consider the **tent map**

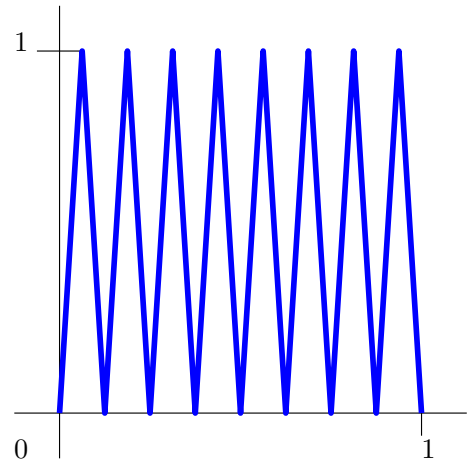
$$\Delta(x) := \sigma_{\mathrm{r}}(2x) - \sigma_{\mathrm{r}}(4x-2) = \begin{cases} 2x & x \in [0, 1/2), \\ 2(1-x) & x \in [1/2, 1]. \end{cases}$$



Δ.



$\Delta^2 = \Delta \circ \Delta$.



Δ^k.

What is the effect of composition?

$$f(\Delta(x)) = \begin{cases} x \in [0, 1/2) & \Longrightarrow \quad f(2x) = f \text{ squeezed into } [0, 1/2], \\ x \in [1/2, 1] & \Longrightarrow \quad f(2(1-x)) = f \text{ reversed, squeezed.} \end{cases}$$

Δ^k uses $\mathcal{O}(k)$ layers & nodes, but has $\mathcal{O}(2^k)$ bumps.

Theorem (T '15).
Let #layers $k \geq 1$ be given.

Theorem (T '15).

Let #layers $k \geq 1$ be given.

Exists ReLU network $f : [0,1] \to [0,1]$

with 4 distinct parameters, $3k^2 + 9$ nodes, $2k^2 + 6$ layers,

Theorem (T '15).

Let #layers $k \geq 1$ be given.

Exists ReLU network $f : [0,1] \to [0,1]$

with 4 distinct parameters, $3k^2 + 9$ nodes, $2k^2 + 6$ layers,

such that every ReLU network $g : \mathbb{R}^d \to \mathbb{R}$

with $\leq k$ layers, $\leq 2^k$ nodes

Theorem (T '15).

Let #layers $k \geq 1$ be given.

Exists ReLU network $f : [0,1] \to [0,1]$

with 4 distinct parameters, $3k^2 + 9$ nodes, $2k^2 + 6$ layers,

such that every ReLU network $g : \mathbb{R}^d \to \mathbb{R}$

with $\leq k$ layers, $\leq 2^k$ nodes

satisfies

$$\int_{[0,1]} |f(x) - g(x)| \, \mathrm{d}x \geq \frac{1}{32}.$$

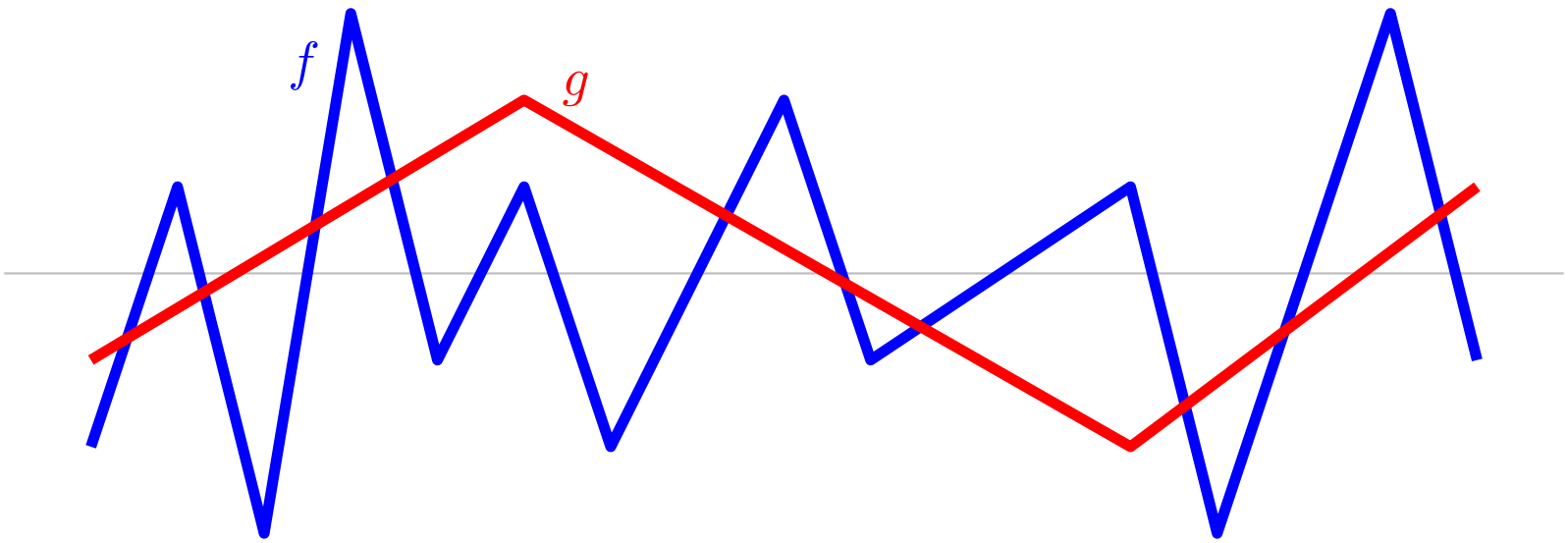
Theorem (T '15).
Let #layers $k \geq 1$ be given.
Exists ReLU network $f : [0,1] \to [0,1]$
with 4 distinct parameters, $3k^2 + 9$ nodes, $2k^2 + 6$ layers,
such that every ReLU network $g : \mathbb{R}^d \to \mathbb{R}$
with $\leq k$ layers, $\leq 2^k$ nodes
satisfies

$$\int_{[0,1]} |f(x) - g(x)| \, \mathrm{d}x \geq \frac{1}{32}.$$

Proof.

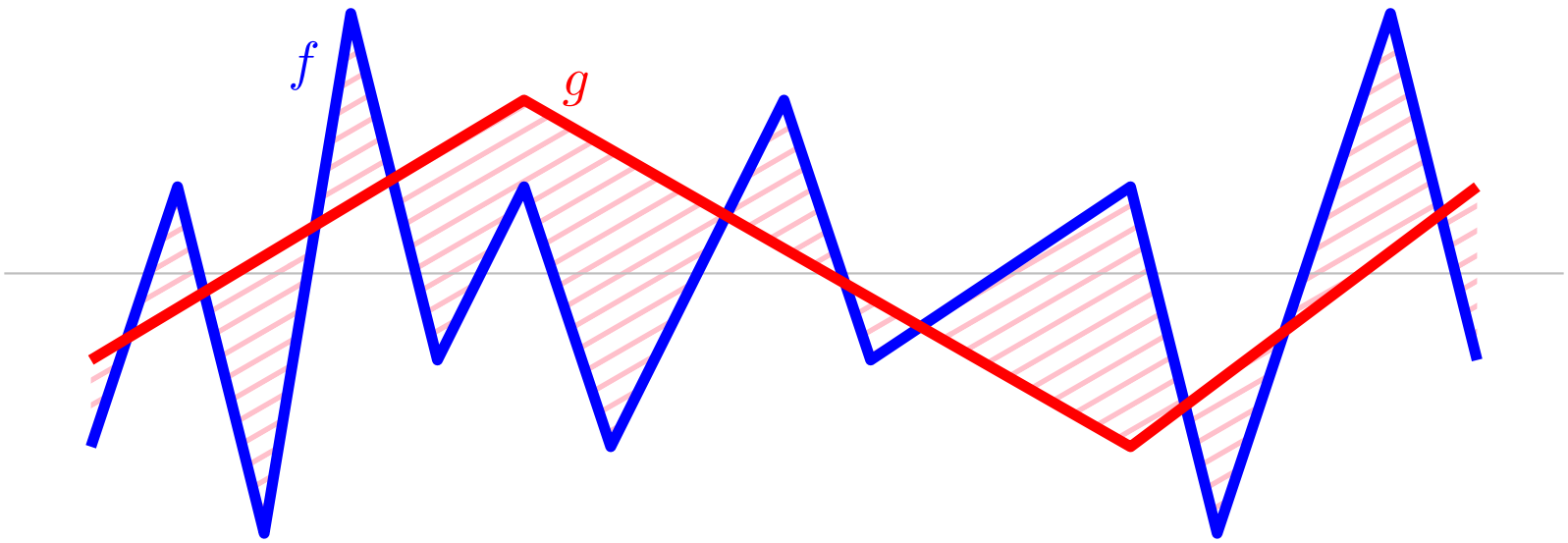
1. g with few oscillations can't apx oscillatory regular f.
2. There exists a regular, oscillatory f. ($f = \Delta^{k^2+3}$.)
3. Width m depth $L \implies$ few ($\mathcal{O}(m^L)$) oscillations.
   This final step rediscovered many times;
   (T '15) gives elementary univariate argument;
   multivariate arguments in (Warren '68), (Arnold ?),
   (Montufar, Pascanu, Cho, Bengio, '14), (BT '18), ...

g with few oscillations;
f highly oscillatory $\Longrightarrow$ $\int_{[0,1]} |g - f|$ large .



g with few oscillations;
f highly oscillatory

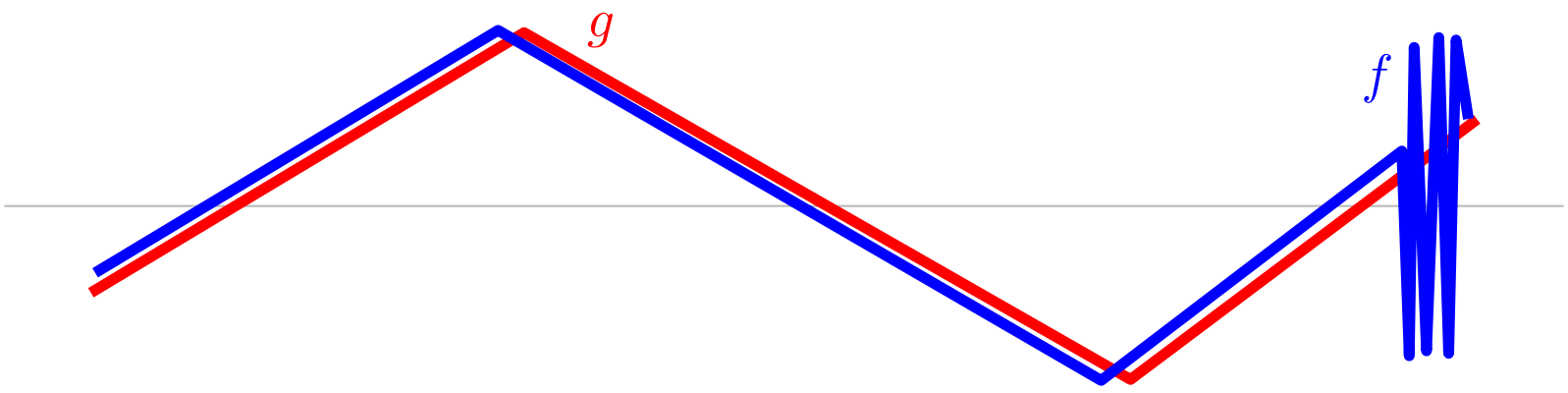
$$\Longrightarrow \qquad \int_{[0,1]} |g - f| \quad \text{large .}$$



g with few oscillations;
f highly oscillatory

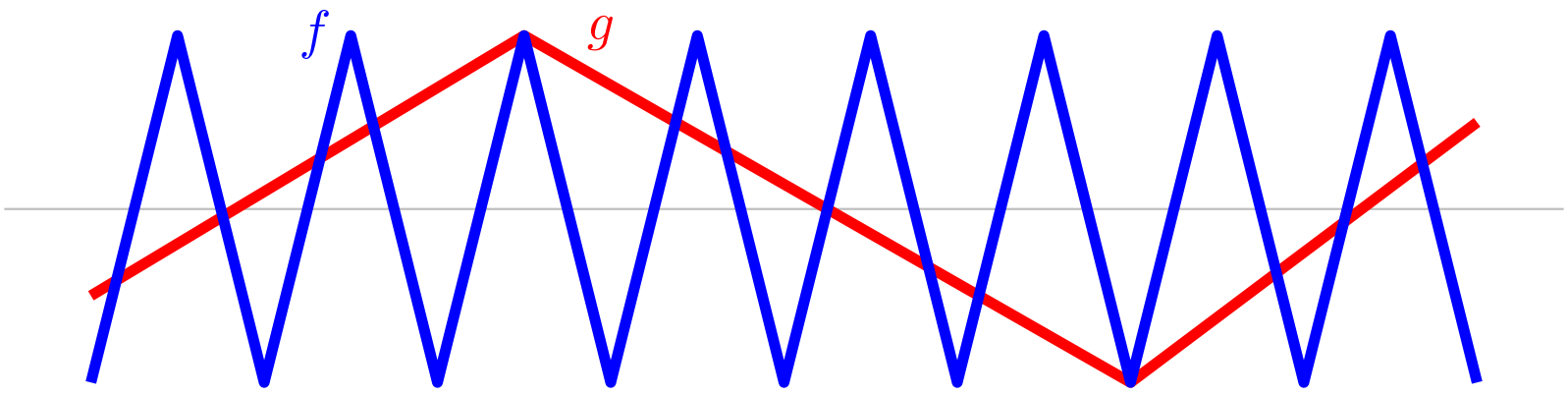
$\overset{?}{\Longrightarrow}$

$\int_{[0,1]} |g - f|$ large .



g with few oscillations;
f highly oscillatory, *regular*

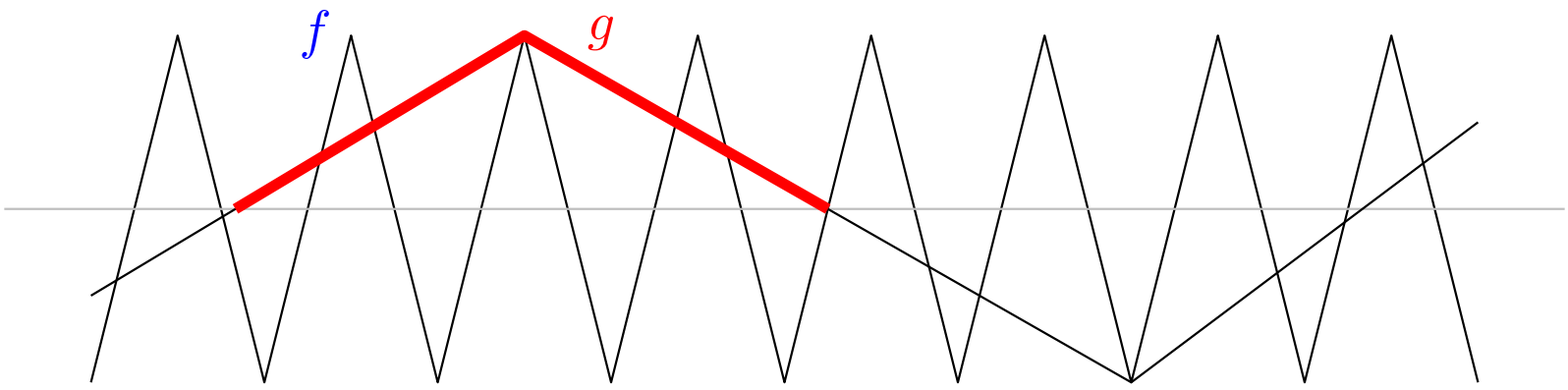
$$\Longrightarrow \qquad \int_{[0,1]} |g - f| \quad \text{large} .$$



Let's use $f = \Delta^{k^2+3}$.

g with few oscillations;
f highly oscillatory, *regular*

$\Longrightarrow$ $\int_{[0,1]} |g - f|$ large .

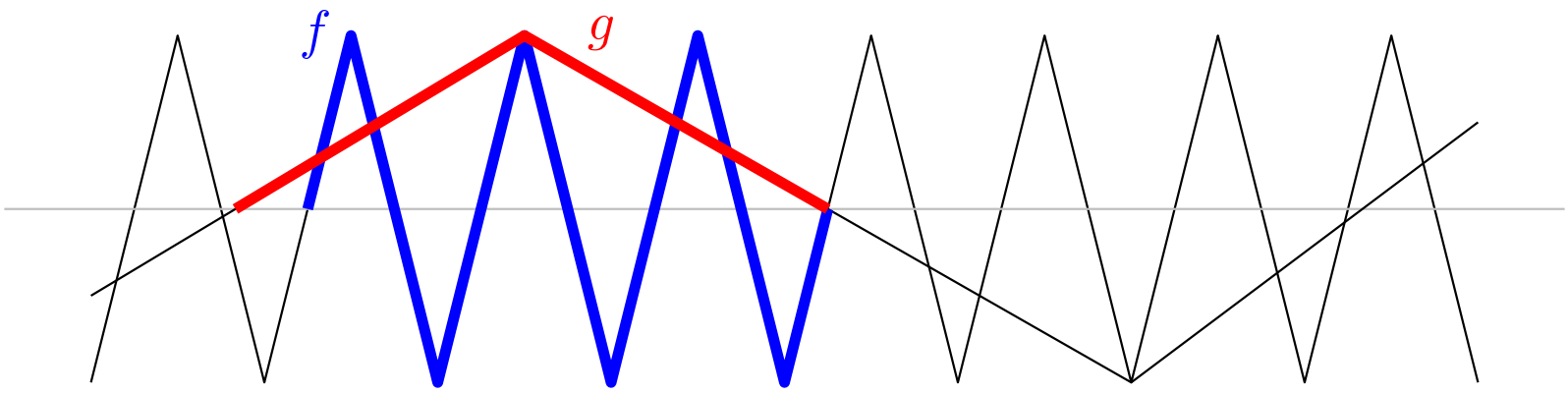


Let's use $f = \Delta^{k^2+3}$.

g with few oscillations;
f highly oscillatory, *regular*

$\Longrightarrow$

$$\int_{[0,1]} |g - f| \quad \text{large .}$$

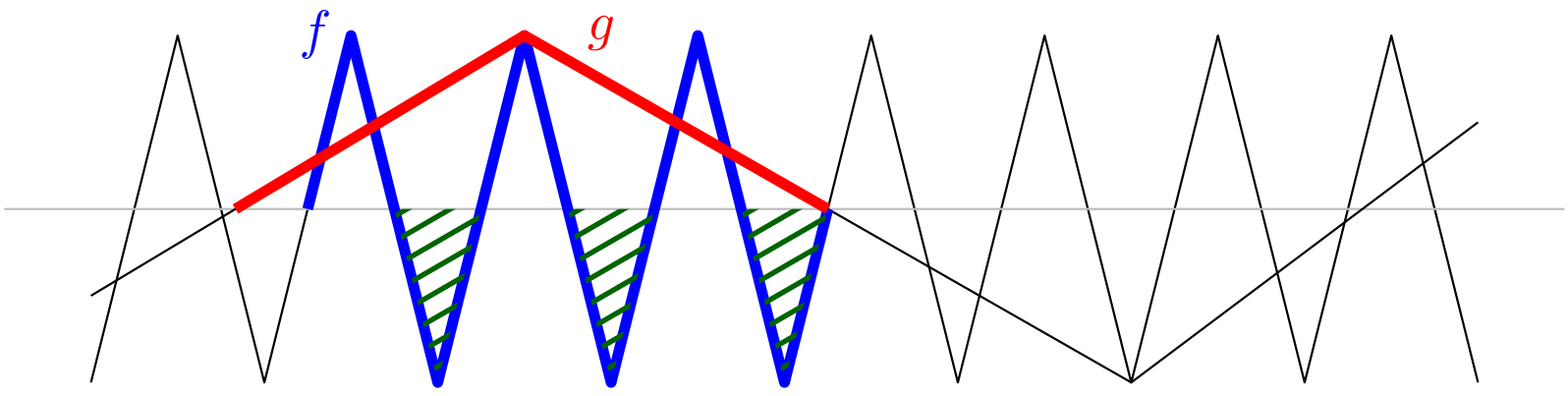


Let's use $f = \Delta^{k^2+3}$.

g with few oscillations;
f highly oscillatory, *regular*

$\Longrightarrow$

$$\int_{[0,1]} |g - f| \quad \text{large} .$$

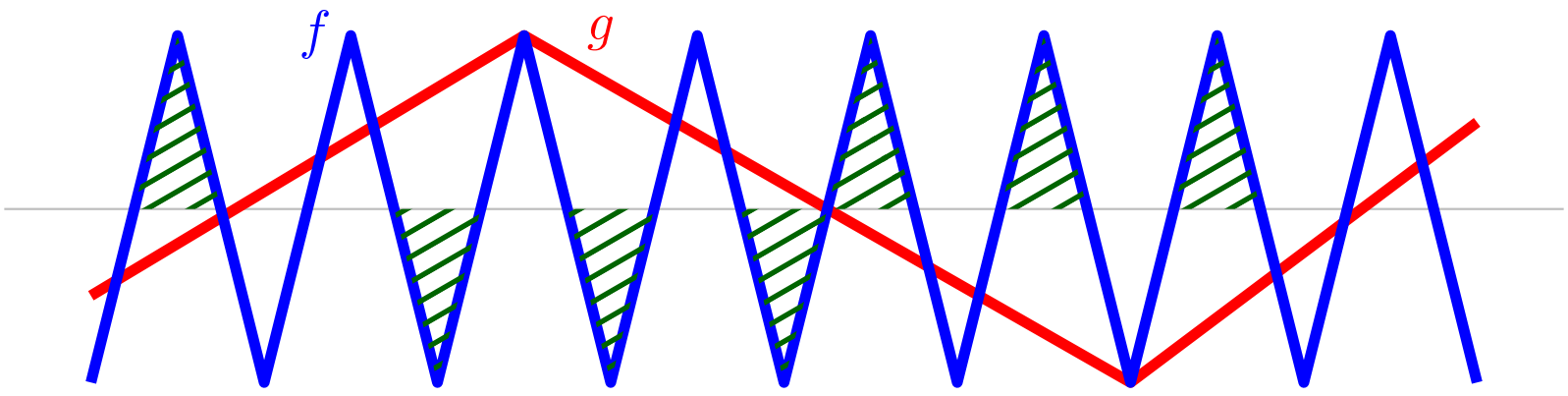


Let's use $f = \Delta^{k^2+3}$.

g with few oscillations;
f highly oscillatory, *regular*

$\Longrightarrow$

$$\int_{[0,1]} |g - f| \quad \text{large .}$$



Let's use $f = \Delta^{k^2+3}$.

Story from *benefits of depth*:

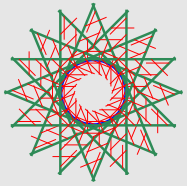
- Certain radial functions have
  polynomial width 2 ReLU layer representation,
  exponential width 1 ReLU layer representation.
- Δ^{k^2+3} can be written with $\mathcal{O}(k^2)$ depth and $\mathcal{O}(1)$ width, requires width $\Omega(2^k)$ if depth $\mathcal{O}(k)$.

Story from *benefits of depth*:

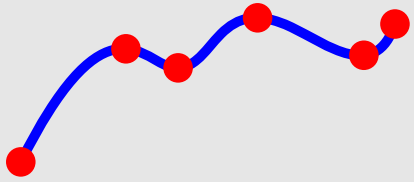
- Certain radial functions have
  polynomial width 2 ReLU layer representation,
  exponential width 1 ReLU layer representation.
- Δ^{k^2+3} can be written with $\mathcal{O}(k^2)$ depth and $\mathcal{O}(1)$ width, requires width $\Omega(2^k)$ if depth $\mathcal{O}(k)$.

Remarks.

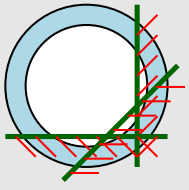
- Δ^k is 2^k-Lipschitz;
  possibly nonsensical, unrealistic.
- These results have stood a few years now;
  many "technical" questions,
  also "realistic" questions.



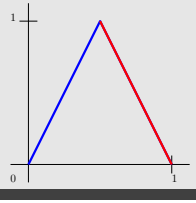
Elementary universal approximation.



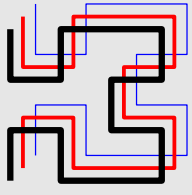
Classical universal approximation.



Benefits of depth.

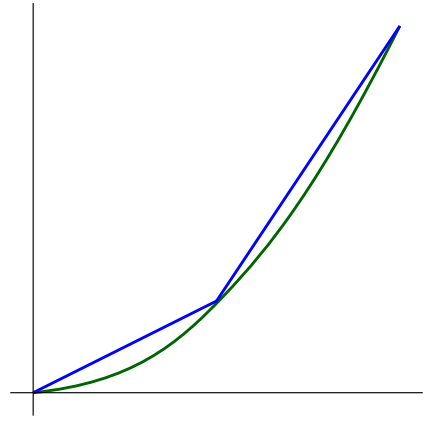


Sobolev spaces.



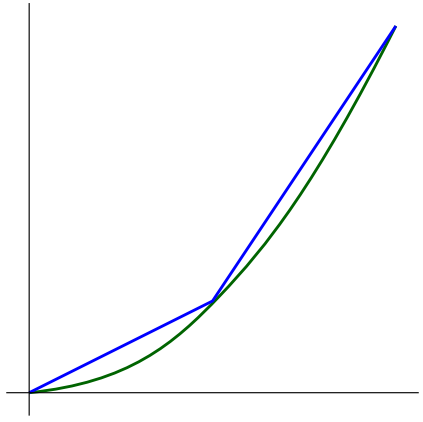
Odds & ends.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.

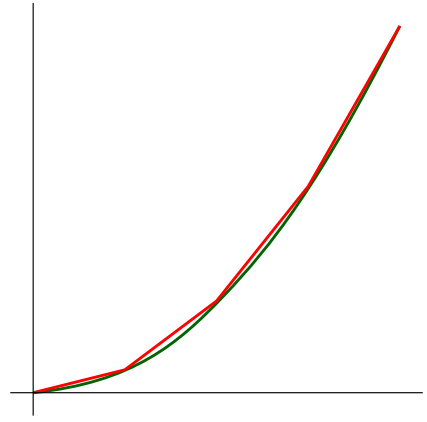


h_1.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.

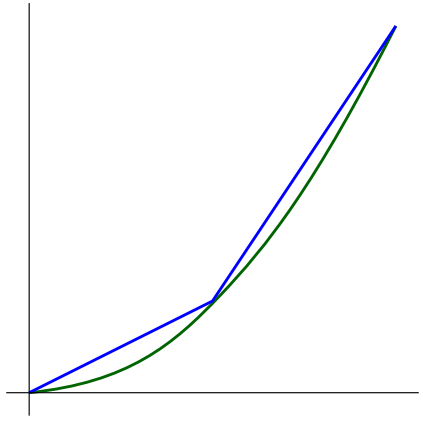


h_1.

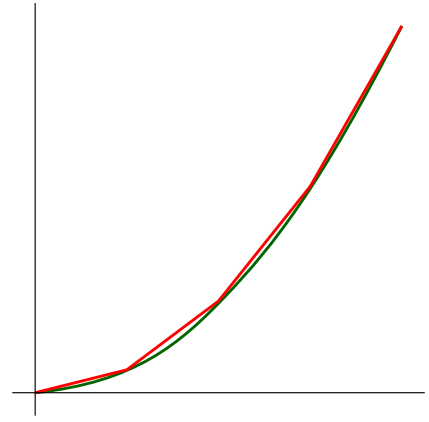


h_2.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



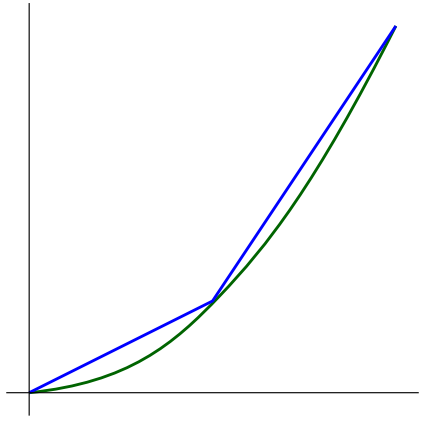
h_1.



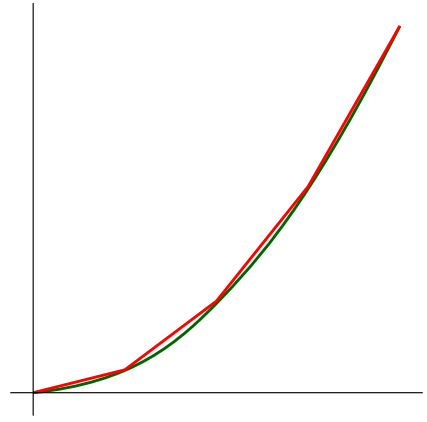
h_2.

$h_1 - h_2$.

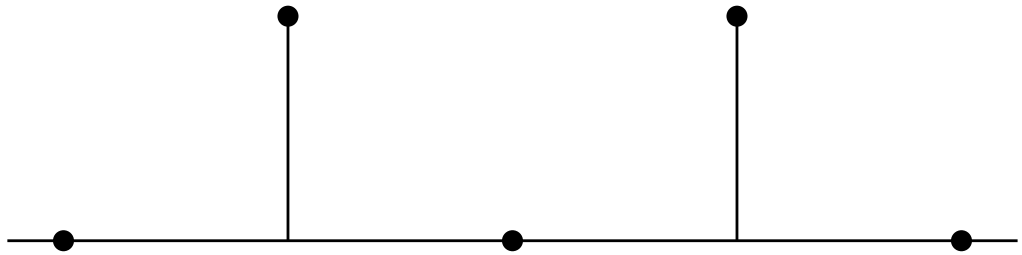
$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



h_1.

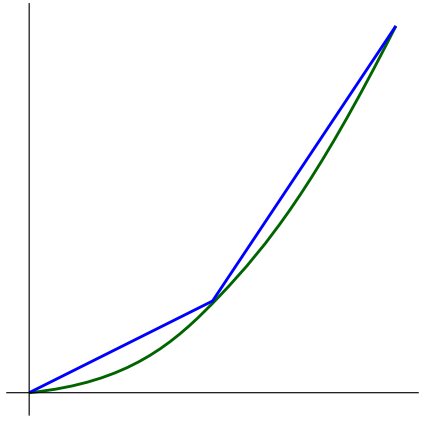


h_2.

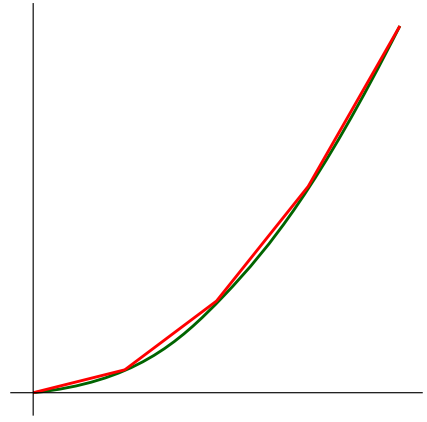


$h_1 - h_2$.

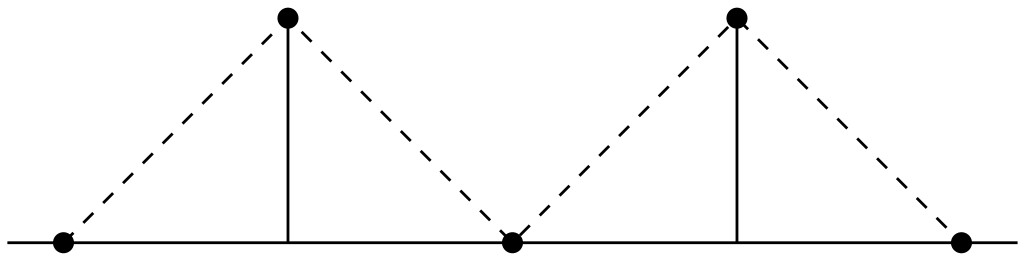
$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



h_1.

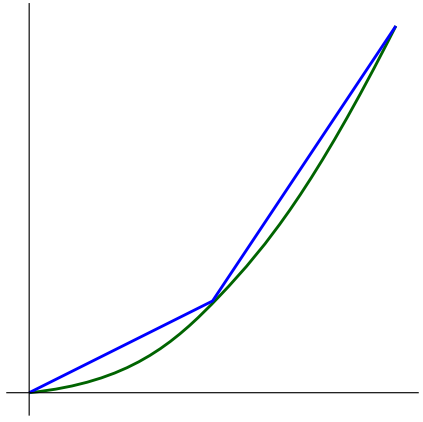


h_2.

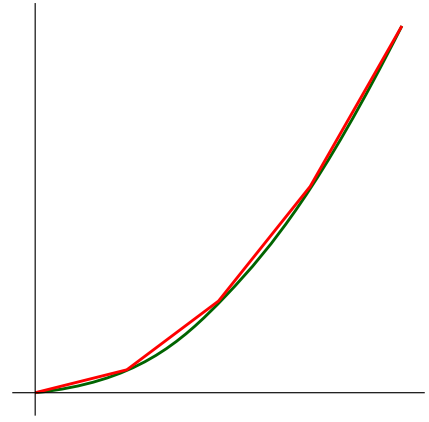


$h_1 - h_2$.

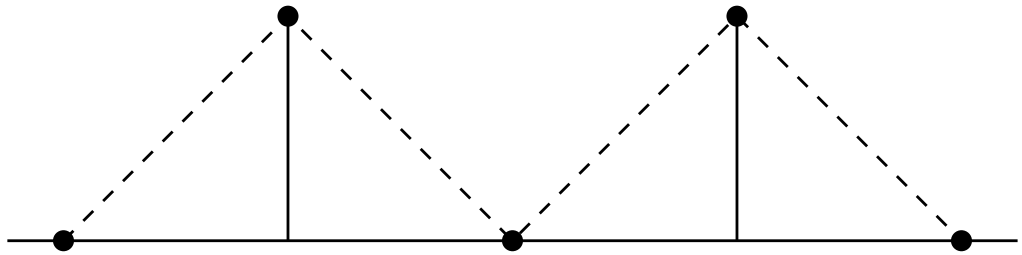
$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



h_1.



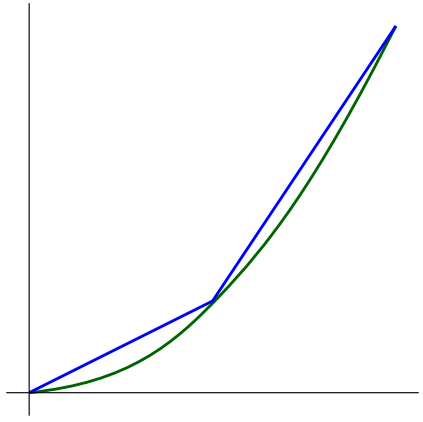
h_2.



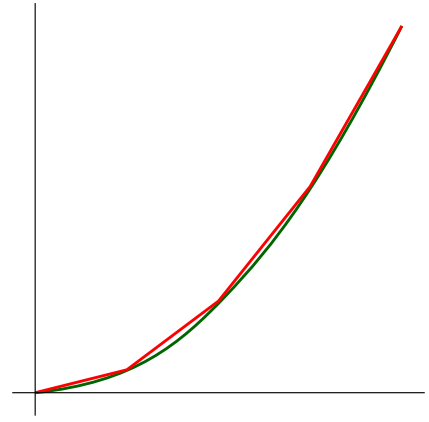
$h_1 - h_2$.

Thus $h_k(x) = x - \sum_{i \le k} \Delta^i(x)/4^i$.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



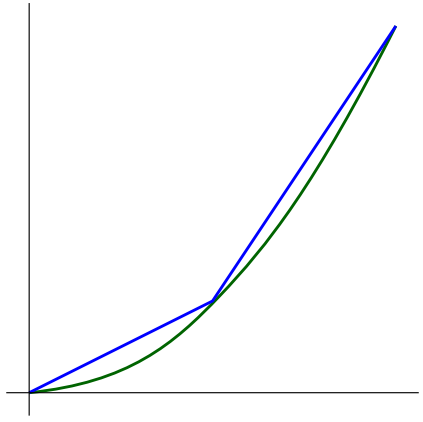
h_1.



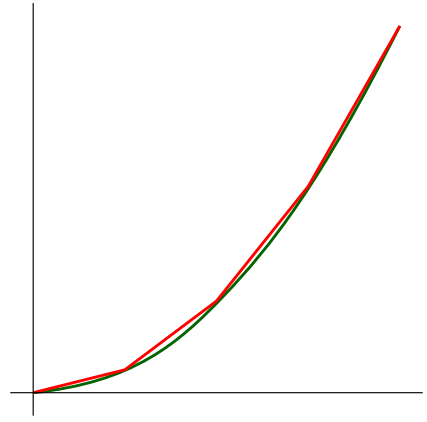
h_2.

Thus $h_k(x) = x - \sum_{i \le k} \Delta^i(x)/4^i$.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



h_1.

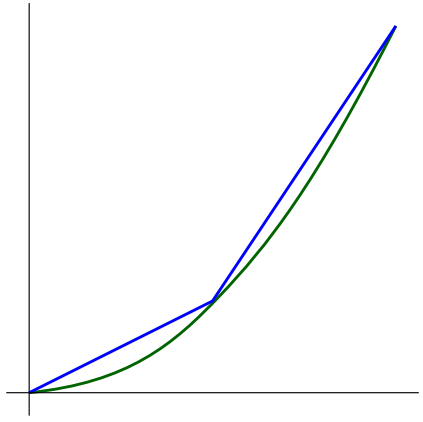


h_2.

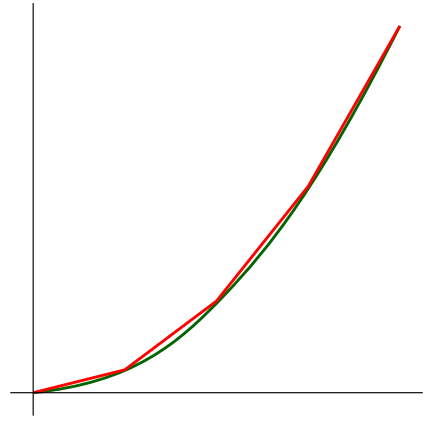
Thus $h_k(x) = x - \sum_{i \le k} \Delta^i(x)/4^i$.

- h_k needs $k = \mathcal{O}(\ln(1/\epsilon))$ to ϵ-apx $x \mapsto x^2$ (Yarotsky, '16), with matching lower bounds.
- Squaring implies **multiplication** via polarization: $x^\intercal y = \frac{1}{2}\left(\|x+y\|^2 - \|x\|^2 - \|y\|^2\right)$.

$h_k :=$ piecewise-affine interpolation of x^2 at $\{0, \frac{1}{2^k}, \frac{2}{2^k}, \ldots, \frac{2^k}{2^k}\}$.



h_1.



h_2.

Thus $h_k(x) = x - \sum_{i \le k} \Delta^i(x)/4^i$.

- h_k needs $k = \mathcal{O}(\ln(1/\epsilon))$ to ϵ-apx $x \mapsto x^2$ (Yarotsky, '16), with matching lower bounds.
- Squaring implies **multiplication** via polarization: $x^\intercal y = \frac{1}{2}\left(\|x+y\|^2 - \|x\|^2 - \|y\|^2\right)$.
- This implies efficient approximation of polynomials; can we do more?

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Proof.
Conditions imply accurate local Taylor expansions.
Therefore can write f as a linear combination of this basis:

polynomials multiplied by local bumps.

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

- There is depth, but it is function independent: only the basis coefficients use f.
- In a sense, this is a shallow representation.
- Lipschitz constant is possibly bad: $\Delta^{1/\epsilon}$ is $1/\epsilon$-Lipschitz, the bumps are $1/\epsilon^{d/r}$-Lipschitz.

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

- There is parallel and subsequent work with similar proof ideas and Lipschitz constants:
  (Safran-Shamir '16), (Petersen-Voigtlaender '17), (Schmidt-Hieber '17).
- Another appearance of polynomials in DN:
  Sum-product networks.
  These were the first to have depth separation
  (Delalleau-Bengio '11).

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

- DN can approximate polynomials efficiently,
  but the reverse is false:
  a single ReLU requires degree $1/\epsilon$.
- Polynomials can not handle flat regions well;
  this is used above,
  and in approximating rational functions (T '17).

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

Theorem (Yarotsky '16).
Let dimension d and smoothness order r be given. Given $f : [0,1]^d \to \mathbb{R}$, all rth order derivatives bounded by 1, exists a network g

with $C_{d,r} \ln(e/\epsilon)$ layers and $C_{d,r}\epsilon^{-d/r} \ln(e/\epsilon)$ nodes

so that

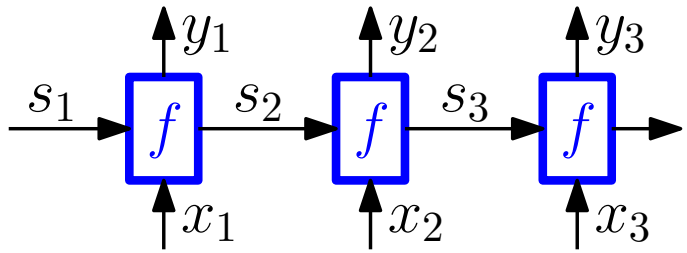
$$\sup_{x\in[0,1]^d} |f(x) - g(x)| \leq \epsilon.$$

Remarks.

- Corresponding lower bounds indicate depth is needed.

Interlude: three questions

1. Are fixed DN architectures closed under addition?
   No, add together perturbed copies of Δ^k.
2. Can RNNs model Turing Machines?

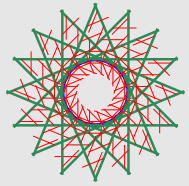


   Hint. ReLU networks can do exact Boolean formulae.
   Set f to state transition table,
   encode tape on s.

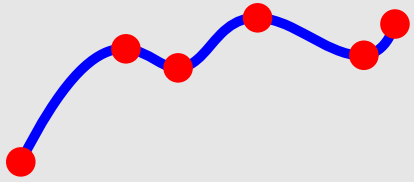
3. Given continuous $g : \mathbb{R}^d \to \mathbb{R}$,
   can we construct custom univariate activations so that

$$g(x) \stackrel{!}{=} \sum_{i=0}^{2d} f_i \left(\sum_{j=1}^{d} h_{i,j}(x_j) \right) ?$$

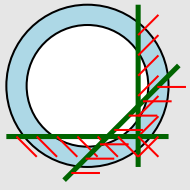
   Hint? Contradicts a Hilbert problem?



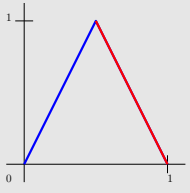
Elementary universal approximation.



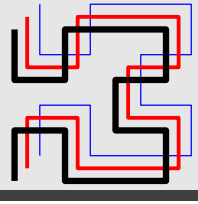
Classical universal approximation.



Benefits of depth.



Sobolev spaces.



Odds & ends.

Generative modeling

Typical setup: *pushforward measure* $g\#\mu$, meaning

$$\text{sample } x \sim \mu, \qquad \text{output } g(x).$$

Many easy constructions have bad/∞ Lipschitz constants!
E.g., mapping uniform into $[0, 1/2]$, $(3/2, 2]$.

Some literature:
(Arora-Ge-Liang-Ma-Zhang '17, BT '18, Bai-Ma-Risteski '19).

Randomly *initialized* networks

Approximation fact in recent optimization papers:
a small perturbation of random initialization
gives any function you want!
(Du-Lee-Li-Wang-Zhai '18, AllenZhu-Li-Song '18).

There is residual error from the noise;
approximating high-Lipschitz functions is problematic!
(BJTX '19.)

Randomly *sampled* networks

Theorem. With probability $\geq 1 - 1/e$,

$$\sup_{\|x\|_2 \leq 1} \left| \int \sigma_{\mathrm{r}}(a^{\mathsf{T}} x - b) \, \mathrm{d}\mu(a,b) - \frac{\|\mu\|_1}{N} \sum_{i=1}^{N} \sigma_{\mathrm{r}}(a_i^{\mathsf{T}} x - b_i) \right|$$
$$\leq \mathcal{O}\left(\frac{B\|\mu\|_1}{\sqrt{N}} \right),$$

where support of μ has $\|(a,b)\| \leq B$.

Randomly *sampled* networks

Theorem. With probability $\geq 1 - 1/e$,

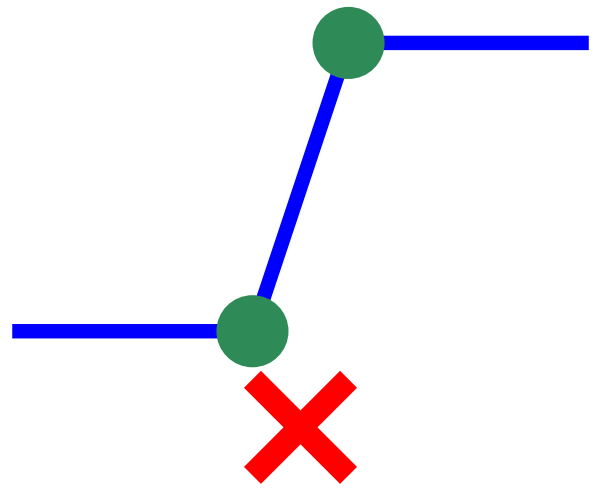
$$\sup_{\|x\|_2 \leq 1} \left| \int \sigma_{\mathrm{r}}(a^{\mathsf{T}} x - b) \, \mathrm{d}\mu(a,b) - \frac{\|\mu\|_1}{N} \sum_{i=1}^{N} \sigma_{\mathrm{r}}(a_i^{\mathsf{T}} x - b_i) \right|$$
$$\leq \mathcal{O}\left(\frac{B\|\mu\|_1}{\sqrt{N}} \right),$$

where support of μ has $\|(a,b)\| \leq B$.

Proof. Invoke Rademacher complexity,
but swap inputs and parameters.

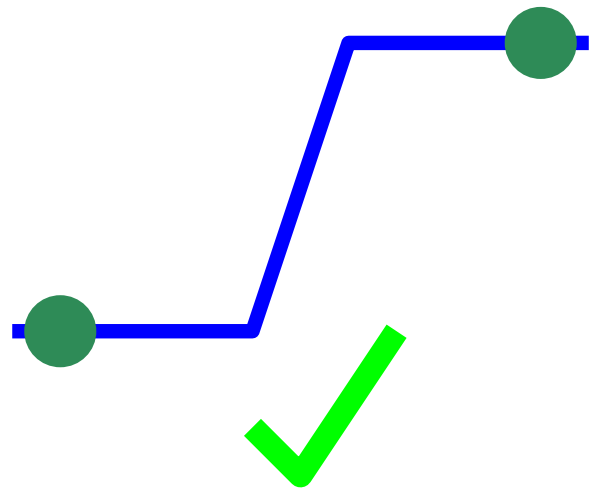
(Koiran-Gurvits '97, Sun-Gilbert-Tewari '18, BJTX '19.)
Also Maurey's Lemma (Barron '93).

Adversarial stability



Adversarial examples lower bound the Lipschitz constant. . .

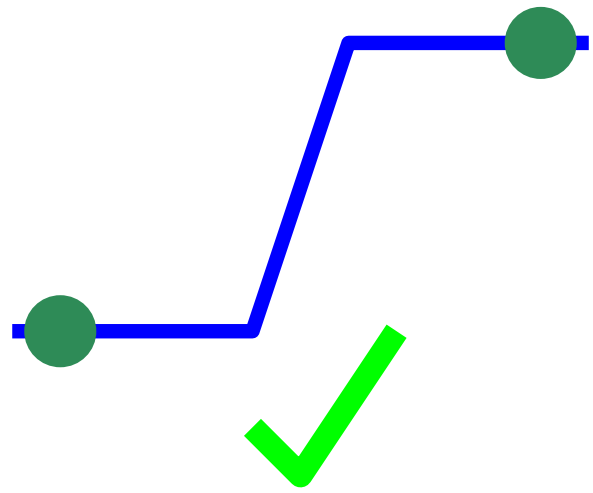
Adversarial stability



Adversarial examples lower bound the Lipschitz constant. . .

. . . but a bad Lipschitz constant
can be good for adversarial examples!

Adversarial stability

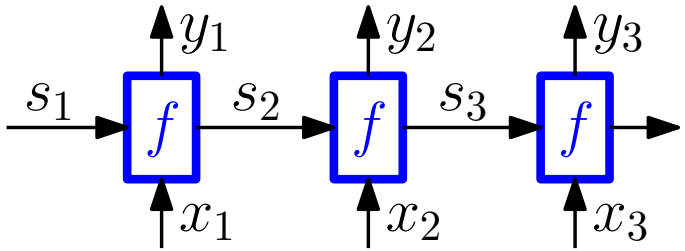


Adversarial examples lower bound the Lipschitz constant. . .

. . . but a bad Lipschitz constant
can be good for adversarial examples!

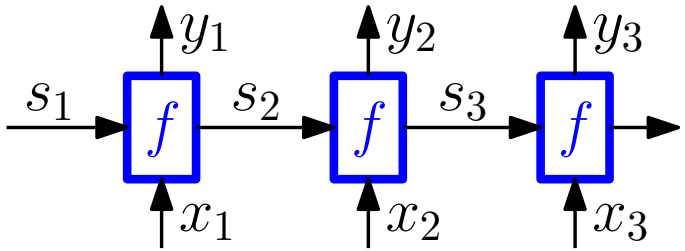
Given the existence of adversarial examples,
uniform approximation too stringent?

Turing machines and RNNs (Siegelmann-Sontag '94)



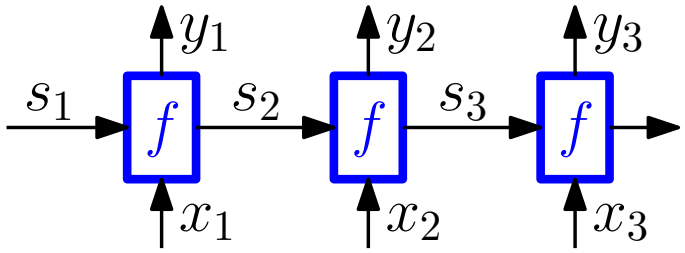
- Make f the TM state transition table, s the tape.

Turing machines and RNNs (Siegelmann-Sontag '94)

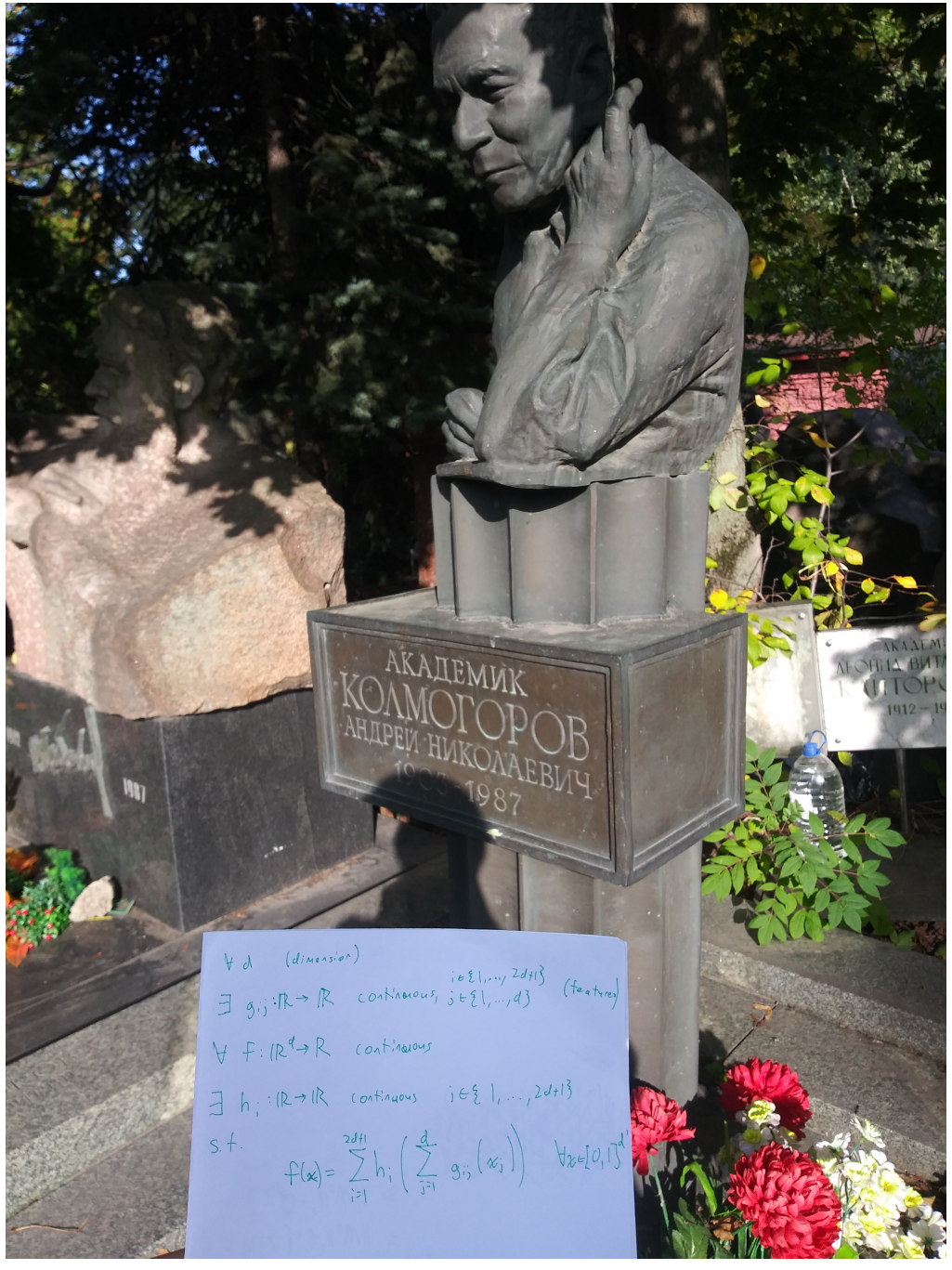


- Make f the TM state transition table, s the tape.
- $x \mapsto \mathbb{1}[x \geq 0]$ is **not computable**; bits need a special encoding within s.

Turing machines and RNNs (Siegelmann-Sontag '94)



- Make f the TM state transition table, s the tape.
- $x \mapsto \mathbb{1}[x \geq 0]$ is **not computable**; bits need a special encoding within s.
- Use a **robust "cantor-like" encoding**.


АКАДЕМИК
КОЛМОГОРОВ
АНДРЕЙ НИКОЛАЕВИЧ
1987
1987
$\forall\, d$ (dimension)
$\exists\ g_{ij}: \mathbb{R} \to \mathbb{R}$ continuous, $i \in \{1, \dots, 2d+1\}$, $j \in \{1, \dots, d\}$ (features)
$\forall\ f: \mathbb{R}^d \to \mathbb{R}$ continuous
$\exists\ h_i: \mathbb{R} \to \mathbb{R}$ continuous $i \in \{1, \dots, 2d+1\}$
s.t.
$f(x) = \sum_{i=1}^{2d+1} h_i\left(\sum_{j=1}^{d} g_{ij}(x_j)\right)$ $\forall x \in [0,1]^d$

Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

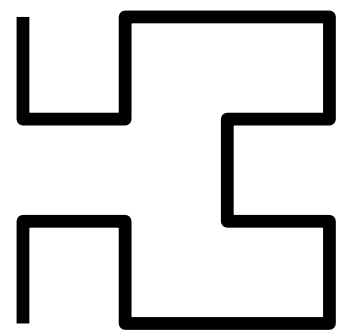
Step 1.
Fix target accuracy $\epsilon > 0$.

Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

Step 2.
Choose $f : \mathbb{R} \to \mathbb{R}$,
nearly injective $Q : \mathbb{R}^d \to \mathbb{R}$,
$g \approx f(Q(x))$

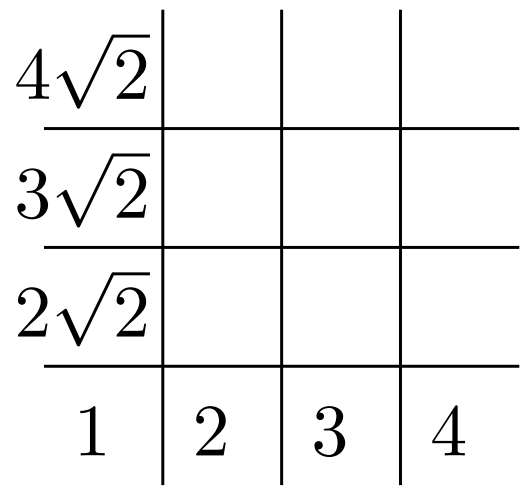


Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

Step 3.
Replace near-injection $Q : \mathbb{R}^d \to \mathbb{R}$
with $\sum_j h_j(x_j)$.

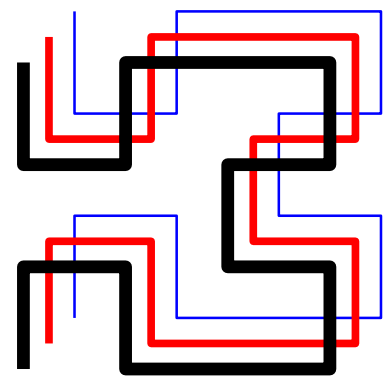


Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

Step 4.
Replace $f(\sum_j h_j(x_j))$
with staggered versions $\sum_i f_i(\sum_j h_{i,j}(x_j))$;
for any $x \in [0,1]^d$,
$\geq$ half are correct.

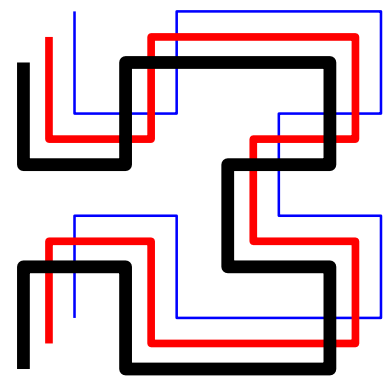


Kolmogorov-Arnold '56

There exist continuous $((h_{i,j})_{i=0}^{2d})_{j=1}^{d} : \mathbb{R} \to \mathbb{R}$,
so that for any continuous $g : \mathbb{R}^d \to \mathbb{R}$,
there exist continuous $(f_i)_{i=0}^{2d} : \mathbb{R} \to \mathbb{R}$
with

$$g(x) = \sum_{i=0}^{2d} f_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big).$$

Step 5.
Embed the solutions for infinitely many ϵ
into one.

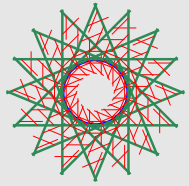


Main story.

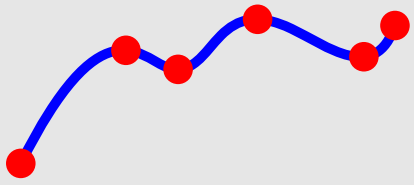
- Can fit continuous functions in various ways;
  the size is bad ($\left(d \cdot \mathrm{Lip}/\epsilon\right)^{\mathcal{O}(d)}$).
- Composition and depth bring some concrete benefits;
  exponential reductions in width!
- Polynomials may be efficiently approximated,
  but also some non-polynomials
  (Sobolov balls, rational functions, flat regions, ...).

Remarks.

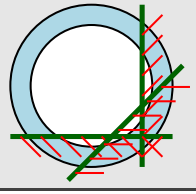
- Infinite width is a useful and common theme.
- Refined depth separations (e.g., a single new layer)
  and practical depth separations
  are still elusive.
- Refined, average-case complexity measures are elusive.



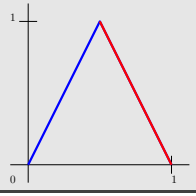
Elementary universal approximation.



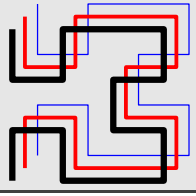
Classical universal approximation.



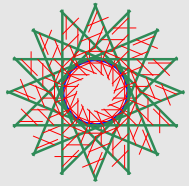
Benefits of depth.



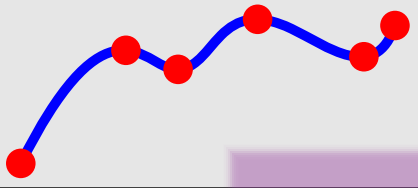
Sobolev spaces.



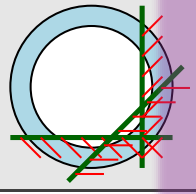
Odds & ends.



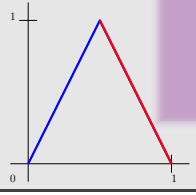
Elementary universal approximation.



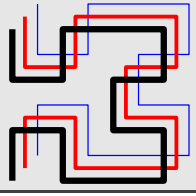
Classical universal approximation.



Thanks. . . any questions?



Sobolev spaces.



Odds & ends.