On the Representation, Interpolation, and Approximation Power of ReLU Neural Networks

Jonathan W. Siegel

Texas A&M University

jwsiegel@tamu.edu

CMBS Conference at Morgan State University
June 22, 2023

Outline

Deep Neural Networks for Scientific Computing

- Recently, deep neural networks have been widely applied to scientific computing:
  - Solving PDEs[1]
  - Learning operators from data[2]
  - Inverse Problem/Inverse Design[3]
  - etc.

---

[1] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378 (2019), pp. 686–707, Jiequn Han, Arnulf Jentzen, and Weinan E. "Solving high-dimensional partial differential equations using deep learning". In: *Proceedings of the National Academy of Sciences* 115.34 (2018), pp. 8505–8510.

[2] Lu Lu et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature Machine Intelligence* 3.3 (2021), pp. 218–229, Zongyi Li et al. "Fourier Neural Operator for Parametric Partial Differential Equations". In: *International Conference on Learning Representations*. 2020.

[3] Lu Lu et al. "Physics-informed neural networks with hard constraints for inverse design". In: *SIAM Journal on Scientific Computing* 43.6 (2021), B1105–B1132.

Deep Neural Networks for Scientific Computing

- Recently, deep neural networks have been widely applied to scientific computing:
  - Solving PDEs[1]
  - Learning operators from data[2]
  - Inverse Problem/Inverse Design[3]
  - etc.
- How good is approximation with deep neural networks?

[1] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378 (2019), pp. 686–707, Jiequn Han, Arnulf Jentzen, and Weinan E. "Solving high-dimensional partial differential equations using deep learning". In: *Proceedings of the National Academy of Sciences* 115.34 (2018), pp. 8505–8510.

[2] Lu Lu et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature Machine Intelligence* 3.3 (2021), pp. 218–229, Zongyi Li et al. "Fourier Neural Operator for Parametric Partial Differential Equations". In: *International Conference on Learning Representations*. 2020.

[3] Lu Lu et al. "Physics-informed neural networks with hard constraints for inverse design". In: *SIAM Journal on Scientific Computing* 43.6 (2021), B1105–B1132.

Deep ReLU Networks

- Consider an affine map $A_{\mathbf{W},b} : \mathbb{R}^n \to \mathbb{R}^k$

$$A_{\mathbf{W},b}(x) = \mathbf{W}x + b. \tag{1}$$

Deep ReLU Networks

- Consider an affine map $A_{\mathbf{W},b} : \mathbb{R}^n \to \mathbb{R}^k$

$$A_{\mathbf{W},b}(x) = \mathbf{W}x + b. \tag{1}$$

- Let $\sigma(x) = \max(0, x)$ denote the ReLU
  - When applied to a vector, σ is applied component-wise

Deep ReLU Networks

- Consider an affine map $A_{\mathbf{W},b} : \mathbb{R}^n \rightarrow \mathbb{R}^k$

$$A_{\mathbf{W},b}(x) = \mathbf{W}x + b. \tag{1}$$

- Let $\sigma(x) = \max(0, x)$ denote the ReLU
  - When applied to a vector, σ is applied component-wise
- A deep ReLU network with width W and depth L mapping $\mathbb{R}^d$ to $\mathbb{R}^k$ is a composition

$$A_{\mathbf{W}_L,b_L} \circ \sigma \circ A_{\mathbf{W}_{L-1},b_{L-1}} \circ \sigma \circ \cdots \circ \sigma \circ A_{\mathbf{W}_1,b_1} \circ \sigma \circ A_{\mathbf{W}_0,b_0} \tag{2}$$

  - Here $A_{\mathbf{W}_1,b_1}, ..., A_{\mathbf{W}_{L-1},b_{L-1}} : \mathbb{R}^W \rightarrow \mathbb{R}^W$

Deep ReLU Networks

- Consider an affine map $A_{\mathbf{W},b} : \mathbb{R}^n \to \mathbb{R}^k$

$$A_{\mathbf{W},b}(x) = \mathbf{W}x + b. \tag{1}$$

- Let $\sigma(x) = \max(0, x)$ denote the ReLU
  - When applied to a vector, σ is applied component-wise
- A deep ReLU network with width W and depth L mapping $\mathbb{R}^d$ to $\mathbb{R}^k$ is a composition

$$A_{\mathbf{W}_L,b_L} \circ \sigma \circ A_{\mathbf{W}_{L-1},b_{L-1}} \circ \sigma \circ \cdots \circ \sigma \circ A_{\mathbf{W}_1,b_1} \circ \sigma \circ A_{\mathbf{W}_0,b_0} \tag{2}$$

  - Here $A_{\mathbf{W}_1,b_1}, \ldots, A_{\mathbf{W}_{L-1},b_{L-1}} : \mathbb{R}^W \to \mathbb{R}^W$
  - We denote the set of these by $\Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$.

What types of functions are in $\Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$

- All functions $f \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$ are continuous and piecewise linear

[4] Juncai He et al. "ReLU Deep Neural Networks and Linear Finite Elements". In: *Journal of Computational Mathematics* 38.3 (2020), pp. 502–527.

What types of functions are in $\Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$

- All functions $f \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$ are continuous and piecewise linear
- The number of pieces can be exponential in the depth L
  - Number of parameters scales like $W^2 L$

[4] Juncai He et al. "ReLU Deep Neural Networks and Linear Finite Elements". In: *Journal of Computational Mathematics* 38.3 (2020), pp. 502–527.

What types of functions are in $\Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$

- All functions $f \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$ are continuous and piecewise linear
- The number of pieces can be exponential in the depth L
  - Number of parameters scales like $W^2 L$
- Classical piecewise linear finite element functions can be represented[4]

[4] Juncai He et al. "ReLU Deep Neural Networks and Linear Finite Elements". In: *Journal of Computational Mathematics* 38.3 (2020), pp. 502–527.

What types of functions are in $\Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$

- All functions $f \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R}^k)$ are continuous and piecewise linear
- The number of pieces can be exponential in the depth L
  - Number of parameters scales like W^2L
- Classical piecewise linear finite element functions can be represented[4]
- All piecewise linear continuous functions can be represented if $L \geq \log(d+1)$
  - Open problem: Can you use fewer layers?

[4] Juncai He et al. "ReLU Deep Neural Networks and Linear Finite Elements". In: *Journal of Computational Mathematics* 38.3 (2020), pp. 502–527.

Sobolev Spaces

- We consider the Sobolev spaces $W^s(L_q(\Omega))$, defined by

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \|f^{(s)}\|_{L^q(\Omega)} \tag{3}$$

[5] Lawrence C Evans. *Partial differential equations.* Vol. 19. American Mathematical Soc., 2010.

Sobolev Spaces

- We consider the Sobolev spaces $W^s(L_q(\Omega))$, defined by

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \|f^{(s)}\|_{L^q(\Omega)} \tag{3}$$

  - If s is not a integer: Write $s = k + \theta$, $\theta \in [0, 1)$

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \sum_{|\alpha|=k} \int_{\Omega\times\Omega} \frac{|f^{(\alpha)}(x) - f^{(\alpha)}(y)|^q}{|x-y|^{d+\theta q}} dxdy \tag{4}$$

  - For simplicity, take $\Omega = [0, 1]^d$ (but any bounded domain will work)

[5] Lawrence C Evans. *Partial differential equations.* Vol. 19. American Mathematical Soc., 2010.

Sobolev Spaces

- We consider the Sobolev spaces $W^s(L_q(\Omega))$, defined by

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \|f^{(s)}\|_{L^q(\Omega)} \tag{3}$$

  - If s is not a integer: Write $s = k + \theta$, $\theta \in [0, 1)$

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \sum_{|\alpha|=k} \int_{\Omega\times\Omega} \frac{|f^{(\alpha)}(x) - f^{(\alpha)}(y)|^q}{|x-y|^{d+\theta q}} dxdy \tag{4}$$

  - For simplicity, take $\Omega = [0, 1]^d$ (but any bounded domain will work)
- Typical space for PDE regularity estimates[5]

[5] Lawrence C Evans. *Partial differential equations.* Vol. 19. American Mathematical Soc., 2010.

Sobolev Spaces

- We consider the Sobolev spaces $W^s(L_q(\Omega))$, defined by

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \|f^{(s)}\|_{L^q(\Omega)} \tag{3}$$

  - If s is not a integer: Write $s = k + \theta$, $\theta \in [0, 1)$

$$\|f\|_{W^s(L_q(\Omega))} = \|f\|_{L_q(\Omega)} + \sum_{|\alpha|=k} \int_{\Omega\times\Omega} \frac{|f^{(\alpha)}(x) - f^{(\alpha)}(y)|^q}{|x-y|^{d+\theta q}} dxdy \tag{4}$$

  - For simplicity, take $\Omega = [0, 1]^d$ (but any bounded domain will work)
- Typical space for PDE regularity estimates[5]
- Our results also apply to Besov spaces

[5] Lawrence C Evans. *Partial differential equations.* Vol. 19. American Mathematical Soc., 2010.

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

- Measure error in the $L_p(\Omega)$ norm

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

- Measure error in the $L_p(\Omega)$ norm
- What are the optimal rates of approximation by deep ReLU networks:

$$\sup_{f \in F_q^s(\Omega)} \inf_{f_L \in \Upsilon^{W,L}} \|f - f_L\|_{L_p(\Omega)}? \tag{6}$$

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

- Measure error in the $L_p(\Omega)$ norm
- What are the optimal rates of approximation by deep ReLU networks:

$$\sup_{f \in F_q^s(\Omega)} \inf_{f_L \in \Upsilon^{W,L}} \|f - f_L\|_{L_p(\Omega)}? \tag{6}$$

- Interested in the asymptotics as $L \to \infty$ with W fixed (large enough)

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

- Measure error in the $L_p(\Omega)$ norm
- What are the optimal rates of approximation by deep ReLU networks:

$$\sup_{f \in F_q^s(\Omega)} \inf_{f_L \in \Upsilon^{W,L}} \|f - f_L\|_{L_p(\Omega)}? \tag{6}$$

- Interested in the asymptotics as $L \to \infty$ with W fixed (large enough)
  - In this regime we get best rates in terms of number of parameters

How Efficient Are Deep ReLU Networks?

- Let $\Omega \subset \mathbb{R}^d$ be a bounded domain
- Consider function classes determined by Sobolev spaces

$$F_q^s(\Omega) = \{\|f\|_{W^s(L_q(\Omega))} \leq 1\} \tag{5}$$

- Measure error in the $L_p(\Omega)$ norm
- What are the optimal rates of approximation by deep ReLU networks:

$$\sup_{f \in F_q^s(\Omega)} \inf_{f_L \in \Upsilon^{W,L}} \|f - f_L\|_{L_p(\Omega)}? \tag{6}$$

- Interested in the asymptotics as $L \to \infty$ with W fixed (large enough)
  - In this regime we get best rates in terms of number of parameters
  - Also considering width and depth varying together (joint with Juncai He)

Sobolev Embedding

- Goal: Approximate $f \in W^s(L_q(\Omega))$ in the L_p-norm

Sobolev Embedding

- Goal: Approximate $f \in W^s(L_q(\Omega))$ in the L_p-norm
- In order for this to be possible, we need

$$W^s(L_q) \subset L_p.$$

Sobolev Embedding

- Goal: Approximate $f \in W^s(L_q(\Omega))$ in the L_p-norm
- In order for this to be possible, we need

$$W^s(L_q) \subset L_p.$$

- If the Sobolev condition strictly fails, i.e. if $\frac{1}{q} - \frac{1}{p} > \frac{s}{d}$, then $W^s(L_q) \not\subset L_p$
- If the Sobolev condition is strictly satisfed, i.e. $\frac{1}{q} - \frac{1}{p} < \frac{s}{d}$, then we have a compact embedding

$$W^s(L_q) \subset\subset L_p. \tag{7}$$

  - This is the case we will be most interested in

Sobolev Embedding

- Goal: Approximate $f \in W^s(L_q(\Omega))$ in the L_p-norm
- In order for this to be possible, we need

$$W^s(L_q) \subset L_p.$$

- If the Sobolev condition strictly fails, i.e. if $\frac{1}{q} - \frac{1}{p} > \frac{s}{d}$, then $W^s(L_q) \not\subset L_p$
- If the Sobolev condition is strictly satisfed, i.e. $\frac{1}{q} - \frac{1}{p} < \frac{s}{d}$, then we have a compact embedding

$$W^s(L_q) \subset\subset L_p. \tag{7}$$

  - This is the case we will be most interested in
- In the boundary case where $\frac{1}{q} - \frac{1}{p} = \frac{s}{d}$ we may or may not have embedding

Classical Approximation Methods

- Linear methods of approximation[6]:

$$\inf_{\substack{P_N \\ \text{rank N}}} \sup_{f\in F^s_q(\Omega)} \|f - P_N(f)\|_{L_p(\Omega)} \eqsim \begin{cases} N^{-s/d} & p \leq q \\ N^{-s/d+1/q-1/p} & p > q. \end{cases} \tag{8}$$

[6] George G Lorentz, Manfred v Golitschek, and Yuly Makovoz. *Constructive approximation: advanced problems*. Vol. 304. Springer, 1996.

Classical Approximation Methods

- Linear methods of approximation[6]:

$$\inf_{\substack{P_N \\ \text{rank N}}} \sup_{f\in F_q^s(\Omega)} \|f - P_N(f)\|_{L_p(\Omega)} \eqsim \begin{cases} N^{-s/d} & p \leq q \\ N^{-s/d+1/q-1/p} & p > q. \end{cases} \tag{8}$$

- Need non-linear (i.e. adaptive) methods when $p > q$ to recover rate $O(N^{-s/d})$
  - with a compact Sobolev embedding
  - e.g. n-term wavelets, adaptive piecewise polynomial, variable knot splines

[6] George G Lorentz, Manfred v Golitschek, and Yuly Makovoz. *Constructive approximation: advanced problems*. Vol. 304. Springer, 1996.

First Approach to Deep Network Approximation

- Yarotsky[7] showed that polynomials can be efficiently approximated with deep ReLU networks

[7] Dmitry Yarotsky. "Error bounds for approximations with deep ReLU networks". In: *Neural Networks* 94 (2017), pp. 103–114.

[8] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

First Approach to Deep Network Approximation

- Yarotsky[7] showed that polynomials can be efficiently approximated with deep ReLU networks
- Using this, we can efficiently approximate (say) wavelets

[7] Dmitry Yarotsky. "Error bounds for approximations with deep ReLU networks". In: *Neural Networks* 94 (2017), pp. 103–114.

[8] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

First Approach to Deep Network Approximation

- Yarotsky[7] showed that polynomials can be efficiently approximated with deep ReLU networks
- Using this, we can efficiently approximate (say) wavelets
- If $f_1, ..., f_n \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R})$, then

$$\sum_{i=1}^{n} f_i \in \Upsilon^{W+1,nL}(\mathbb{R}^d, \mathbb{R}). \tag{9}$$

- So, up to logarithmic factors, deep networks recover the $O(L^{-s/d})$ classical rate as long as we have a compact Sobolev embedding[8]

[7] Dmitry Yarotsky. "Error bounds for approximations with deep ReLU networks". In: *Neural Networks* 94 (2017), pp. 103–114.

[8] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

First Approach to Deep Network Approximation

- Yarotsky[7] showed that polynomials can be efficiently approximated with deep ReLU networks
- Using this, we can efficiently approximate (say) wavelets
- If $f_1, ..., f_n \in \Upsilon^{W,L}(\mathbb{R}^d, \mathbb{R})$, then

$$\sum_{i=1}^{n} f_i \in \Upsilon^{W+1,nL}(\mathbb{R}^d, \mathbb{R}). \tag{9}$$

- So, up to logarithmic factors, deep networks recover the $O(L^{-s/d})$ classical rate as long as we have a compact Sobolev embedding[8]
- Can we do better?

[7] Dmitry Yarotsky. "Error bounds for approximations with deep ReLU networks". In: *Neural Networks* 94 (2017), pp. 103–114.

[8] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Yes: Superconvergence!

- A fascinating result discovered by Yarotsky[9]:

Theorem

Suppose that $p = q = \infty$ and $0 < s \leq 1$. So $W^s(L_\infty(\Omega))$ is the class of s-Hölder continuous functions. Then for sufficiently large W (depending upon d)

$$\inf_{f_L \in \Upsilon^{W,L}(\mathbb{R}^d)} \|f - f_L\|_{L_\infty(\Omega)} \leq C\|f\|_{W^s(L_\infty(\Omega))} L^{-2s/d}. \tag{10}$$

[9] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

Yes: Superconvergence!

- A fascinating result discovered by Yarotsky[9]:

Theorem

Suppose that $p = q = \infty$ and $0 < s \leq 1$. So $W^s(L_\infty(\Omega))$ is the class of s-Hölder continuous functions. Then for sufficiently large W (depending upon d)

$$\inf_{f_L \in \Upsilon^{W,L}(\mathbb{R}^d)} \|f - f_L\|_{L_\infty(\Omega)} \leq C\|f\|_{W^s(L_\infty(\Omega))} L^{-2s/d}. \quad (10)$$

- This is sharp for deep ReLU networks

[9] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

Prior Work: Extensions

- Yarotsky's superconvergence result has been generalized[10] to $s > 1$

[10] Jianfeng Lu et al. "Deep network approximation for smooth functions". In: *SIAM Journal on Mathematical Analysis* 53.5 (2021), pp. 5465–5506.

[11] Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

[12] Sean Hon and Haizhao Yang. "Simultaneous neural network approximations in sobolev spaces". In: *arXiv preprint arXiv:2109.00161* (2021).

[13] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Prior Work: Extensions

- Yarotsky's superconvergence result has been generalized[10] to $s > 1$
- Optimal approximation rates when both depth and width vary[11]

[10] Jianfeng Lu et al. "Deep network approximation for smooth functions". In: *SIAM Journal on Mathematical Analysis* 53.5 (2021), pp. 5465–5506.

[11] Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

[12] Sean Hon and Haizhao Yang. "Simultaneous neural network approximations in sobolev spaces". In: *arXiv preprint arXiv:2109.00161* (2021).

[13] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Prior Work: Extensions

- Yarotsky's superconvergence result has been generalized[10] to $s > 1$
- Optimal approximation rates when both depth and width vary[11]
- Derivatives can also be approximated[12] if $s > 1$

[10] Jianfeng Lu et al. "Deep network approximation for smooth functions". In: *SIAM Journal on Mathematical Analysis* 53.5 (2021), pp. 5465–5506.

[11] Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

[12] Sean Hon and Haizhao Yang. "Simultaneous neural network approximations in sobolev spaces". In: *arXiv preprint arXiv:2109.00161* (2021).

[13] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Prior Work: Extensions

- Yarotsky's superconvergence result has been generalized[10] to $s > 1$
- Optimal approximation rates when both depth and width vary[11]
- Derivatives can also be approximated[12] if $s > 1$
- Interpolation with first approach to get rates in the non-linear regime[13]
  - Yields rate $L^{-\kappa s/d}$ with $1 < \kappa < 2$

---

[10] Jianfeng Lu et al. "Deep network approximation for smooth functions". In: *SIAM Journal on Mathematical Analysis* 53.5 (2021), pp. 5465–5506.

[11] Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

[12] Sean Hon and Haizhao Yang. "Simultaneous neural network approximations in sobolev spaces". In: *arXiv preprint arXiv:2109.00161* (2021).

[13] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Main Problem

- Our interest: What is the optimal rate for all pairs s, p, q for which we have a (compact) embedding?
    - Do we get superconvergence in the non-linear regime (i.e. when $q < p \leq \infty$)?
    - Existing superconvergence results only apply when $q = \infty$

[14] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Main Problem

- Our interest: What is the optimal rate for all pairs s, p, q for which we have a (compact) embedding?
  - Do we get superconvergence in the non-linear regime (i.e. when $q < p \leq \infty$)?
  - Existing superconvergence results only apply when $q = \infty$
- Two key difficulties[14]:
  - Upper Bounds: Existing methods only give superconvergence in linear regime

[14] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Main Problem

- Our interest: What is the optimal rate for all pairs s, p, q for which we have a (compact) embedding?
  - Do we get superconvergence in the non-linear regime (i.e. when $q < p \leq \infty$)?
  - Existing superconvergence results only apply when $q = \infty$
- Two key difficulties[14]:
  - Upper Bounds: Existing methods only give superconvergence in linear regime
  - Lower Bounds: Existing approaches only give lower bounds when $p = \infty$

[14] Ronald DeVore, Boris Hanin, and Guergana Petrova. "Neural Network Approximation". In: *arXiv preprint arXiv:2012.14501* (2020).

Main Result: Upper Bounds[15]

Theorem

Let $\Omega = [0,1]^d$ be the unit cube and let $0 < s < \infty$ and $1 \leq q \leq p \leq \infty$. Assume that $1/q - 1/p < s/d$, which guarantees that we have the compact Sobolev embedding

$$W^s(L_q(\Omega)) \subset\subset L^p(\Omega). \tag{11}$$

Then there exists an absolute constant $K < \infty$ and such that

$$\inf_{f_L \in \Upsilon^{Kd,L}(\mathbb{R}^d)} \|f - f_L\|_{L_p(\Omega)} \lesssim \|f\|_{W^s(L_q(\Omega))} L^{-2s/d}. \tag{12}$$

- We obtain superconvergence in all cases!

[15] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]
- Suppose that $\mathbf{x} \in \{0,1\}^N$

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]
- Suppose that $\mathbf{x} \in \{0,1\}^N$
- How many parameters do we need to represent $\mathbf{x}$?
  - i.e. want a network f, s.t. $f(i) = \mathbf{x}_i$ for $i = 0, ..., N-1$.

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]
- Suppose that $\mathbf{x} \in \{0, 1\}^N$
- How many parameters do we need to represent $\mathbf{x}$?
  - i.e. want a network f, s.t. $f(i) = \mathbf{x}_i$ for $i = 0, ..., N - 1$.
- Naively, we would need $O(N)$ parameters
  - Say use a piecewise linear function

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]
- Suppose that $\mathbf{x} \in \{0,1\}^N$
- How many parameters do we need to represent $\mathbf{x}$?
  - i.e. want a network f, s.t. $f(i) = \mathbf{x}_i$ for $i = 0, ..., N-1$.
- Naively, we would need $O(N)$ parameters
  - Say use a piecewise linear function
- Remarkably, we only need $O(\sqrt{N})$!

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction

- The key to superconvergence is the *bit-extraction* technique[16]
- Suppose that $\mathbf{x} \in \{0,1\}^N$
- How many parameters do we need to represent $\mathbf{x}$?
  - i.e. want a network f, s.t. $f(i) = \mathbf{x}_i$ for $i = 0, ..., N-1$.
- Naively, we would need $O(N)$ parameters
  - Say use a piecewise linear function
- Remarkably, we only need $O(\sqrt{N})$!
- Previous results proved by combining bit-extraction with piecewise polynomial approximation on a *regular* grid
  - Works in the linear regime $p \leq q$
  - Works for all spaces which admit suitable piecewise polynomial approximations

[16] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Bit Extraction (cont.)

- Divide $\{0, 1, ..., N-1\}$ into $O(\sqrt{N})$ sub-intervals of $I_1, ..., I_n$ of length $O(\sqrt{N})$
  - $I_j = \{k_j, k_j + 1, ..., k_{j+1} - 1\}$

Bit Extraction (cont.)

- Divide $\{0, 1, \ldots, N-1\}$ into $O(\sqrt{N})$ sub-intervals of $I_1, \ldots, I_n$ of length $O(\sqrt{N})$
  - $I_j = \{k_j, k_j + 1, \ldots, k_{j+1} - 1\}$
- Two piecewise linear functions:
  - Map I_j to k_j
  - Map I_j to $b_j = 0.\mathbf{x}_{k_j} \ldots \mathbf{x}_{k_{j+1}-1}$
  - Requires $O(\sqrt{N})$ layers

Bit Extraction (cont.)

- Divide $\{0, 1, \ldots, N-1\}$ into $O(\sqrt{N})$ sub-intervals of $I_1, \ldots, I_n$ of length $O(\sqrt{N})$
  - $I_j = \{k_j, k_j + 1, \ldots, k_{j+1} - 1\}$
- Two piecewise linear functions:
  - Map I_j to k_j
  - Map I_j to $b_j = 0.\mathbf{x}_{k_j} \ldots \mathbf{x}_{k_{j+1}-1}$
  - Requires $O(\sqrt{N})$ layers
- Construct network which maps

$$\begin{pmatrix} i \\ k \\ 0.x_1 x_2 \cdots x_n \\ z \end{pmatrix} \to \begin{pmatrix} i-1 \\ k \\ 0.x_2 \cdots x_n \\ z + x_1 \chi(i = k) \end{pmatrix} \tag{13}$$

  - Can be done with a constant size network
  - Compose this $O(\sqrt{N})$ times

Efficient Representation of Sparse Vectors[17]

- Approximation in non-linear regime ($p > q$) requires *adaptivity* or *sparsity*

[17] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Efficient Representation of Sparse Vectors[17]

- Approximation in non-linear regime ($p > q$) requires *adaptivity* or *sparsity*

Proposition

Let $M \geq 1$ *and* $N \geq 1$ *and* $\mathbf{x} \in \mathbb{Z}^N$ *be an* N*-dimensional vector satisfying*

$$\|\mathbf{x}\|_{\ell^1} \leq M. \tag{14}$$

- *Then if* $N \geq M$*, there exists a neural network* $g \in \Upsilon^{17,L}(\mathbb{R}, \mathbb{R})$ *with depth* $L \leq C\sqrt{M(1 + \log(N/M))}$ *which satisfies* $g(i) = \mathbf{x}_i$ *for* $i = 1, ..., N$.
- *Further, if* $N < M$*, then there exists a neural network* $g \in \Upsilon^{21,L}(\mathbb{R}, \mathbb{R})$ *with depth* $L \leq C\sqrt{N(1 + \log(M/N))}$ *which satisfies* $g(i) = \mathbf{x}_i$ *for* $i = 1, ..., N$.

[17] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

VC-dimension

- Let $\mathcal{F}$ be a class of functions

VC-dimension

- Let $\mathcal{F}$ be a class of functions
- A set of points $x_1, ..., x_N$ is shattered by $\mathcal{F}$ if for any $\epsilon_1, ..., \epsilon_N \in \{\pm 1\}$ there exists an $f \in \mathcal{F}$ such that

$$\text{sign}(f(x_i)) = \epsilon_i \tag{15}$$

VC-dimension

- Let $\mathcal{F}$ be a class of functions
- A set of points $x_1, ..., x_N$ is shattered by $\mathcal{F}$ if for any $\epsilon_1, ..., \epsilon_N \in \{\pm 1\}$ there exists an $f \in \mathcal{F}$ such that

$$\mathrm{sign}(f(x_i)) = \epsilon_i \tag{15}$$

- The VC-dimension of $\mathcal{F}$ is the largest N such that $\mathcal{F}$ shatters a set of N points
  - Degree d polynomials have VC-dimension $d+1$
  - Linear functions on $\mathbb{R}^d$ have VC-dimension $d+1$

L_∞ Lower Bounds

- Consider a grid of N^d points $\{0, 1/N, 2/N, ..., (N-1)/N\}^d$

[18] Nick Harvey, Christopher Liaw, and Abbas Mehrabian. "Nearly-tight VC-dimension bounds for piecewise linear neural networks". In: *Conference on learning theory*. PMLR. 2017, pp. 1064–1068, Paul Goldberg and Mark Jerrum. "Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 361–369.

[19] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

L_∞ Lower Bounds

- Consider a grid of N^d points $\{0, 1/N, 2/N, ..., (N-1)/N\}^d$
- We can interpolate the values $c\epsilon_i N^{-s}$ by a function $f \in F^s_\infty(\Omega)$
  - Here ϵ_i represent arbitrary signs at the grid points

[18] Nick Harvey, Christopher Liaw, and Abbas Mehrabian. "Nearly-tight VC-dimension bounds for piecewise linear neural networks". In: *Conference on learning theory*. PMLR. 2017, pp. 1064–1068, Paul Goldberg and Mark Jerrum. "Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 361–369.

[19] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

L_∞ Lower Bounds

- Consider a grid of N^d points $\{0, 1/N, 2/N, ..., (N-1)/N\}^d$
- We can interpolate the values $c\epsilon_i N^{-s}$ by a function $f \in F^s_\infty(\Omega)$
  - Here ϵ_i represent arbitrary signs at the grid points
- The VC-dimension of $\Upsilon^{W,L}(\mathbb{R}^d)$ is bounded by[18]

$$CW^3L^2 \tag{16}$$

[18] Nick Harvey, Christopher Liaw, and Abbas Mehrabian. "Nearly-tight VC-dimension bounds for piecewise linear neural networks". In: *Conference on learning theory*. PMLR. 2017, pp. 1064–1068, Paul Goldberg and Mark Jerrum. "Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 361–369.

[19] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

L_∞ Lower Bounds

- Consider a grid of N^d points $\{0, 1/N, 2/N, ..., (N-1)/N\}^d$
- We can interpolate the values $c\epsilon_i N^{-s}$ by a function $f \in F^s_\infty(\Omega)$
  - Here ϵ_i represent arbitrary signs at the grid points
- The VC-dimension of $\Upsilon^{W,L}(\mathbb{R}^d)$ is bounded by[18]

$$CW^3L^2 \tag{16}$$

- This gives lower bounds when[19] $p = \infty$

[18] Nick Harvey, Christopher Liaw, and Abbas Mehrabian. "Nearly-tight VC-dimension bounds for piecewise linear neural networks". In: *Conference on learning theory*. PMLR. 2017, pp. 1064–1068, Paul Goldberg and Mark Jerrum. "Bounding the Vapnik-Chervonenkis dimension of concept classes parameterized by real numbers". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 361–369.

[19] Dmitry Yarotsky. "Optimal approximation of continuous functions by very deep ReLU networks". In: *arXiv preprint arXiv:1802.03620* (2018), Zuowei Shen, Haizhao Yang, and Shijun Zhang. "Optimal approximation rate of ReLU networks in terms of width and depth". In: *Journal de Mathématiques Pures et Appliquées* 157 (2022), pp. 101–135.

Main Result: Lower Bounds[21]

- Remarkably, we can still use VC-dimension when $p < \infty$!

Theorem

Suppose that K is a translation invariant class of functions whose VC-dimension is at most n. Then for any $p > 0$ there exists an $f \in W^s(L_\infty(\Omega))$ such that

$$\inf_{g\in K} \|f - g\|_{L^p(\Omega)} \geq C(d,p)n^{-\frac{s}{d}}\|f\|_{W^s(L_\infty(\Omega))}. \tag{17}$$

- Argument uses the Sauer-Shelah lemma[20] plus entropy arguments

[20] Saharon Shelah. "A combinatorial problem; stability and order for models and theories in infinitary languages". In: *Pacific Journal of Mathematics* 41.1 (1972), pp. 247–261.

[21] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Main Result: Lower Bounds[21]

- Remarkably, we can still use VC-dimension when $p < \infty$!

Theorem

Suppose that K is a translation invariant class of functions whose VC-dimension is at most n. Then for any $p > 0$ there exists an $f \in W^s(L_\infty(\Omega))$ such that

$$\inf_{g \in K} \|f - g\|_{L^p(\Omega)} \geq C(d,p) n^{-\frac{s}{d}} \|f\|_{W^s(L_\infty(\Omega))}. \tag{17}$$

- Argument uses the Sauer-Shelah lemma[20] plus entropy arguments
- Implies $L^{-2s/d}$ is sharp, optimal in terms of parameter count

[20] Saharon Shelah. "A combinatorial problem; stability and order for models and theories in infinitary languages". In: *Pacific Journal of Mathematics* 41.1 (1972), pp. 247–261.

[21] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Fundamental Lower Bound: Metric Entropy

Definition (Kolmogorov)

Let X be a Banach space and $B \subset X$. The metric entropy numbers of B, $\epsilon_n(B)_X$ are given by

$$\epsilon_n(B)_X = \inf\{\epsilon : \ B \text{ is covered by } 2^n \text{ balls of radius } \epsilon\}. \tag{18}$$

- Roughly speaking, $\epsilon_n(B)_K$ measures how accurately elements of B can be specified with n bits.

[22] Albert Cohen et al. "Optimal stable nonlinear approximation". In: *Foundations of Computational Mathematics* (2021), pp. 1–42.

[23] M Š Birman and MZ Solomjak. "Piecewise-polynomial approximations of functions of the classes W_p^α". In: *Mathematics of the USSR-Sbornik* 2.3 (1967), p. 295.

Fundamental Lower Bound: Metric Entropy

Definition (Kolmogorov)

Let X be a Banach space and $B \subset X$. The metric entropy numbers of B, $\epsilon_n(B)_X$ are given by

$$\epsilon_n(B)_X = \inf\{\epsilon : \ B \text{ is covered by } 2^n \text{ balls of radius } \epsilon\}. \tag{18}$$

- Roughly speaking, $\epsilon_n(B)_K$ measures how accurately elements of B can be specified with n bits.
- Gives a fundamental lower bound on the rates of stable approximation[22]

[22] Albert Cohen et al. "Optimal stable nonlinear approximation". In: *Foundations of Computational Mathematics* (2021), pp. 1–42.

[23] M Š Birman and MZ Solomjak. "Piecewise-polynomial approximations of functions of the classes W_p^α". In: *Mathematics of the USSR-Sbornik* 2.3 (1967), p. 295.

Fundamental Lower Bound: Metric Entropy

Definition (Kolmogorov)

Let X be a Banach space and $B \subset X$. The metric entropy numbers of B, $\epsilon_n(B)_X$ are given by

$$\epsilon_n(B)_X = \inf\{\epsilon : \ B \text{ is covered by } 2^n \text{ balls of radius } \epsilon\}. \tag{18}$$

- Roughly speaking, $\epsilon_n(B)_K$ measures how accurately elements of B can be specified with n bits.
- Gives a fundamental lower bound on the rates of stable approximation[22]
- If compact Sobolev embedding holds, then[23]

$$\epsilon_n(B^s(L_q(\Omega)))_{L^p(\Omega)} \eqsim n^{-s/d} \tag{19}$$

[22] Albert Cohen et al. "Optimal stable nonlinear approximation". In: *Foundations of Computational Mathematics* (2021), pp. 1–42.

[23] M Š Birman and MZ Solomjak. "Piecewise-polynomial approximations of functions of the classes W_p^α". In: *Mathematics of the USSR-Sbornik* 2.3 (1967), p. 295.

Continuous Lower Bound: Bernstein n-widths

Definition (Bernstein)

Let X be a Banach space and $B \subset X$. The Bernstein n-widths of B are

$$b_n(B)_X = \sup_{\mathcal{F}_n \subset X} \sup\{r \geq 0 : B_r(\mathcal{F}_n) \subset X \cap \mathcal{F}_n\}, \quad (20)$$

where the supremum is over all linear subspaces $\mathcal{F}_n$ of dimension $n+1$ and $B_r(\mathcal{F}_n)$ is the ball of radius r in the subspace $B_r(\mathcal{F}_n)$.

- For continuous approximation methods, we have[24]

$$\sup_{f \in B} \|f_n - f\|_X \geq b_n(B)_X \quad (21)$$

- $b_n(F_2^s)_{L_2(\Omega)} \eqsim n^{-s/d}$
  - Superconvergence parameter selection must be discontinuous

[24] Ronald A DeVore, Ralph Howard, and Charles Micchelli. "Optimal nonlinear approximation". In: *Manuscripta mathematica* 63.4 (1989), pp. 469–478.

Deep Network Interpolation

- Suppose we have points $x_1, ..., x_N \in \mathbb{R}$ and values $y_1, ..., y_N$

[25] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Deep Network Interpolation

- Suppose we have points $x_1, ..., x_N \in \mathbb{R}$ and values $y_1, ..., y_N$
- How many parameters does a deep network need to interpolate, i.e. want $f \in \Upsilon^{W,L}(\mathbb{R})$ s.t. $f(x_i) = y_i$

[25] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Deep Network Interpolation

- Suppose we have points $x_1, ..., x_N \in \mathbb{R}$ and values $y_1, ..., y_N$
- How many parameters does a deep network need to interpolate, i.e. want $f \in \Upsilon^{W,L}(\mathbb{R})$ s.t. $f(x_i) = y_i$
- If x_i are *evenly spaced* and $y_i \in \{0, 1\}$ then we need only $O(\sqrt{N})$ parameters
  - Bit extraction[25]

[25] Peter Bartlett, Vitaly Maiorov, and Ron Meir. "Almost linear VC dimension bounds for piecewise polynomial networks". In: *Advances in neural information processing systems* 11 (1998).

Continuous Values[26]

- Suppose we want to interpolate arbitrary real values, i.e. $y_i \in \mathbb{R}$?

[26] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Continuous Values[26]

- Suppose we want to interpolate arbitrary real values, i.e. $y_i \in \mathbb{R}$?
- Need $\Omega(N)$ parameters
  - No bit extraction possible!

Theorem

Let $x_1, ..., x_N$ be given. Suppose that for any $y_1, ..., y_n \in \mathbb{R}$ there is an $f \in \Upsilon^{W,L}(\mathbb{R})$ such that $f(x_i) = y_i$. Then the number of parameters $P = W^2L \geq cn$ for an absolute constant c.

[26] Jonathan W Siegel. "Optimal Approximation Rates for Deep ReLU Neural Networks on Sobolev Spaces". In: *arXiv preprint arXiv:2211.14400* (2022).

Arbitrary Interpolation Points[27]

- Suppose we want to interpolate at arbitrary points $x_1, ..., x_N \in \mathbb{R}$

[27] Jonathan W Siegel. "Sharp Lower Bounds on Interpolation by Deep ReLU Neural Networks at Irregularly Spaced Data". In: *arXiv preprint arXiv:2302.00834* (2023), Eduardo D Sontag. "Shattering all sets of 'k'points in "general position" requires (k—1)/2 parameters". In: *Neural Computation* 9.2 (1997), pp. 337–348.

Arbitrary Interpolation Points[27]

- Suppose we want to interpolate at arbitrary points $x_1, ..., x_N \in \mathbb{R}$
- Need $\Omega(N)$ parameters
    - No bit extraction possible!

Theorem

Suppose that the neural network class $\Upsilon^{W,L}(\mathbb{R})$ can shatter **every** *set of n points. Then the number of parameters $P = W^2L \geq cn$ for an absolute constant c.*

[27] Jonathan W Siegel. "Sharp Lower Bounds on Interpolation by Deep ReLU Neural Networks at Irregularly Spaced Data". In: *arXiv preprint arXiv:2302.00834* (2023), Eduardo D Sontag. "Shattering all sets of 'k'points in "general position" requires (k—1)/2 parameters". In: *Neural Computation* 9.2 (1997), pp. 337–348.

The Sobolev Endpoint Case

- We can use these results to understand the Sobolev endpoint

The Sobolev Endpoint Case

- We can use these results to understand the Sobolev endpoint
- Consider $W^1(L_1([0,1])) \subset L_\infty([0,1])$

The Sobolev Endpoint Case

- We can use these results to understand the Sobolev endpoint
- Consider $W^1(L_1([0,1])) \subset L_\infty([0,1])$
- If we can get approximation error $1/N$, then we must be able to shatter any set of N points

The Sobolev Endpoint Case

- We can use these results to understand the Sobolev endpoint
- Consider $W^1(L_1([0,1])) \subset L_\infty([0,1])$
- If we can get approximation error $1/N$, then we must be able to shatter any set of N points
- Implies that the optimal rate for $W^1(L_1([0,1]))$ in $L_\infty([0,1])$ is $O(P^{-1})$ (*no superconvergence*)

Conclusion

- Determined sharp approximation rates for deep ReLU networks on Sobolev spaces

Conclusion

- Determined sharp approximation rates for deep ReLU networks on Sobolev spaces
- Some open problems:
  - Sobolev endpoint is more subtle
  - Obtain a similar theory for shallow neural networks
  - Extensions to other activation functions and architectures
  - Understanding the optimization process and generalization of deep networks as well

Thank you for your attention!