Felix Voigtländer

Gitta Kutyniok

Morten Nielsen

Approximation with deep networks

Rémi Gribonval - ENS Lyon & Inria - DANTE team

remi.gribonval@inria.fr

preprint: https://arxiv.org/abs/**1905.01208**

Studying the « expressivity » of DNNs

- **DNN = rich architecture to implement functions**
  - $f_\theta : \mathbb{R}^d \to \mathbb{R}^k$ parameterized by θ (weights & biases)

Trained networks

- e.g. goal = regression

$$f_{\hat{\theta}}(x) \approx \mathbb{E}(Z|X = x)$$

- $\hat{\theta}$ typically found using stochastic gradient descent: NOT THIS TALK

Designed networks

- e.g. goal = solve LASSO

$$f_{\hat{\theta}}(x) \approx \arg\min_{\alpha} \tfrac{1}{2}\|x - \mathbf{A}\alpha\|^2 + \lambda\|\alpha\|_1$$

- typically proximal iterations
- learned variant LISTA

Studying the « expressivity » of DNNs

DNN = rich architecture to implement functions

- $f_\theta : \mathbb{R}^d \to \mathbb{R}^k$ parameterized by θ (weights & biases)

Trained networks

- e.g. goal = regression

$$f_{\hat{\theta}}(x) \approx \mathbb{E}(Z|X = x)$$

- $\hat{\theta}$ typically found using stochastic gradient descent: NOT THIS TALK

Designed networks

- e.g. goal = solve LASSO

$$f_{\hat{\theta}}(x) \approx \arg\min_{\alpha} \tfrac{1}{2}\|x - \mathbf{A}\alpha\|^2 + \lambda\|\alpha\|_1$$

- typically proximal iterations
- learned variant LISTA

Best achievable error given a budget ?

- typical budget = #neurons or #connections

Role of "architecture" ?

- activation function(s), *aka* nonlinearity, e.g. ReLU
- depth, skip-connections …

Universal approximation property

A celebrated result

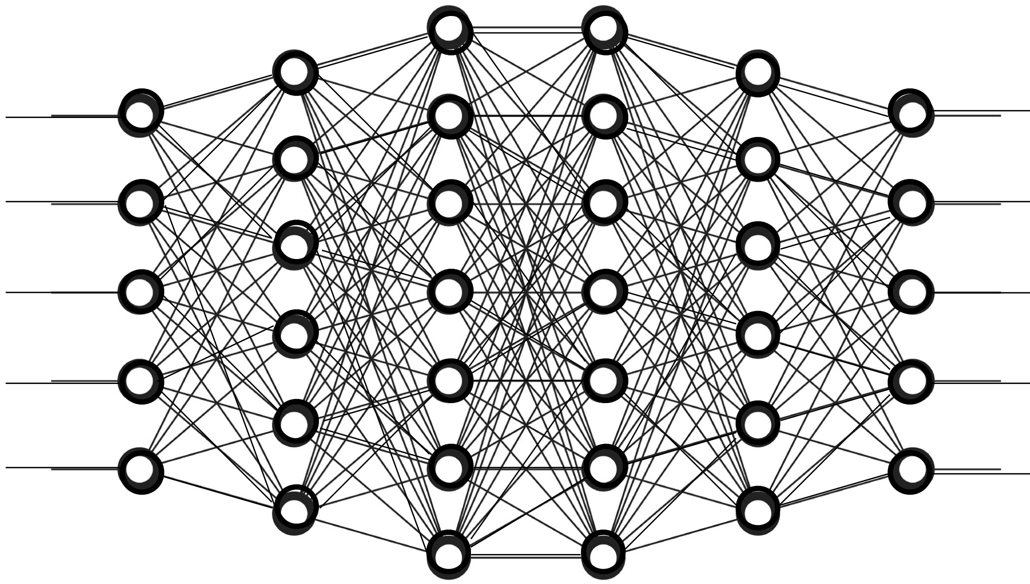
- *One hidden layer* enough to approximate arbitrarily well any continuous function ***on any compact subset*** of $\mathbb{R}^d$, with any “sigmoid-like” activation
  - Hornik, Stinchcombe, White 1989; Cybenko 1989

Tradeoffs / Limitations?

- One hidden layer sufficient … with « enough » neurons
- Approximation *rates* wrt #neurons for “smooth” function
  - Barron, DeVore, Mhaskar, and many more since the 1990s
- Two hidden layers or more needed *on non-compact domains in dimension d>1*

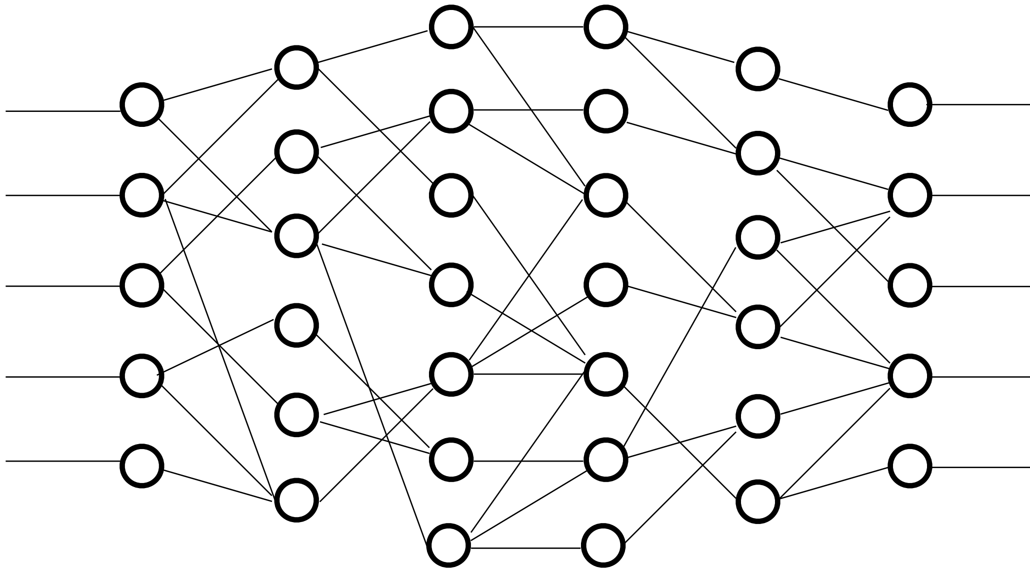
Why sparsely connected networks ?

- **Definition:** sparsity of network
  - parameters θ = weights & biases
  - $\|\theta\|_0$ = # connections <= *n*



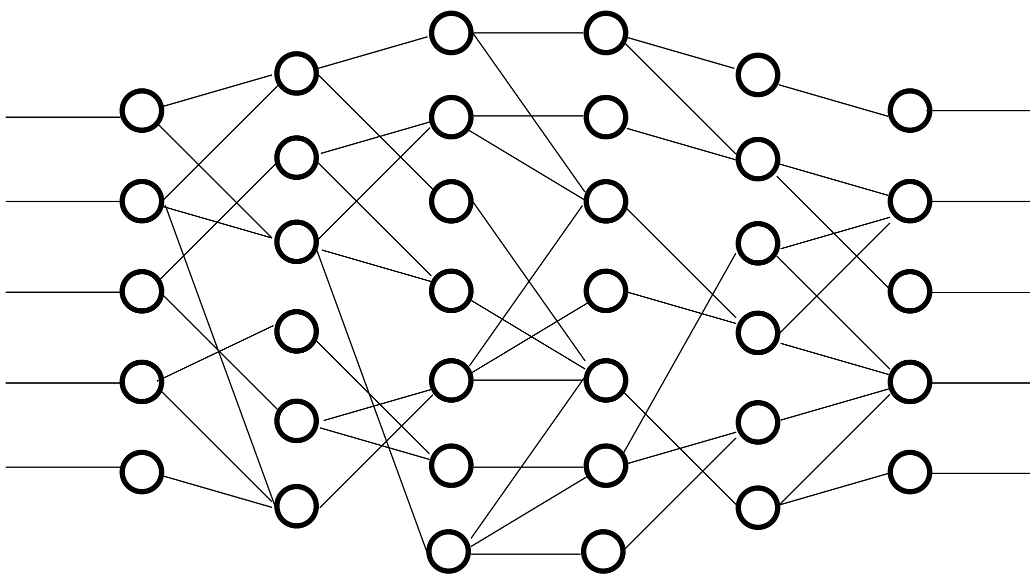
Why sparsely connected networks ?

- **Definition:** sparsity of network
  - parameters θ = weights & biases
  - $\|\theta\|_0$ = # connections <= n



Why sparsely connected networks ?

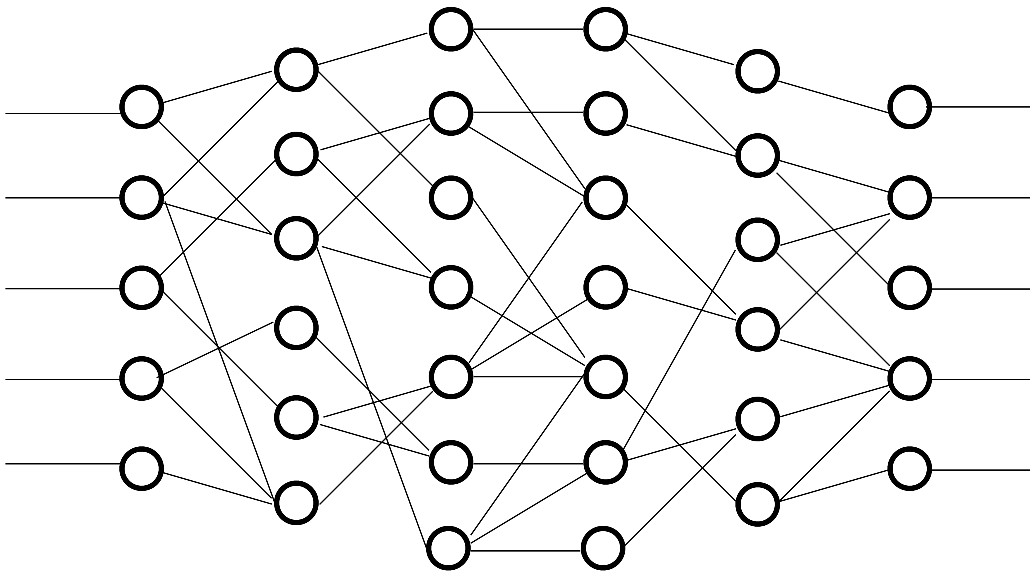
- **Definition**: sparsity of network
  - parameters θ = weights & biases
  - $\|\theta\|_0$ = # connections <= n



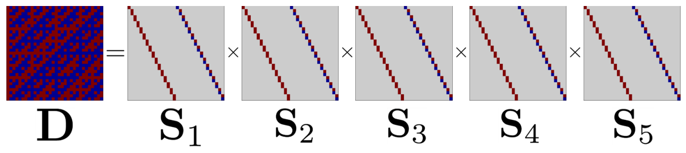
- **Reasonable** proxy to estimate
  - Flops
  - Bits & bytes
  - Sample complexity, e.g. VC dimension
    - see e.g. Bartlett et al 2017

Why sparsely connected networks ?

- **Definition**: sparsity of network
  - parameters θ = weights & biases
  - $\|\theta\|_0$ = # connections <= *n*

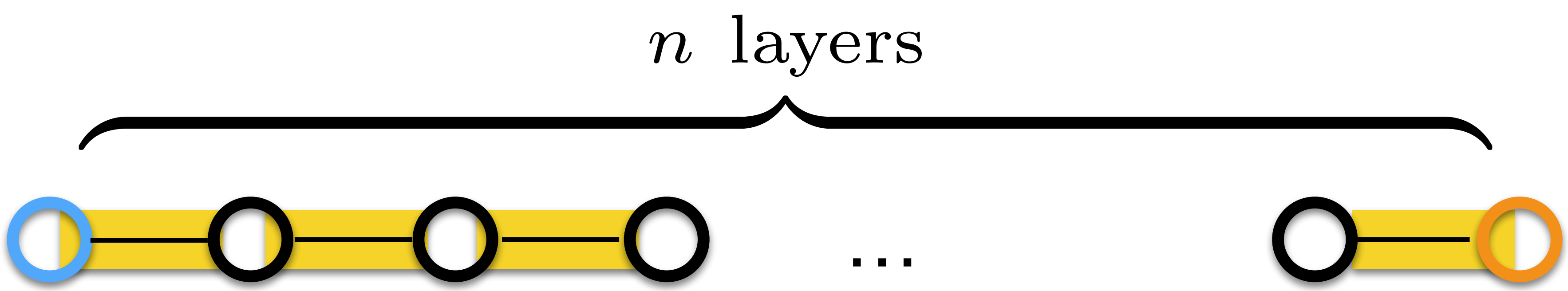


- **Reasonable** proxy to estimate
  - Flops
  - Bits & bytes
  - Sample complexity, e.g. VC dimension
    - see e.g. Bartlett et al 2017
- **Example**: fast linear transforms
  - Activation $\varrho = \mathrm{id}$
  - Butterfly structure for FFT, Hadamard

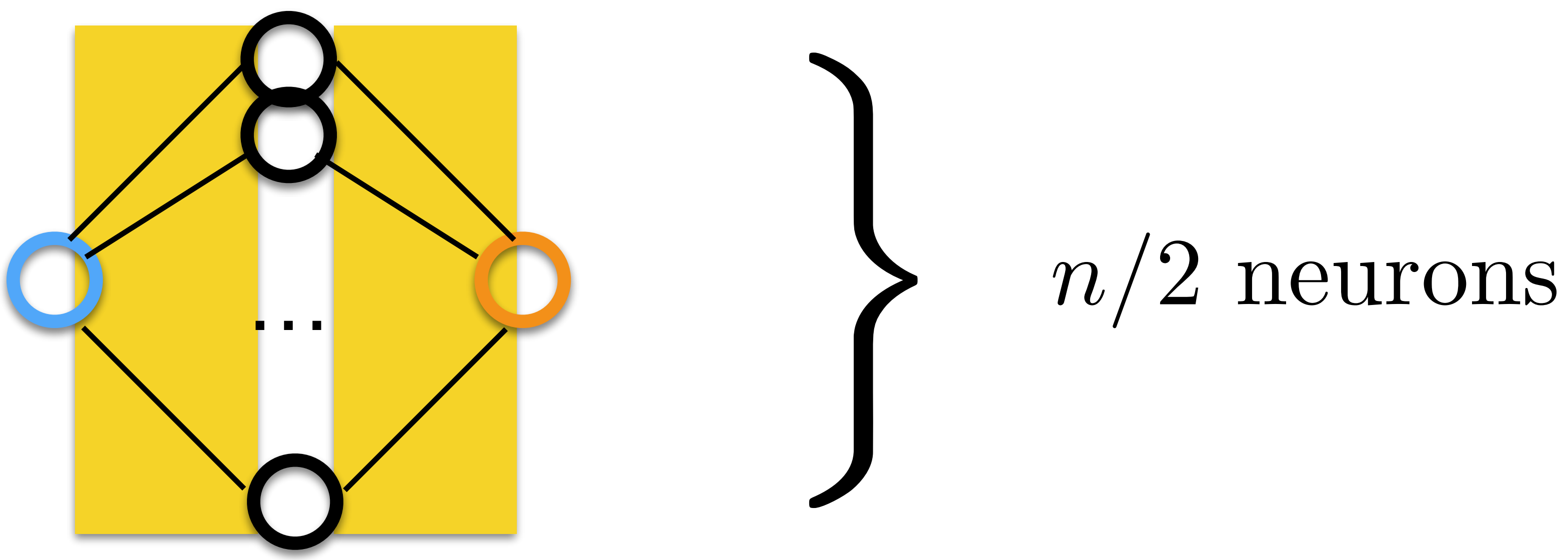


Same sparsity - various network shapes

- Deep & narrow

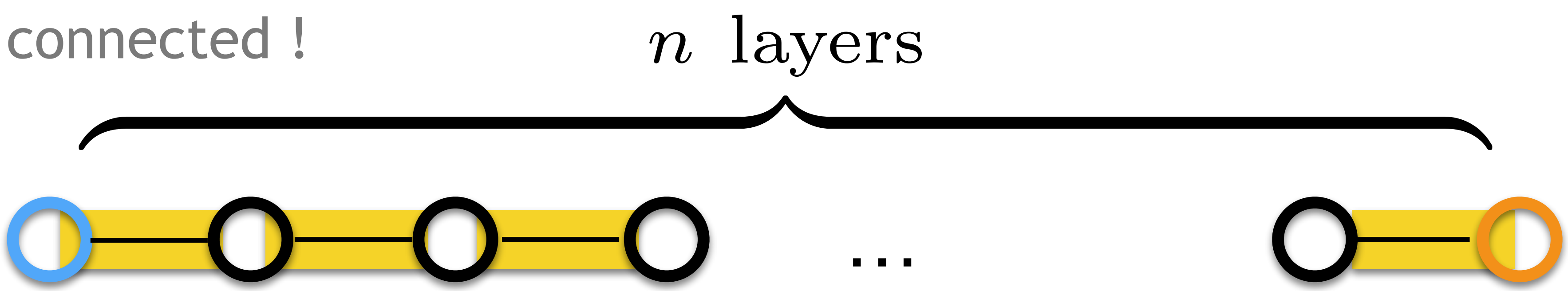


- Shallow & wide

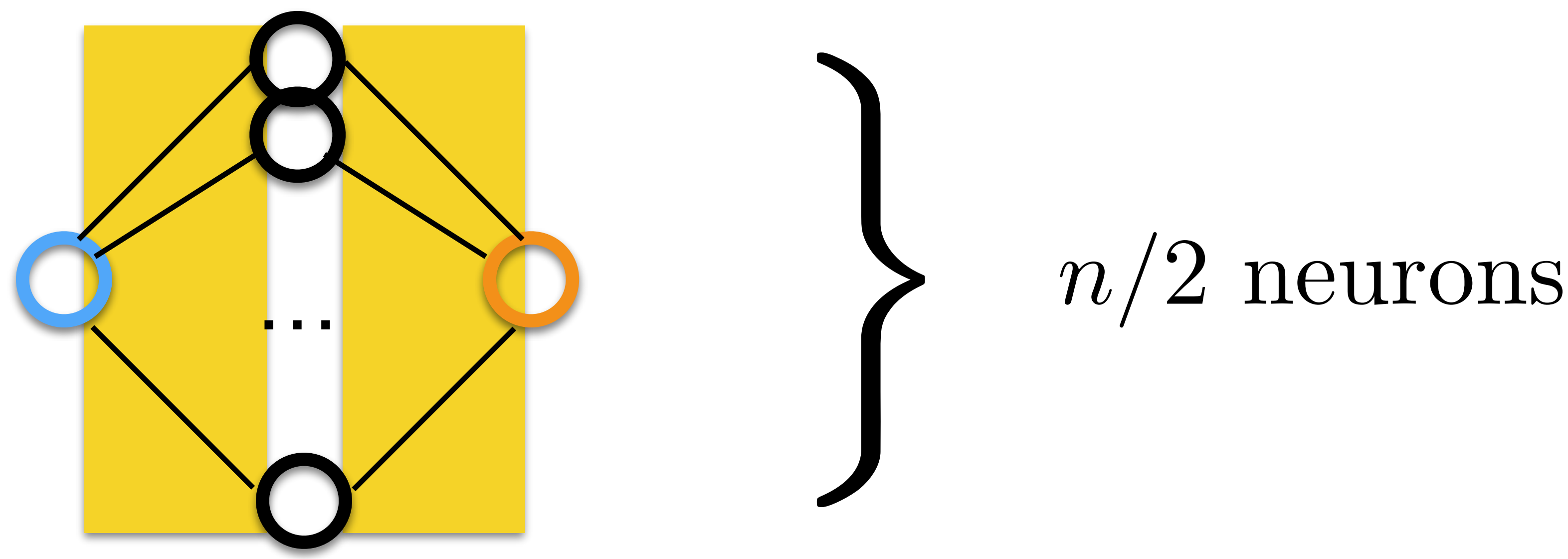


Same sparsity - various network shapes

- Deep & narrow
  - … fully connected !

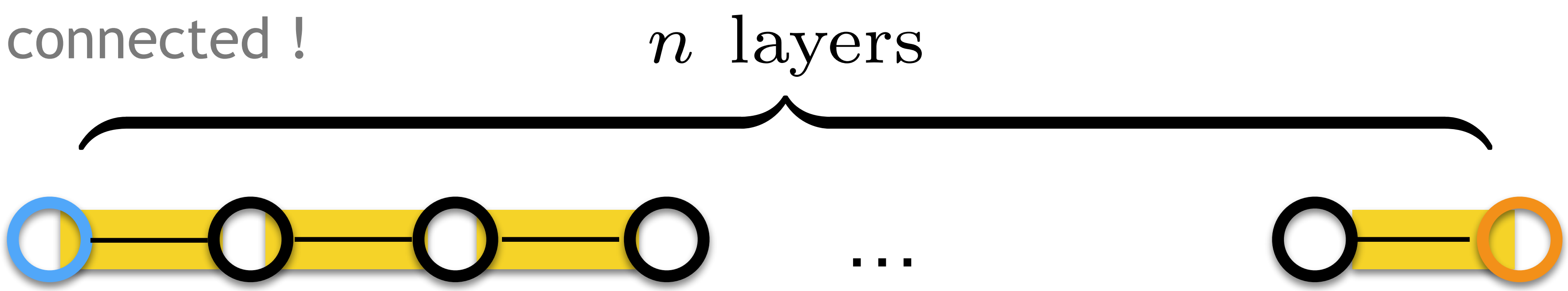


- Shallow & wide
  - … fully connected !



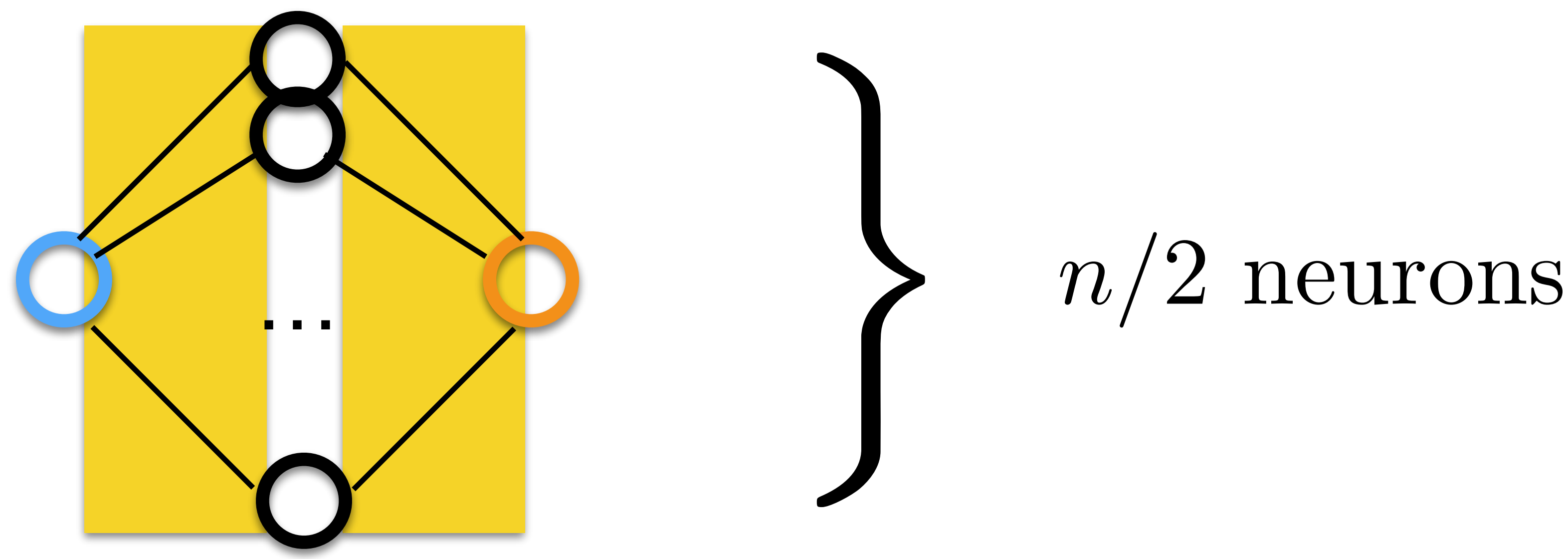
Same sparsity - various network shapes

- Deep & narrow
  - … fully connected !



- … and many more *sparsely* connected possibilities

- Shallow & wide
  - … fully connected !



Approximation with sparse networks

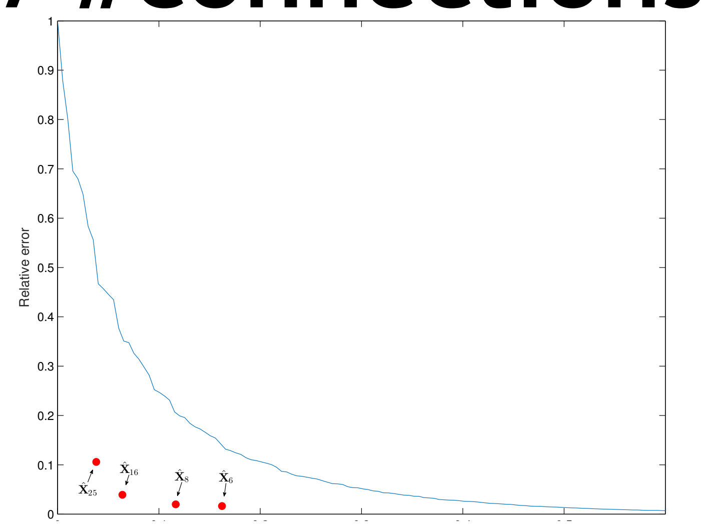
- **Approximation error:** given $f \in L^p(\Omega)$ where $\Omega \subset \mathbb{R}^d$

$$E_n(f) = \inf_{\theta} \|f - f_\theta\|_p$$

  - subject to **sparse connection** constraint $\|\theta\|_0 \leq n$
  - + possibly other constraints (**depth** $L(n)$, choice of activation, …)

- **Tradeoffs error / #connections**

$E_n(f)$



n

example: FAuST (learned fast transforms) vs SVD

Direct *vs* inverse estimate

f is "smooth" (belongs to Sobolev / Besov / modulation space, is "cartoon-like", …)

→ Direct estimates →

$$E_n(f) \lesssim n^{-\alpha}$$

Direct *vs* inverse estimate

f is “smooth” (belongs to Sobolev / Besov / modulation space, is “cartoon-like”, …)

Direct estimates

$$E_n(f) \lesssim n^{-\alpha}$$

- Optimal rate for these function classes:
  - known (nonlinear width)
  - achieved by deep networks :-)
  - same as wavelets, curvelets

  - cf e.g. work of Philip Grohs and co-workers
  - **cf talk by Taiji Suzuki**

Direct *vs* inverse estimate

f is "smooth" (belongs to Sobolev / Besov / modulation space, is "cartoon-like", ...)

Direct estimates →

← Inverse estimates ?

$$E_n(f) \lesssim n^{-\alpha}$$

- Optimal rate for these function classes:
  - known (nonlinear width)
  - achieved by deep networks :-)
  - same as wavelets, curvelets
  - cf e.g. work of Philip Grohs and co-workers
  - **cf talk by Taiji Suzuki**

- What can we say about *f*?
- *Role of activation?*
- *Role of depth?*

Agenda

- Why sparsely connected networks?
- **Approximation spaces**
  - Role of activation function
  - Role of skip connections
- Role of depth

Notion of approximation space

- **Definition: approximation *class***

$$A^\alpha := \{f \in L^p(\Omega) : E_n(f) = O(n^{-\alpha})\}$$

  - *proto-norm*

$$\|f\|_{A^\alpha} := \|f\|_p + \sup_n n^\alpha E_n(f)$$

  - *+variants with finer measures of decay*
- *class may depend on network "architecture"*
  - *presence of « skip-connections »*
  - *choice of activation function(s)*
  - *fixed or varying number of layers L(n) = depth*
- ***larger class = more expressive architecture***

Counting neurons *vs* connections

- Either define approximation error $E_n(f)$ counting
  - #connections $\longrightarrow$ $A^{\alpha}_{\text{weights}}$
  - or #neurons $\longrightarrow$ $A^{\alpha}_{\text{neurons}}$
- **Theorem**: two families are intertwined

$$A^{\alpha}_{\texttt{weights}} \subset A^{\alpha}_{\texttt{neurons}} \subset A^{\alpha/2}_{\texttt{weights}}$$

Role of activation function ϱ

- **(Very) degenerate cases exist**
  - Case of *affine* activation function :
    - A^α = space of all affine transforms
  - Case of *polynomial* activation, with *bounded depth*:
    - A^α = (sub)space of polynomials

Role of activation function ϱ

- **(Very) degenerate cases exist**
  - Case of *affine* activation function :
    - A^{α} = space of all affine transforms
  - Case of *polynomial* activation, with *bounded depth*:
    - A^{α} = (sub)space of polynomials
  - There is a (pathological) *analytic* activation such that with L=3 (two hidden layers) and $n = 3d^2(6d+3)$ connections, for any $f \in L^p([0,1]^d), 0 < p < \infty$

$$E_n(f) = 0$$

    - Maiorov & Pinkus 99

Role of activation function ϱ

- **(Very) degenerate cases exist**
  - Case of *affine* activation function :
    - A^α = space of all affine transforms
  - Case of *polynomial* activation, with *bounded depth*:
    - A^α = (sub)space of polynomials
  - There is a (pathological) *analytic* activation such that with L=3 (two hidden layers) and $n = 3d^2(6d+3)$ connections, for any $f \in L^p([0,1]^d), 0 < p < \infty$

$$E_n(f) = 0$$

    - Maiorov & Pinkus 99
    - in other words, approximation class is trivial

$$A^\alpha = L^p([0,1]^d)$$

The case of *spline* activation functions

Theorem 1

- On bounded domain
  - If ϱ is continuous and *piecewise polynomial* of degree at most *r*, then $A^\alpha(\varrho) \subset A^\alpha(\mathtt{ReLU}^r)$
  - *Equality when activation is a spline (r-1 times continuously differentiable) and not a polynomial*

The case of *spline* activation functions

Theorem 1

- On bounded domain
  - If ϱ is continuous and *piecewise polynomial* of degree at most *r*, then $A^\alpha(\varrho) \subset A^\alpha(\texttt{ReLU}^r)$
  - *Equality when activation is a spline (r-1 times continuously differentiable) and not a polynomial*
  - Moreover, *the expressivity of ReLU powers saturates at r=2*

    if number of layers L(n) growth polynomially, with $A^\alpha(\varrho) := A^\alpha(\varrho, L(\cdot))$

$$A^\alpha(\texttt{ReLU}) \subsetneq A^\alpha(\texttt{ReLU}^2) = A^\alpha(\texttt{ReLU}^r) \subsetneq L^p, \quad \forall r \geq 2$$

The case of *spline* activation functions

Theorem 1

- On bounded domain
  - If ϱ is continuous and *piecewise polynomial* of degree at most *r*, then $A^{\alpha}(\varrho) \subset A^{\alpha}(\mathtt{ReLU}^{r})$
  - *Equality when activation is a spline (r-1 times continuously differentiable) and not a polynomial*
  - Moreover, *the expressivity of ReLU powers saturates at r=2*

    if number of layers L(n) growth polynomially, with $A^{\alpha}(\varrho) := A^{\alpha}(\varrho, L(\cdot))$

$$A^{\alpha}(\mathtt{ReLU}) \subsetneq A^{\alpha}(\mathtt{ReLU}^{2}) = A^{\alpha}(\mathtt{ReLU}^{r}) \subsetneq L^{p}, \quad \forall r \geq 2$$

Under the hood: $$\mathtt{ReLU}^{2^{s}} = \underbrace{\mathtt{ReLU}^{2} \circ \ldots \circ \mathtt{ReLU}^{2}}_{s}$$

Guidelines to choose an activation ?

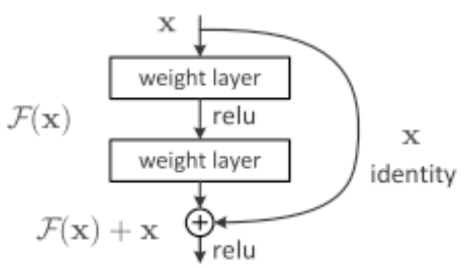
- **Expressive power ?**
  - the *same* (on compact domains) for
    - ReLU
    - Any continuous piecewise affine function
      - absolute value — *cf scattering transforms of Mallat and co-authors*
      - soft-thresholding — *cf Learned Iterative Shrinkage Thresholding, LISTA*
      - leaky-ReLU, C-ReLU, …
  - *potentially larger* for squared ReLU
    - and *the same* as that of any spline of degree at least two
    - potentially harder to train too ? vanishing / exploding gradients
- What about architecture : skip-connections ?

Role of skip-connections

- **Strict networks**
  - ***same*** activation at all neurons

$$\varrho$$

  - limitation: cannot implement
    - skip-connections,
    - ResNets
    - U-nets



- **Generalized networks**
  - ***two*** possible activations at each neuron

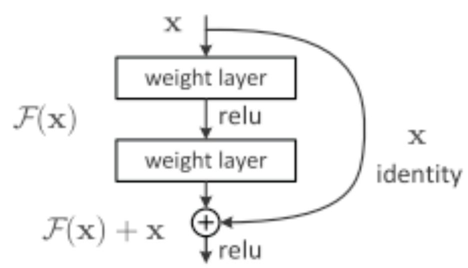
$$\varrho \text{ or } \mathtt{id}$$

Role of skip-connections

- **Strict networks**
  - ***same*** activation at all neurons

    $$\varrho$$

  - limitation: cannot implement
    - skip-connections,
    - ResNets
    - U-nets



- **Generalized networks**
  - ***two*** possible activations at each neuron

    $$\varrho \text{ or } \mathtt{id}$$

- **Theorem 2:** under weak assumptions the class A^α equipped with $\|f\|_{A^\alpha} := \|f\|_p + \sup_n n^\alpha E_n(f)$ is
  - *a complete normed vector space;*
  - *identical for strict & generalized networks*
- assumptions are satisfied by the ReLU and its powers, $\mathtt{ReLU}^r, r \geq 1$
  - *main property: can represent / approximate locally uniformly the identity*

Role of skip-connections

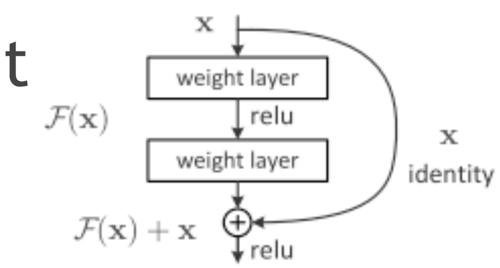
- **Strict networks**
  - ***same*** activation at all neurons

$$\varrho$$

- **Generalized networks**
  - ***two*** possible activations at each neuron

$$\varrho \text{ or } \mathtt{id}$$

- limitation: cannot implement
  - skip-connections,
  - ResNets
  - U-nets



- **Theorem 2:** under weak assumptions the class A^α equipped with $\|f\|_{A^\alpha} := \|f\|_p + \sup_n n^\alpha E_n(f)$ is
  - *a complete normed vector space;*
  - *identical for strict & generalized networks*

→ **Denoted** $A^\alpha(\varrho)$

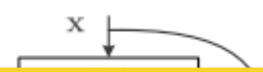
- assumptions are satisfied by the ReLU and its powers, $\mathtt{ReLU}^r, r \geq 1$
  - *main property: can represent / approximate locally uniformly the identity*

Role of skip-connections

- **Strict networks**
  - ***same*** activation at all neurons

    $$\varrho$$

  - limitation: cannot implement
    - skip-c
    - ResN
    - U-net



- **Generalized networks**
  - ***two*** possible activations at each neuron

    ϱ or $\mathtt{id}$

Suggests (TBC) unchanged expressiveness with / without skip-connections

- **Theorem 2:** under weak assumptions the class A^α equipped with $\|f\|_{A^\alpha} := \|f\|_p + \sup_n n^\alpha E_n(f)$ is
  - *a complete normed vector space;*
  - *identical for strict & generalized networks* → **Denoted** $A^\alpha(\varrho)$
- assumptions are satisfied by the ReLU and its powers, $\mathtt{ReLU}^r, r \geq 1$
  - *main property: can represent / approximate locally uniformly the identity*

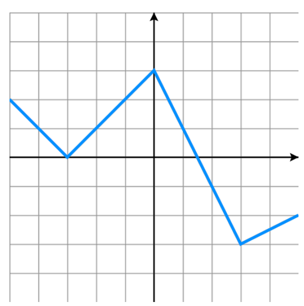
Agenda

- Why sparsely connected networks ?
- Approximation spaces
- **Role of depth**

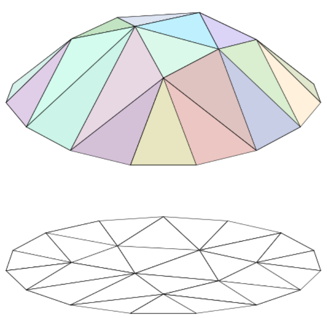
Depth and ReLU networks

Property 1

- any realization of a ReLU-network is continuous and piecewise (affine) linear
  - **d=1**



  - **d>1**



Converse?

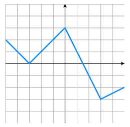
✓ **d=1:** ***any*** piecewise linear function is a realization of a ReLU-network **with one hidden layer**

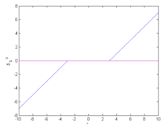
★ **d>1:** no longer true

- One hidden layer: realization not compactly supported, ***not even integrable*** (unless it is zero)
- Need **at least two hidden layers** to be integrable

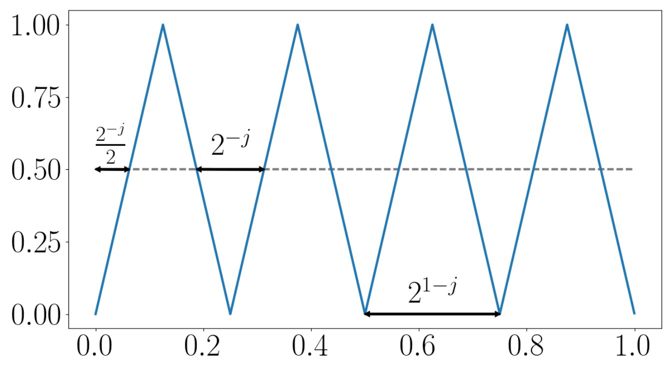
Benefits of depth ?

- **ReLU-networks in dimension d=1**
  - Can implement *any* piecewise affine function
    - **For L=2** (one hidden layer), #breakpoints = #neurons





    - **For large L** (deep network) #breakpoints can be *exponential* in #neurons
    - **Typical example** = sawtooth function
      - see e.g. Mhaskar & Poggio 2016, Telgarsky 2016

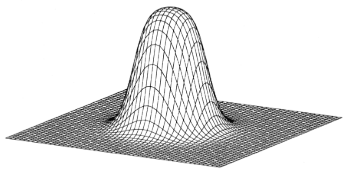


- composition of *j* hat functions
- **implemented** by (deep) network of depth *j* with *O(j)* neurons / connections
- ***badly approximated*** by shallow network *(needs exponentially many neurons)*

“Shallow” ReLU-nets have limited expressivity

Theorem 3:

- Consider a nonzero $C_c^3(\mathbb{R}^d)$ function *f*



  - with networks of depth bounded by L we have $E_n(f) \geq C(f) n^{-2L}, \ \forall n$

- In other words: for $\alpha > 2L$ we have $C_c^3(\mathbb{R}^d) \cap A^\alpha(\mathtt{ReLU}, L) = \{0\}$

- Cf Theorem 4.5 in: Petersen and F. Voigtlaender. Optimal approximation of piecewise smooth functions using deep ReLU neural networks. arXiv preprint arXiv:1709.05289, 2017.

Corollary:

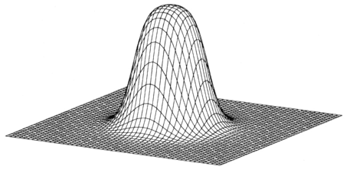
- Consider a function family B such that $C_c^3(\mathbb{R}^d) \cap B \neq \{0\}$

  examples: any classical Sobolev or Besov space, of *arbitrary* positive smoothness;
  the set of « cartoon-like » images

  if $B \subset A^\alpha(\mathtt{ReLU}, L)$ then $L > \alpha/2$

"Shallow" ReLU-nets have limited expressivity

- **Theorem 3:**
  - Consider a nonzero $C_c^3(\mathbb{R}^d)$ function *f*



    - with networks of depth bounded by L we have $E_n(f) \geq C(f) n^{-2L}, \ \forall n$
  - In other words: for $\alpha > 2L$ we have $C_c^3(\mathbb{R}^d) \cap A^\alpha(\mathtt{ReLU}, L) = \{0\}$
  - Cf Theorem 4.5 in: Petersen and F. Voigtlaender. Optimal approximation of piecewise smooth functions using deep ReLU neural networks. arXiv preprint arXiv:1709.05289, 2017.
- **Corollary:**
  - Consider a function family B such that $C_c^3(\mathbb{R}^d) \cap B \neq \{0\}$
    examples: any classical Sobolev or Besov space, of *arbitrary* positive smoothness;
    the set of « cartoon-like » images

    if $B \subset A^\alpha(\mathtt{ReLU}, L)$ then $L > \alpha/2$

With ReLU: "expressivity requires depth"

Role of depth

Theorem 4

sparsely connected networks of bounded depth L

- Direct estimate for Besov spaces

$$B^{\alpha d} \subset A^{\alpha}(\mathtt{ReLU}^r, L)$$

  - for a certain range of rates α

- Inverse estimate for Besov spaces

$$A^{\alpha}(\mathtt{ReLU}^r, L) \subset B^{\alpha/\lfloor L/2 \rfloor}$$

  - proved for d=1
  - best possible Besov exponent, for any *d*

Role of depth

Theorem 4

sparsely connected networks of bounded depth L

- Direct estimate for Besov spaces

$$B^{\alpha d} \subset A^{\alpha}(\mathtt{ReLU}^r, L)$$

  - for a certain range of rates α

- Inverse estimate for Besov spaces

$$A^{\alpha}(\mathtt{ReLU}^r, L) \subset B^{\alpha/\lfloor L/2 \rfloor}$$

  - proved for d=1
  - best possible Besov exponent, for any *d*

Proof sketch

- Direct result
  - Characterize Besov with wavelets
  - Implement n-term wavelet expansion with *O(n)*-sparsely connected network of depth L=3

Role of depth

Theorem 4

sparsely connected networks of bounded depth L

- Direct estimate for Besov spaces

$$B^{\alpha d} \subset A^{\alpha}(\texttt{ReLU}^r, L)$$

  - for a certain range of rates α

- Inverse estimate for Besov spaces

$$A^{\alpha}(\texttt{ReLU}^r, L) \subset B^{\alpha/\lfloor L/2 \rfloor}$$

  - proved for d=1
  - best possible Besov exponent, for any *d*

Proof sketch

- Direct result
  - Characterize Besov with wavelets
  - Implement n-term wavelet expansion with *O(n)*-sparsely connected network of depth L=3

- Inverse result
  - **Lemma:** if $\|\theta\|_0 \leq n$ then f_θ is piecewise poly with $O(n^{\lfloor L/2 \rfloor})$ pieces
  - Apply Petrushev's inverse estimate for free-knot splines

Role of depth

Theorem 4

sparsely connected networks of bounded depth L

- Direct estimate for Besov spaces

$$B^{\alpha d} \subset A^{\alpha}(\mathtt{ReLU}^r, L)$$

  - for a certain range of rates α

- Inverse estimate for Besov spaces

$$A^{\alpha}(\mathtt{ReLU}^r, L) \subset B^{\alpha/\lfloor L/2 \rfloor}$$

  - proved for d=1
  - best possible Besov exponent, for any *d*

Proof sketch

- Direct result
  - Characterize Besov with wavelets
  - Implement n-term wavelet expansion with *O(n)*-sparsely connected network of depth L=3

- Inverse result
  - **Lemma:** if $\|\theta\|_0 \leq n$ then f_θ is piecewise poly with $O(n^{\lfloor L/2 \rfloor})$ pieces
  - Apply Petrushev's inverse estimate for free-knot splines

deeper DNN → expresses *rougher* functions

Role of depth

Theorem 4

sparsely connected networks of bounded depth L

- Direct estimate for Besov spaces

$$B^{\alpha d} \subset A^{\alpha}(\mathtt{ReLU}^r, L)$$

  - for a certain range of rates α

- Inverse estimate for Besov spaces

$$A^{\alpha}(\mathtt{ReLU}^r, L) \subset B^{\alpha/\lfloor L/2 \rfloor}$$

  - proved for d=1
  - best possible Besov exponent, for any *d*

Proof sketch

- Direct result
  - Characterize Besov with wavelets
  - Implement n-term wavelet expansion with *O(n)*-sparsely connected network of depth L=3

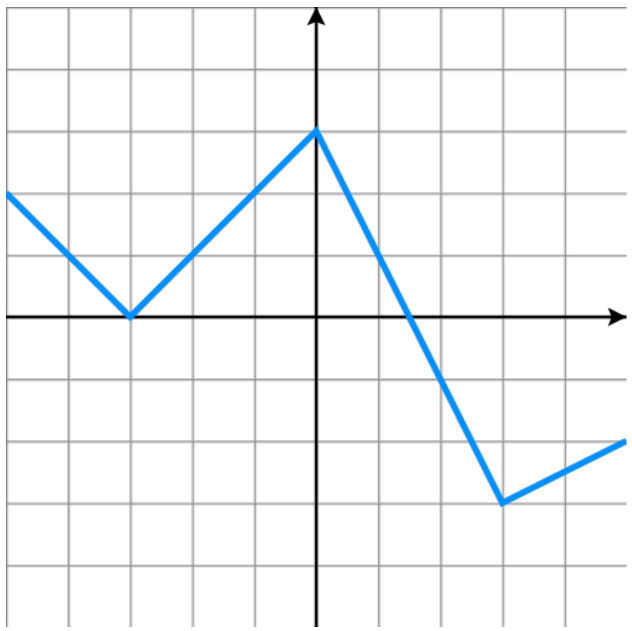
- Inverse result
  - **Lemma:** if $\|\theta\|_0 \leq n$ then f_θ is piecewise poly with $O(n^{\lfloor L/2 \rfloor})$ pieces

    role of **pairs** of layers ?
  - Apply Petrushev's inverse estimate for free-knot splines

deeper DNN → expresses *rougher* functions

Under the hood

- Many of these results rely on … counting pieces !
  - For ReLU networks of depth L in dimension d=1
    - if #neurons = n then $\sharp\texttt{pieces} = \mathcal{O}(n^{L-1})$
    - if #connections = n then $\sharp\texttt{pieces} = \mathcal{O}(n^{\lfloor L/2 \rfloor})$



Set theoretic picture

$L^p(\Omega)$

Set theoretic picture

- Approximation rate α with *n*-term wavelet expansions
  - constructive (wavelet thresholding)

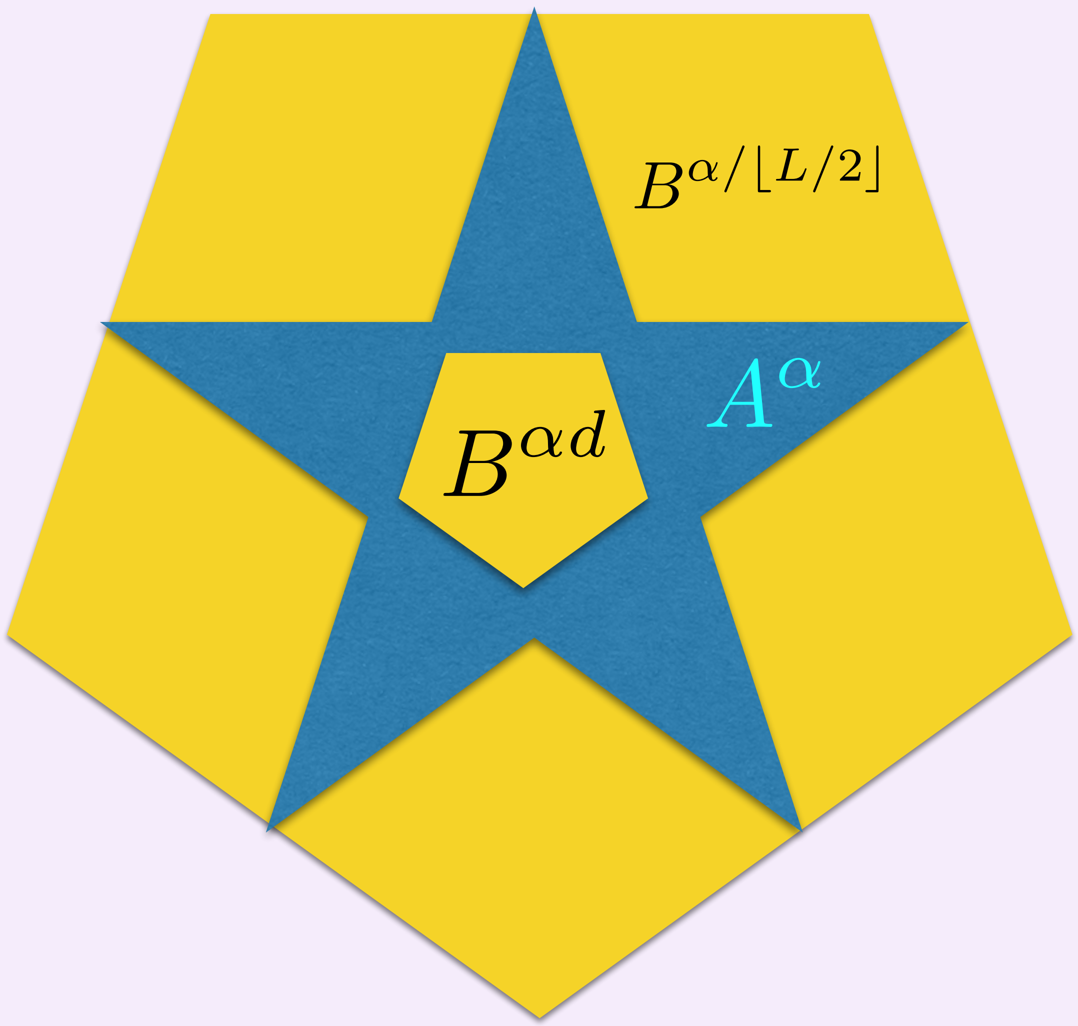
Set theoretic picture

- Approximation rate α with *n*-term wavelet expansions
  - constructive (wavelet thresholding)
- Same rate, ReLU-networks with *n* connections
  - non-constructive
  - more expressive

$L^p(\Omega)$

Set theoretic picture

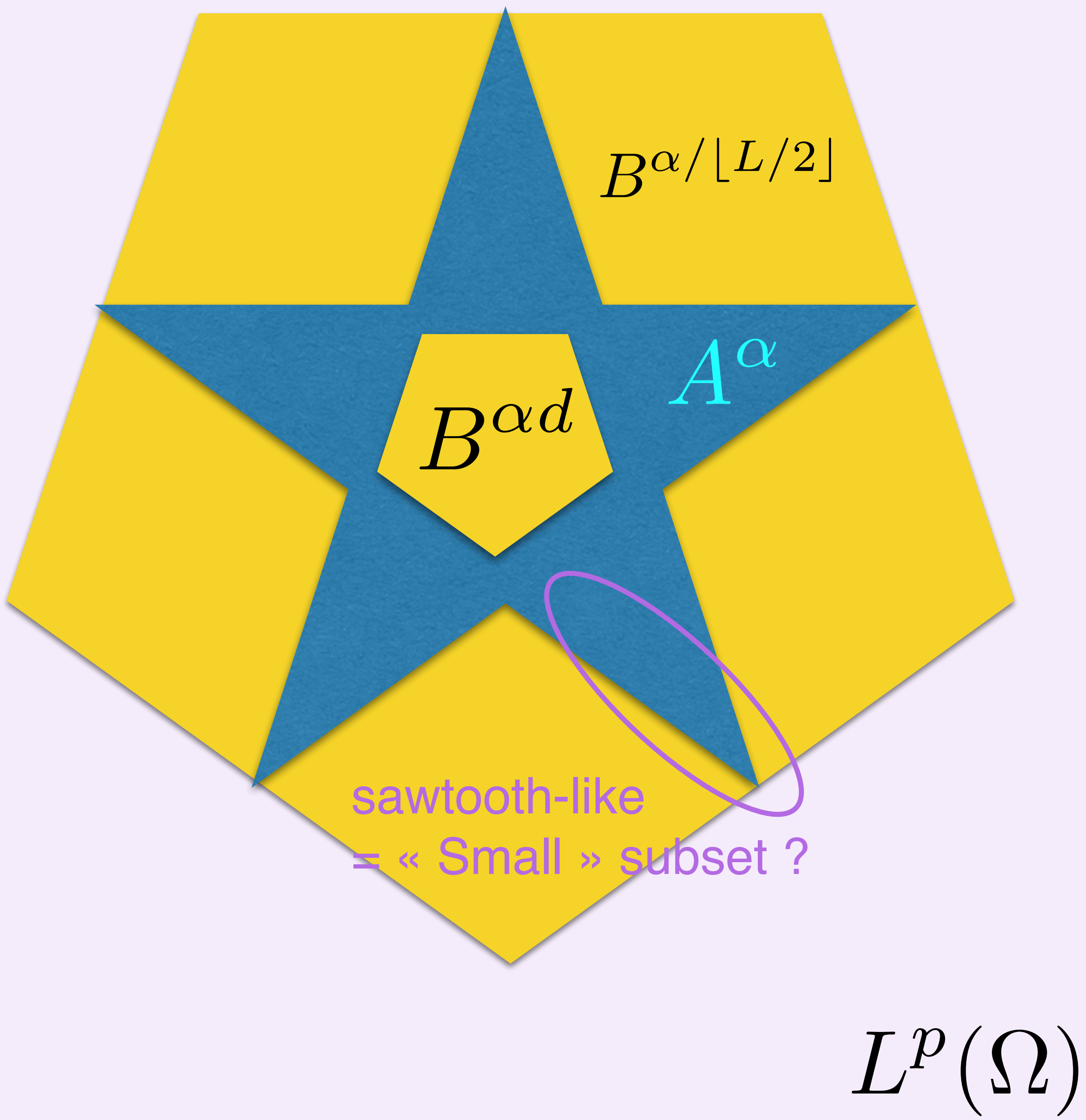
- Approximation rate α with *n*-term wavelet expansions
  - constructive (wavelet thresholding)
- Same rate, ReLU-networks with *n* connections
  - non-constructive
  - more expressive



$L^p(\Omega)$

Set theoretic picture

- Approximation rate α with *n*-term wavelet expansions
  - constructive (wavelet thresholding)
- Same rate, ReLU-networks with *n* connections
  - non-constructive
  - more expressive
- For some functions in A^α, *n*-term wavelet expansions only reach the rate $\frac{\alpha}{d\lfloor L/2\rfloor}$
  - $n' = \mathcal{O}(n^{d\lfloor L/2\rfloor})$ wavelets are required to reach the rate α for such functions



Summary & perspectives

Summary: Approximation with DNNs

Role of architecture

- **Strict vs generalized networks:** same expressiveness
- **Challenge:** expressiveness of **plain vs skip connections / ResNets?**

➡ *main / only difference =* ***ease of training*** *with stochastic gradient ?*

Summary: Approximation with DNNs

Role of architecture

- **Strict vs generalized networks:** same expressiveness
- **Challenge:** expressiveness of **plain vs skip connections / ResNets?**

➡ *main / only difference = **ease of training** with stochastic gradient ?*

Role of nonlinearity

- $\mathtt{ReLU}(t) = \max(t, 0) = t_+$ as expressive as any piecewise affine activation
- $\mathtt{ReLU}^2$ as expressive as any continuous piecewise polynomial activation
- Expressiveness of $\mathtt{ReLU}^r$ "saturates" at r=2

➡ **Challenge:** training of $\mathtt{ReLU}^2$-networks ? vanishing gradients ?

Summary: Approximation with DNNs

Role of architecture

- **Strict vs generalized networks:** same expressiveness
- **Challenge:** expressiveness of **plain vs skip connections / ResNets?**

➡ *main / only difference = **ease of training** with stochastic gradient ?*

Role of nonlinearity

- $\mathrm{ReLU}(t) = \max(t, 0) = t_+$ as expressive as any piecewise affine activation
- ReLU^2 as expressive as any continuous piecewise polynomial activation
- Expressiveness of ReLU^r “saturates” at r=2

➡ **Challenge:** training of ReLU^2-networks ? vanishing gradients ?

Role of depth

- Deep enough, any dimension: DNN **strictly more expressive than wavelets**

Overall summary & perspectives

- **First step: expressivity of different architectures**
  - *… spaces yet to be better* ***characterized***
  - ***convolutional architectures****,* ***ResNets****,* ***U-nets****,* ***max-pooling*** *?*

preprint: Approximation spaces of deep neural networks
https://arxiv.org/abs/1905.01208

see also Nonlinear Approximation and (Deep) ReLU Networks
[Daubechies, DeVore, Foucart, Hanin, Petrova, 2019]

- **Next steps ?**
  - *… constructive approximation/training* ***algorithms*** *?*
    - ***surely NP-hard***
    - ***assumptions needed for bounded complexity & provable performance***
  - *…* ***guidelines*** *for choosing a DNN architecture ?*
  - *…* ***statistical guarantees*** *?*

see e.g. Nonparametric regression using deep neural networks with ReLU activation function [J. Schmidt-Hieber, 2017]