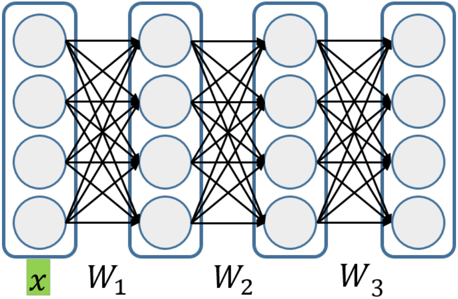
Adaptivity of deep ReLU network and its generalization error analysis

Taiji Suzuki[†,‡]

[†]The University of Tokyo
Department of Mathematical Informatics
[‡]AIP-RIKEN

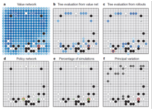
22nd/Feb/2019
The 2nd Korea-Japan Machine Learning Workshop

Deep learning model

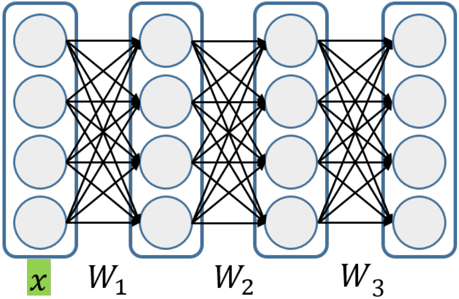


$$f(x) = \eta(W_L \eta(W_{L-1} \dots W_2 \eta(W_1 x + b_1) + b_2 \dots))$$

- High performance learning system
- Many applications: Deepmind, Google, Facebook, Open AI, Baidu, ...




Deep learning model



$$f(x) = \eta(W_L \eta(W_{L-1} \ldots W_2 \eta(W_1 x + b_1) + b_2 \ldots))$$

- High performance learning system
- Many applications: Deepmind, Google, Facebook, Open AI, Baidu, ...

We need theories.

Outline of this talk

Why does deep learning perform so well?

"Adaptivity" of deep neural network:

- Adaptivity to the shape of the target function.
- Adaptivity to the dimensionality of the input data.

$\rightarrow$ sparsity, non-convexity

Outline of this talk

Why does deep learning perform so well?

"Adaptivity" of deep neural network:

- Adaptivity to the shape of the target function.
- Adaptivity to the dimensionality of the input data.

  $\rightarrow$ sparsity, non-convexity

Approach:

- Estimation error analysis on a Besov space.
  - spatial inhomogeneity of smoothness
  - avoiding curse of dimensionality
- Will be shown that any linear estimators such as kernel methods are outperformed by DL.

Reference

Taiji Suzuki:
Adaptivity of deep ReLU network for learning in Besov and mixed smooth Besov spaces: optimal rate and curse of dimensionality.
ICLR2019, to appear. (arXiv:1810.08033).

Outline

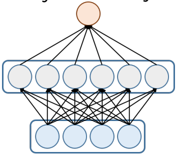
Universal approximator

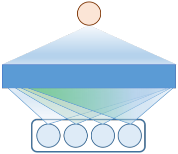
Two layer neural network:

$$f(x) = \sum_{j=1}^{m} v_j \eta(w_j^\top x + b_j).$$

As $m \to \infty$, the two layer network can approximate an arbitrary function with an arbitrary precision.

$$\hat{f}(x) = \textstyle\sum_{j=1}^{m} v_j \eta(w_j^\top x + b_j) \quad \simeq \quad f^{\mathrm{o}}(x) = \int h^{\mathrm{o}}(w,b)\eta(w^\top x + b)\mathrm{d}w\mathrm{d}b$$





(Sonoda & Murata, 2015)

Year		Basis function	space
1987	Hecht-Nielsen	–	$C(\mathbb{R}^d)$
1988	Gallant & White	Cos	$L_2(K)$
	Irie & Miyake	integrable	$L_2(\mathbb{R}^d)$
1989	Carroll & Dickinson	Continuous sigmoidal	$L_2(K)$
	Cybenko	Continuous sigmoidal	$C(K)$
	Funahashi	Monotone & bounded	$C(K)$
1993	Mhaskar & Micchelli	Polynomial growth	$C(K)$
2015	Sonoda & Murata	admissible	L_1, L_2

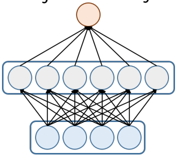
Universal approximator

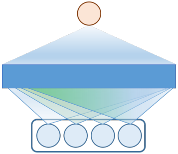
Two layer neural network:

$$f(x) = \sum_{j=1}^{m} v_j \eta(w_j^\top x + b_j).$$

As $m \to \infty$, the two layer network can approximate an arbitrary function with an arbitrary precision.

$$\hat{f}(x) = \sum_{j=1}^{m} v_j \eta(w_j^\top x + b_j) \quad \simeq \quad f^{\mathrm{o}}(x) = \int h^{\mathrm{o}}(w,b)\eta(w^\top x + b)\mathrm{d}w\mathrm{d}b$$

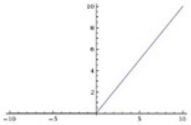


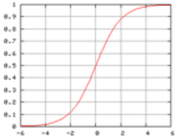


(Sonoda & Murata, 2015)

Activation functions:

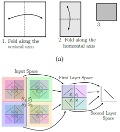
ReLU: $\eta(u) = \max\{u, 0\}$ **Sigmoid:** $\eta(u) = \frac{1}{1+\exp(-u)}$

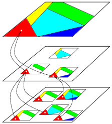


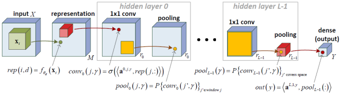


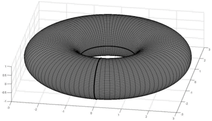
Expressive power of deep neural network

- **Combinatorics/Hyperplane Arrangements** (Montufar et al., 2014)
  Number of linear regions (ReLU)
- **Polynomial expansions, tensor analysis** (Cohen et al., 2016; Cohen & Shashua, 2016)
  Number of monomials (Sum product)
- **Algebraic topology** (Bianchini & Scarselli, 2014)
  Betti numbers (Pfaffian)
- **Riemannian geometry + Dynamic mean field theory** (Poole et al., 2016)
  Extrinsic curvature





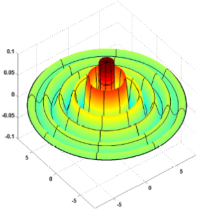


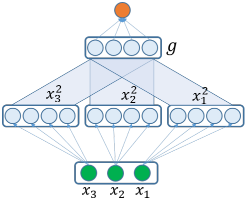


Deep neural network has exponentially large power of expression against the number of layers.

Depth separation between 2 and 3 layers

2 layer NN is already universal approximator. **When is deeper network useful?**



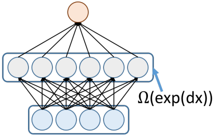


○

There is a function represented by

$$f^{\mathrm{o}}(x) = g(\|x\|^2) = g(x_1^2 + \cdots + x_d^2)$$

that can be better approximated by 3 layer NN than 2 layer NN (c.f., Eldan and Shamir (2016))



×

d_x: the dimension of the input x

- 3 layers: $O(\mathrm{poly}(d_x, \epsilon^{-1}))$ internal nodes are sufficient.
- 2 layers: At least $\Omega(1/\epsilon^{d_x})$ internal nodes are required.

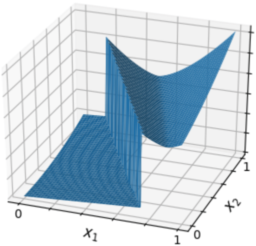
$\rightarrow$ **DL can avoid curse of dimensionality.**

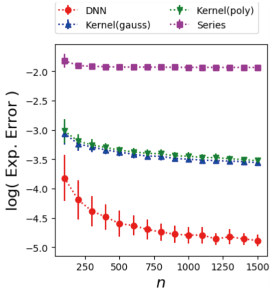
Non-smooth function

For estimating a *non-smooth function*, deep is better (Imaizumi & Fukumizu, 2018):

$$f^{\mathrm{o}}(x) = \sum_{k=1}^{K} \mathbf{1}_{R_k}(x) h_k(x)$$

where R_k is a region with smooth boundary and h_k is a smooth function.





Depth separation

What makes difference between deep and shallow methods?

Depth separation

What makes difference between deep and shallow methods?

$\rightarrow$ **Non-convexity of the model (sparseness)**

Easy example: Linear activation

Reduced rank regression:

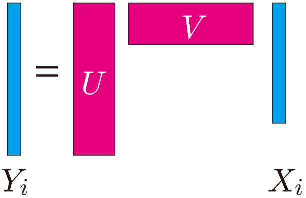
$$Y_i = UVX_i + \xi_i \quad (i = 1, \dots, n)$$

where $U \in \mathbb{R}^{M \times r}, V \in \mathbb{R}^{r \times N}$ $(r \ll M, N)$, and $Y_i \in \mathbb{R}^M, X_i \in \mathbb{R}^N$.

- Linear estimator $\hat{f}(x) = \sum_{i=1}^n Y_i \varphi(X_1, \dots, X_n, x)$,
- Deep learning $\hat{f}(x) = \hat{U}\hat{V}x$.

$$\frac{r(M+N)}{n} \ll \frac{MN}{n}$$

Deep　　　Shallow



Non-convexity is essential. → sparsity.

Nonlinear regression problem

Nonlinear regression problem:

$$y_i = f^{\mathrm{o}}(x_i) + \xi_i \quad (i = 1, \ldots, n),$$

where $\xi_i \sim N(0, \sigma^2)$, and $x_i \sim P_X([0,1]^d)$ (i.i.d.).

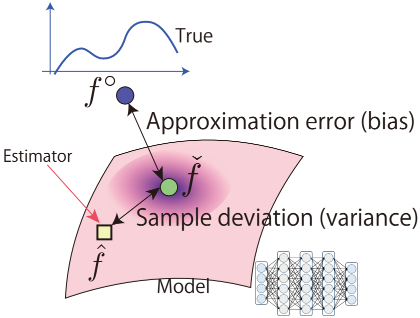
We want to estimate f^{o} from data $(x_i, y_i)_{i=1}^n$.

Least squares estimator:

$$\hat{f} = \operatorname*{argmin}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

where $\mathcal{F}$ is a neural network model.

Bias and variance trade-off



$$\underbrace{\|f^{\mathrm{o}} - \hat{f}\|_{L_2(P)}}_{\text{Estimation error}} \leq \underbrace{\|f^{\mathrm{o}} - \check{f}\|_{L_2(P)}}_{\substack{\text{Approximation error} \\ \text{(bias)}}} + \underbrace{\|\check{f} - \hat{f}\|_{L_2(P)}}_{\substack{\text{Sample deviation} \\ \text{(variance)}}}$$

- Large model: small approximation error, large sample deviation
- Small model: large approximation error, small sample deviation

$\rightarrow$ **Bias and variance trade-off**

Outline

Agenda of this talk

Deep learning can make use of *sparsity*.

Appropriate function class with non-convexity:

- Q: A typical setting is Hölder space. Can we generalize it?
- A: <u>**Besov space**</u> and <u>**mixed-smooth Besov space**</u> (tensor product space)

Curse of dimensionality:

- Q: Deep learning can suffer from curse of dimensionality.
  Can we ease the effect of dimensionality under a suitable condition?
- A: Yes, if the true function is included in <u>**mixed-smooth Besov space**</u>.

Outline

Minimax optimal framework

What is a "good" estimator?

- Minimax optimal rate:

$$\inf_{\hat{f}:\text{estimator}} \sup_{f^{\mathrm{o}} \in \mathcal{F}} \mathrm{E}[\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2] \leq n^{-?}$$

$\rightarrow$ If an estimator $\hat{f}$ achieves the minimax optimal rate, then it can be seen a "good" estimator.

What kind $\mathcal{F}$ do we think?

Hölder, Sobolev, Besov

$\Omega = [0,1]^d \subset \mathbb{R}^d$

- **Hölder space** $(\mathcal{C}^\beta(\Omega))$

$$\|f\|_{\mathcal{C}^\beta} = \max_{|\alpha|\le m} \|\partial^\alpha f\|_\infty + \max_{|\alpha|=m} \sup_{x\in\Omega} \frac{|\partial^\alpha f(x) - \partial^\alpha f(y)|}{|x-y|^{\beta-m}}$$

- **Sobolev space** $(W_p^k(\Omega))$

$$\|f\|_{W_p^k} = \left(\sum_{|\alpha|\le k} \|D^\alpha f\|_{L^p(\Omega)}^p \right)^{\frac{1}{p}}$$

- **Besov space** $(B_{p,q}^s(\Omega))$ $(0 < p, q \le \infty, 0 < s \le m)$

$$\omega_m(f,t)_p := \sup_{\|h\|\le t} \left\| \sum_{j=1}^m (-1)^{m-j} \binom{m}{j} f(\cdot + jh) \right\|_{L^p(\Omega)},$$

$$\|f\|_{B_{p,q}^s(\Omega)} = \|f\|_{L^p(\Omega)} + \left(\int_0^\infty [t^{-s}\omega_m(f,t)_p]^q \frac{\mathrm{d}t}{t} \right)^{1/q}.$$

Relation between the spaces

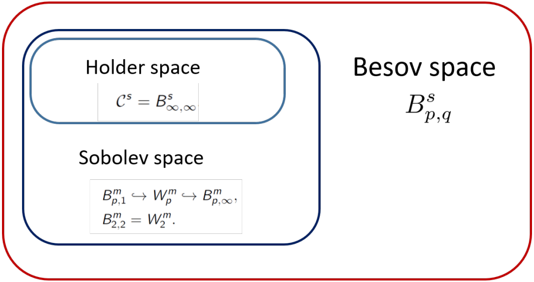
Suppose $\Omega = [0,1]^d \subset \mathbb{R}$.

- For $m \in \mathbb{N}$,

$$B^m_{p,1} \hookrightarrow W^m_p \hookrightarrow B^m_{p,\infty},$$
$$B^m_{2,2} = W^m_2.$$

- For $0 < s < \infty$ and $s \notin \mathbb{N}$,

$$\mathcal{C}^s = B^s_{\infty,\infty}.$$



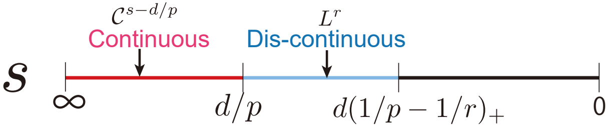
- Continuous regime: $s > d/p$

$$B^s_{p,q} \hookrightarrow C^0$$

- L^r-integrability: $s > d(1/p - 1/r)_+$

$$B^s_{p,q} \hookrightarrow L^r$$

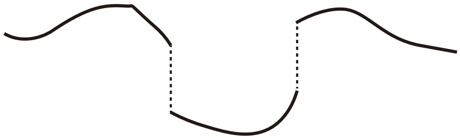
(If $d/p \geq s$, the elements are not necessarily continuous).



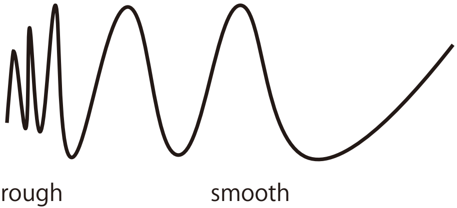
- Example: $B^1_{1,1}([0,1]) \subset \{\text{bounded total variation}\} \subset B^1_{1,\infty}([0,1])$

Properties of Besov space

- **Discontinuity:** $d/p > s$

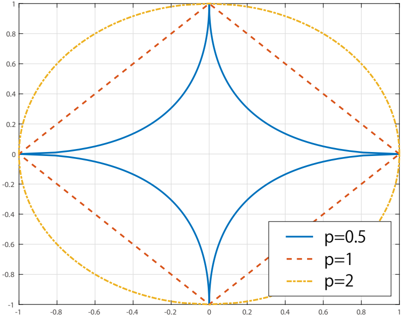


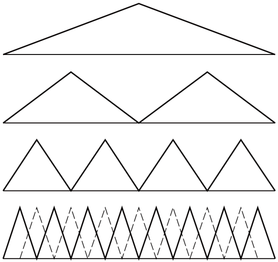
- **Spatial inhomogeneity of smoothness: small** p



Question: Can deep learning capture these properties?

Connection to sparsity





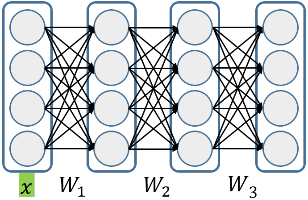
Multiresolution expansion

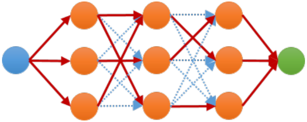
$$f = \sum_{k \in \mathbb{N}+} \sum_{j \in J(k)} \alpha_{k,j} \psi(2^k x - j),$$

$$\|f\|_{B^s_{p,q}} \simeq \left[\sum_{k=0}^{\infty} \{ 2^{sk} (2^{-kd} \sum_{j \in J(k)} |\alpha_{k,j}|^p)^{1/p} \}^q \right]^{1/q}$$

Sparse coefficients $\rightarrow$ spatial inhomogeneity of smoothness

Deep learning model



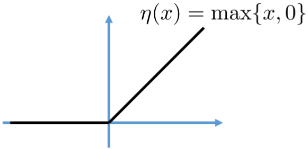


$$f(x) = (W^{(L)}\eta(\cdot) + b^{(L)}) \circ (W^{(L-1)}\eta(\cdot) + b^{(L-1)}) \circ \cdots \circ (W^{(1)}x + b^{(1)})$$

- $\mathcal{F}(L, W, S, B)$: deep networks with

  depth L**, width** W**, sparsity** S**, norm bound** B.

- η is **ReLU** activation: $\eta(u) = \max\{u, 0\}$.
  (currently most popular)



Approximation by deep NN in Besov space

$\mathcal{F}(L, W, S, B)$: deep networks with depth L, width W, sparsity S, norm bound B.

Proposition (Approximation ability for Besov space)

Suppose that $0 < p, q, r \leq \infty$ *and* $0 < s < \infty$ *satisfy* $m > 2s$ *and*

$$s > d(1/p - 1/r)_+$$

For $N \in \mathbb{N}$, *by setting*

$$L = 3\lceil \log_2 \left(\frac{3^{d \vee m} N^{\frac{s}{d}}}{c_{(d,m)}} \right) + 5 \rceil \lceil \log_2 (d \vee m) \rceil, \qquad W = 6N(d \vee m^2),$$

$$S = 6(L-1)(d \vee m^2) + N, \qquad B = O(N^{(d/p - s)_+}),$$

it holds that

$$\sup_{f^{\mathrm{o}} \in U(B^s_{p,q}([0,1]^d))} \inf_{\check{f} \in \mathcal{F}(L,W,S,B)} \|f^{\mathrm{o}} - \check{f}\|_{L^r([0,1]^d)} \lesssim N^{-s/d}.$$

Remark: Shallow network cannot achieve this rate.

Approximation by deep NN in Besov space

$\mathcal{F}(L, W, S, B)$: deep networks with depth L, width W, sparsity S, norm bound B.

Proposition (Approximation ability for Besov space)

Suppose that $0 < p, q, r \leq \infty$ *and* $0 < s < \infty$ *satisfy* $m > 2s$ *and*

$$s > d(1/p - 1/r)_+$$

For $N \in \mathbb{N}$*, by setting*

$$L = O(\log(N)), \qquad W = O(N),$$
$$S = O(N\log(N)), \qquad B = O(N^{(d/p-s)_+}),$$

it holds that

$$\sup_{f^\circ \in U(B^s_{p,q}([0,1]^d))} \inf_{\check{f} \in \mathcal{F}(L,W,S,B)} \|f^\circ - \check{f}\|_{L^r([0,1]^d)} \lesssim N^{-s/d}.$$

Remark: Shallow network cannot achieve this rate.

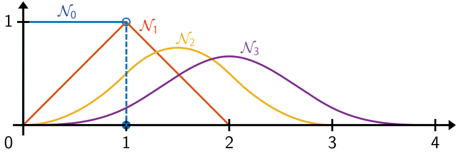
B-spline

$$\mathcal{N}(x) = \begin{cases} 1 & (x \in [0,1]), \\ 0 & (\text{otherwise}). \end{cases}$$

Cardinal B-spline of order m:

$$\mathcal{N}_m(x) = (\underbrace{\mathcal{N} * \mathcal{N} * \cdots * \mathcal{N}}_{m+1 \text{ times}})(x).$$

$\rightarrow$ Piece-wise polynomial of order m.



$$\mathcal{N}^{(d)}_{k,j}(x_1, \ldots, x_d) = \prod_{i=1}^{d} \mathcal{N}_m(2^k x_i - j_i)$$

Cardinal B-spline interpolation (DeVore & Popov, 1988)

- **Atomic decomposition**
  $f \in L^p$ is in $B^s_{p,q}$ if and only if f can be decomposed into

$$f = \sum_{k \in \mathbb{N}+} \sum_{j \in J(k)} \alpha_{k,j} \mathcal{N}^{(d)}_{k,j},$$

  (where $J(k) = \{j \in \mathbb{Z}^d \mid -m < j_i < 2^{k_i+1} + m\}$) such that

$$N(f) := \left[\sum_{k=0}^{\infty} \{2^{sk} (2^{-kd} \sum_{j \in J(k)} |\alpha_{k,j}|^p)^{1/p}\}^q \right]^{1/q} < \infty.$$

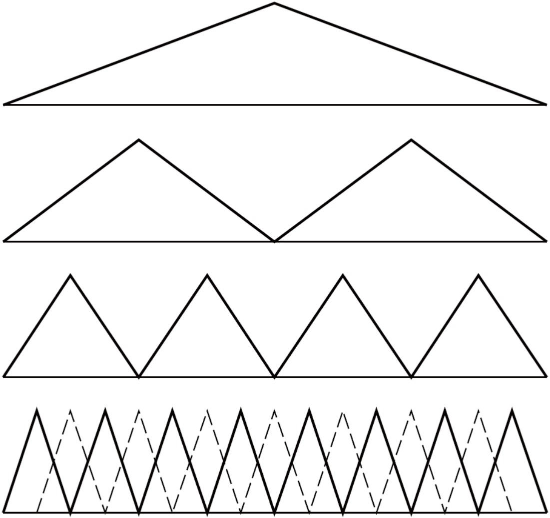
  ($\alpha_{k,j}$ is determined in a certain way.)

- **Norm equivalence**

$$\|f\|_{B^s_{p,q}} \simeq N(f).$$

Basic strategy: approximate each basis $\mathcal{N}^{(d)}_{k,j}$ by deep NN "efficiently".
※ cardinal B-spline is not a wavelet basis.

Cardinal B-spline expansion ($m = 1$)



Under the condition $s > d(1/p - 1/r)_+$, it holds that

$$\sup_{f^\circ \in U(B^s_{p,q}([0,1]^d))} \inf_{\check{f} \in \mathcal{F}(L,W,S,B)} \|f^\circ - \check{f}\|_{L^r([0,1]^d)} \lesssim N^{-s/d}.$$

- Setting $p = q = \infty$ and $r = \infty$, then $B^s_{p,q}(\Omega) = C^s(\Omega)$
  $\Rightarrow$ The result by Yarotsky (2016) is recovered as a special case.

Under the condition $s > d(1/p - 1/r)_+$, it holds that

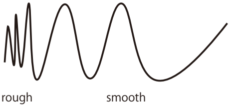
$$\sup_{f^\circ \in U(B^s_{p,q}([0,1]^d))} \inf_{\check{f} \in \mathcal{F}(L,W,S,B)} \|f^\circ - \check{f}\|_{L^r([0,1]^d)} \lesssim N^{-s/d}.$$

- Setting $p = q = \infty$ and $r = \infty$, then $B^s_{p,q}(\Omega) = C^s(\Omega)$
  $\Rightarrow$ The result by Yarotsky (2016) is recovered as a special case.

- **Nonlinear adaptive sampling recovery** is required (Dũng, 2011b).
  "Non-adaptive method" only achieves

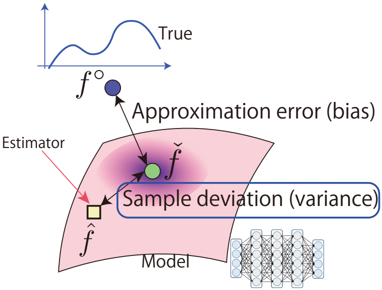
$$N^{-(s/d-(1/p-1/r)_+)},$$

for $1 < p < r \leq 2,\ s > d(1/p - 1/r)_+$ which is **not optimal if** $p < r$.
(Non-adaptive method: it uses N "fixed" bases to approximate the target function by $\sum_{i=1}^N \alpha_i \psi_i(x)$)
$\rightarrow$ **Methods with fixed bases cannot achieve the opt. rate!**



(small p situation)

Empirical risk minimization and estimation error



We have already obtained the approximation error.
Next, we derive the estimation error of the least squares estimator:

$$\hat{f} = \mathop{\mathrm{argmin}}_{f \in \mathcal{F}(L,W,S,B)} \sum_{i=1}^{n} (y_i - f(x_i))^2.$$

Bias and variance decomposition

A standard covering number argument gives

$$\mathrm{E}[\|f^{\mathrm{o}} - \hat{f}\|_{L^2(P_X)}^2] \lesssim \underbrace{\frac{S[L\log(BW) + \log(Ln)]}{n}}_{\text{Variance}} + \underbrace{\inf_{f \in \mathcal{F}(L,W,S,B)} \|f - f^{\mathrm{o}}\|_{L^2(P_X)}^2}_{\text{Bias}}$$

Bias and variance decomposition

A standard covering number argument gives

$$\mathrm{E}[\|f^{\mathrm{o}} - \hat{f}\|_{L^2(P_X)}^2] \lesssim \underbrace{\frac{S[L\log(BW) + \log(Ln)]}{n}}_{\text{Variance}} + \underbrace{\inf_{f \in \mathcal{F}(L,W,S,B)} \|f - f^{\mathrm{o}}\|_{L^2(P_X)}^2}_{\text{Bias}}$$

If $f^{\mathrm{o}} \in B_{p,q}^s(\Omega)$, we know that

$$\text{Bias} = N^{-s/d} \quad \text{(approximation error)}$$

for $L = O(\log(N)), W = O(N), S = O(N\log(N)), B = O(N^{(d/p-s)_+})$.
$\Rightarrow$ Balance the bias and variance terms.

Estimation error analysis

$$y_i = f^{\mathrm{o}}(x_i) + \xi_i \ (i = 1, \dots, n),$$

where $x_i \sim P(X)$ with density $p \in L^{r/(r-2)}([0,1]^d)$ for $r < (1/p - s/d)_+^{-1}$.

$\mathcal{F}(L, W, S, B)$: ReLU-NN with width W, depth L ans sparsity S with parameters are bounded by B.

$$\hat{f} = \operatorname*{argmin}_{f \in \mathcal{F}(L,W,S,B)} \sum_{i=1}^{n} (y_i - \bar{f}(x_i))^2$$

($\bar{f}$ is the *clipping* of f: $\bar{f} = \min\{\max\{f, -R\}, R\}$; realizable by ReLU)

Proposition

For f^{o} s.t. $\|f^{\mathrm{o}}\|_{B^s_{p,q}([0,1]^d)} \leq 1$ and $\|f^{\mathrm{o}}\|_\infty \leq R$, and $0 < p, q \leq \infty$ with $s > d(\frac{1}{p} - \frac{1}{2})_+$, by letting $N \asymp n^{\frac{d}{2s+d}}$,

$$\mathrm{E}[\|f^{\mathrm{o}} - \hat{f}\|^2_{L^2(P_X)}] \leq n^{-\frac{2s}{2s+d}} \log(n)^3.$$

Setting $p = q = \infty$, the result of Schmidt-Hieber (2017) is recovered as a special case.

Estimation error analysis

$$y_i = f^{\mathrm{o}}(x_i) + \xi_i \ (i = 1, \dots, n),$$

where $x_i \sim P(X)$ with density $p \in L^{r/(r-2)}([0,1]^d)$ for $r < (1/p - s/d)_+^{-1}$.

$\mathcal{F}(L, W, S, B)$: ReLU-NN with width W, depth L ans sparsity S with parameters are bounded by B.

$$\hat{f} = \mathop{\mathrm{argmin}}_{f \in \mathcal{F}(L,W,S,B)} \sum_{i=1}^{n} (y_i - \bar{f}(x_i))^2$$

($\bar{f}$ is the *clipping* of f: $\bar{f} = \min\{\max\{f, -R\}, R\}$; realizable by ReLU)

Proposition

For f^{o} s.t. $\|f^{\mathrm{o}}\|_{B^s_{p,q}([0,1]^d)} \leq 1$ and $\|f^{\mathrm{o}}\|_\infty \leq R$, and $0 < p, q \leq \infty$ with $s > d(\frac{1}{p} - \frac{1}{2})_+$, by letting $N \asymp n^{\frac{d}{2s+d}}$,

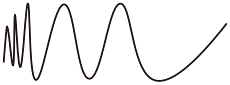
$$\mathrm{E}[\|f^{\mathrm{o}} - \hat{f}\|^2_{L^2(P_X)}] \leq n^{-\frac{2s}{2s+d}} \log(n)^3.$$

Minimax optimal rate.

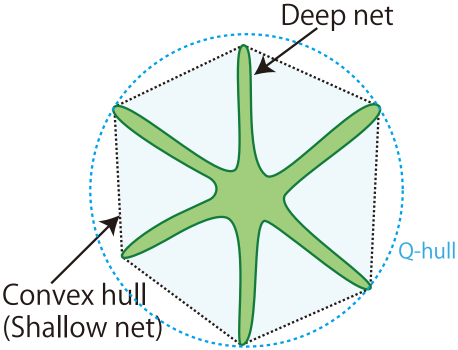
Best linear estimator vs. deep learning

- **Linear estimator** (Donoho & Johnstone, 1998; Zhang et al., 2002)
  $$\hat{f}(x) = \sum_{i=1}^{n} y_i \varphi(x_1, \ldots, x_n; x)$$
  Kernel ridge estimator, Sieve method, Nadaraya-Watson estimator, ...
  (e.g., $\hat{f}(x) = K_{x,X}(K_{X,X} + \lambda \mathrm{I})^{-1} Y$). For $s > 1/p$,
  $$n^{-\frac{2s - 2(1/p - 1/2)_+}{2s + 1 - 2(1/p - 1/2)_+}}$$
  $$\vee$$
- **Deep learning** (our bound)
  $$n^{-\frac{2s}{2s+1}}$$
  for $s > (1/p - 1/2)_+$.
  (sparse estimator achieves this rate for $s > \max\{1/p, 1/2\}$ (Donoho & Johnstone, 1998))

There appears difference when $p < 2$.
$p < 2$ **corresponds to spatial incoherence of smoothness.**



Why does this difference happen?



$$\inf_{\hat{f}:\text{Linear}} \sup_{f^{\mathrm{o}} \in \mathcal{F}} \mathrm{E}[\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2] = \inf_{\hat{f}:\text{Linear}} \sup_{f^{\mathrm{o}} \in \mathrm{conv}(\mathcal{F})} \mathrm{E}[\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2].$$

(More strictly, it can be extended to "Q-hull.")

Outline

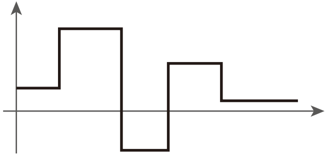
1. Literature overview

2. Approximating and estimating functions in Besov space and related spaces
   - Deep NN representation for Besov space
   - Function class with more explicit sparsity
   - Deep NN representation for "mixed smooth" Besov space

Functions with jumps

$$J_K = \left\{ a_0 + \sum_{i=1}^{K} \mathbf{1}_{[t_i,1]} \mid t_i \in (0,1], |a_0|, \sum_{i=1}^{K} |a_i| \leq 1 \right\}$$

→ Its convex hull includes the **functions of bounded variation**.



Theorem

$$\inf_{\hat{f}:\textbf{Linear}} \sup_{f^{\mathrm{o}} \in J_K} \mathrm{E}\left[\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2 \right] \geq \Omega\left(\frac{1}{\sqrt{n}} \right).$$

But, for a deep learning estimator $\hat{f}$, we obtain

$$\sup_{f^{\mathrm{o}} \in J_K} \mathrm{E}\left[\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2 \right] \leq O\left(\frac{1}{n} \log(n)^3 \right).$$

Function class with sparse parameter

- Weak ℓ^p-norm of the coefficient:

$$\|\alpha\|\|_{\mathrm{w}\ell^p} := \sup_{i \in \mathbb{Z}_+} i^{1/p} |\alpha|_{(i)}$$

where $|\alpha|_{(i)}$ is the i-th largest absolute value.

- Function class with sparse coefficient:

$$\mathcal{J}^p := \left\{ \sum_{(k,\ell)} \alpha_{k,\ell} \psi_{k,\ell} \;\middle|\; \|\alpha\|_{\mathrm{w}\ell^p} \leq C, \sum_{k>m} |\alpha_{k,\ell}|^2 \leq C 2^{-\beta m} \right\}$$

where $\psi_{k,\ell}(x) = 2^{k/2}\psi(2^k x - \ell)$. ψ could be Haar wavelet.

- Finite combination of $\mathcal{J}^p$:

$$\mathcal{K}_p := \left\{ \sum_{i=1}^{S} c_i f_i(A_i \cdot - b_i) \;\middle|\; |c_i|, |\det A_i|^{-1}, \|A_i\|_\infty, \|b_i\|_\infty \leq C, f_i \in \mathcal{J}^p \right\}.$$

Convergence rate of deep NN

Theorem

	Minimax rate	Deep learning
J_k	$\Omega(n^{-1})$	$O(n^{-1}\log(n)^3)$
$\mathcal{K}^p$	$\Omega(n^{-\frac{2\alpha}{2\alpha+1}}(\log(n))^{-\frac{4\alpha^2}{2\alpha+1}})$	$O\left(n^{-\frac{2\alpha}{2\alpha+1}}\log(n)^3\right)$

where $0 < p < 2,\ \alpha = 1/p - 1/2$.

- For $0 < p < 1$ (sparse situation), DL is better than the linear estimator:

$$n^{-1}\log(n)^3, n^{-\frac{2\alpha}{2\alpha+1}}\log(n)^3 \quad \ll \quad n^{-1/2}$$

Deep　　　　Shallow (Linear)

Outline

Difficulty

$$n^{-\frac{2s}{2s+d}}$$

d influences the exponent of the convergence rate.
$\rightarrow$ **Curse of dimensionality**

Relation to existing work

Besov space with dominating mixed smoothness (tensor product space)

$$MB^{\mathbf{r}}_{p,p} = B^{r_1}_{p,p} \otimes \cdots \otimes B^{r_d}_{p,p}$$

The estimation accuracy $\|\hat{f} - f^{\mathrm{o}}\|^2_{L_2(P)}$.

Space	Hölder ($\forall\beta$)	Barron class	m-Sobolev ($\beta \leq 2$)	m-Besov ($\forall\beta$)
Approximation				
	Yarotsky (2016), Liang and Srikant (2016)	Barron (1993)	Montanelli and Du (2017)	This work
Approx. rate	$\tilde{O}(m^{-\frac{\beta}{d}})$	$\tilde{O}(m^{-1/2})$	$\tilde{O}(m^{-\beta})$	$\tilde{O}(m^{-\beta})$
Estimation				
	Schmidt-Hieber (2017)	Barron (1993)	—	This work
Estimation. rate	$\tilde{O}(n^{-\frac{2\beta}{2\beta+d}})$	$\tilde{O}(n^{-\frac{1}{2}})$	—	$\tilde{O}(n^{-\frac{2\beta}{2\beta+1+\log_2(e)}})$

Tensor product space

Tensor product of Besov space (dominating mixed smoothness)

$$MB^{\beta}_{p,p} = B^{\beta}_{p,p}(\mathbb{R}) \otimes_p \cdots \otimes_p B^{\beta}_{p,p}(\mathbb{R})$$
$$f(x_1,\dots,x_d) \in \overline{\mathrm{span}\{f_1(x_1) \times \cdots \times f_d(x_d)\}}$$
$$(\lim_{R\to\infty} \textstyle\sum_{r=1}^{R} f_r^{(1)}(x_1) f_r^{(2)}(x_2) \dots f_r^{(d)}(x_d))$$

Can be extended to $p \neq q$ $MB^{\beta}_{p,q}$ (see, for example, Sickel and Ullrich (2009); Dũng (2011a)).

Tensor product space

Tensor product of Besov space (dominating mixed smoothness)

$$MB^{\beta}_{p,p} = B^{\beta}_{p,p}(\mathbb{R}) \otimes_p \cdots \otimes_p B^{\beta}_{p,p}(\mathbb{R})$$

$$f(x_1, \ldots, x_d) \in \overline{\mathrm{span}\{f_1(x_1) \times \cdots \times f_d(x_d)\}}$$

$$(\lim_{R\to\infty} \textstyle\sum_{r=1}^{R} f_r^{(1)}(x_1) f_r^{(2)}(x_2) \ldots f_r^{(d)}(x_d))$$

Can be extended to $p \neq q$ $MB^{\beta}_{p,q}$ (see, for example, Sickel and Ullrich (2009); Dũng (2011a)).

When $p \geq 1$, let the norm of the space $B^{\beta}_{p,p} \otimes_p \mathcal{G}$ for a Banach space $\mathcal{G}$ be

$$\|f\|_{B^{\boldsymbol{\beta}}_{p,p} \otimes_p \mathcal{G}} := \inf \left\{ \left(\sum_{r=1}^{R} \|f_r^{(1)}\|^p_{B^{\beta}_{p,p}} \right)^{1/p} \sup \left[\left\| \sum_{r=1}^{R} \lambda_r g_r^{(2)} \right\|_{\mathcal{G}} \;\middle|\; \left(\sum_{r=1}^{R} |\lambda_r|^p \right)^{1/p} \leq 1 \right] \right\}$$

for $f = \sum_{r=1}^{R} f_r^{(1)}(x_1) g_r^{(2)}(x_2)$ where $f_r^{(1)} \in B^{\beta}_{p,p}$ and $g_r^{(2)} \in \mathcal{G}$.

- $B^{\boldsymbol{\beta}}_{p,p} \otimes_p \mathcal{G}$ is obtained by completion of the finite sum w.r.t. this norm.
- $MB^{\boldsymbol{\beta}}_{p,p} := B^{\beta}_{p,p} \otimes_p (\cdots B^{\beta}_{p,p} \otimes_p (B^{\beta}_{p,p} \otimes_p B^{\beta}_{p,p}))$
- For $p < 1$ and $p = \infty$, a different norm is induced.
  (see Light and Cheney (1985))

Tensor product space

Tensor product of Besov space (dominating mixed smoothness)

$$MB^{\beta}_{p,p} = B^{\beta}_{p,p}(\mathbb{R}) \otimes_p \cdots \otimes_p B^{\beta}_{p,p}(\mathbb{R})$$
$$f(x_1, \dots, x_d) \in \overline{\mathrm{span}\{f_1(x_1) \times \cdots \times f_d(x_d)\}}$$
$$(\lim_{R\to\infty} \textstyle\sum_{r=1}^{R} f_r^{(1)}(x_1) f_r^{(2)}(x_2) \dots f_r^{(d)}(x_d))$$

Can be extended to $p \neq q$ $MB^{\beta}_{p,q}$ (see, for example, Sickel and Ullrich (2009); Dũng (2011a)).

- Tensor product of Besov ($MB^2_{p,q}(\mathbb{R}^2)$):

$$\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial^2 f}{\partial x_1^2}, \frac{\partial^2 f}{\partial x_2^2}, \frac{\partial^2 f}{\partial x_1 \partial x_2}, \frac{\partial^3 f}{\partial x_1 \partial x_2^2}, \frac{\partial^3 f}{\partial x_1^2 \partial x_2}, \frac{\partial^4 f}{\partial x_1^2 \partial x_2^2}$$

  (e.g., Korobov space)
- Sobolev ($W^2_p(\mathbb{R}^2)$):

$$\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial^2 f}{\partial x_1^2}, \frac{\partial^2 f}{\partial x_2^2}, \frac{\partial^2 f}{\partial x_1 \partial x_2}$$

Examples

$$f(g_1(x_1), g_2(x_2), \ldots, g_d(x_d))$$

$g_k \in B^s_{p,q}(\mathbb{R})$, f: sufficiently smooth.

- Additive model:

$$f(x) = \sum_{r=1}^{d} f_d(x_d)$$

- Tensor model:

$$f(x) = \sum_{r=1}^{R} \prod_{k=1}^{d} f_{r,k}(x_k)$$

Approximation by NN

Theorem

Suppose that $0 < p, q, r \le \infty$ *and* $\beta > (1/p - 1/r)_+$. *For all* $f \in MB^{\boldsymbol{\beta}}_{p,q}([0,1]^d)$ *s.t.* $\|f^{\mathrm{o}}\|_{MB^{\beta}_{p,q}([0,1]^d)} \le 1$ *and* $N \ge 1$, *there exists ReLU-NN* $\check{f}$ *with*

- *Width* $W = O(NC_{N,d})$
- *Depth* $L = O(\log(N))$
- *Sparsity* $S = O(W \times L \times \log(N))$

and the parameters are bounded by $\|W^{(\ell)}\|_\infty, \|b^{(\ell)}\|_\infty < O(N^{(1/p-\beta)_+})$ *such that*

$$\|f^{\mathrm{o}} - \check{f}\|_{L^r([0,1]^d)} \le \begin{cases} N^{-\beta} C_{d,N}^{(1/\min(r,1)-1/q)_+} & (p \ge r), \\ N^{-\beta} C_{d,N}^{(1/r-1/q)_+} & (p < r, r < \infty), \\ N^{-\beta} C_{d,N}^{(1-1/q)_+} & (r = \infty), \end{cases}$$

where $C_{d,N} := (1 + \frac{d-1}{\log(N)})^{\log(N)}(1 + \frac{\log(N)}{d-1})^{d-1}(\lesssim d^{\log(N)} \wedge \log(N)^{d-1})$.

- Ordinal Besov space $B^{\beta}_{p,q}([0,1]^d)$: $N^{-\beta/d}$.
- Proof idea: **Sparse grid** technique (Dũng, 2011a; Smolyak, 1963) combined with **adaptive nonlinear interpolation**.

Estimation error bound

$$y_i = f^{\mathrm{o}}(x_i) + \xi_i \ (i = 1, \ldots, n),$$

where $x_i \sim P(X)$ with density $p(x) < G$ on $[0, 1]^d$.

$\mathcal{F}(L, W, S, B)$: ReLU-NN with width W, depth L ans sparsity S with parameters are bounded by B.

$$\hat{f} = \underset{f \in \mathcal{F}(L,W,S,B)}{\mathrm{argmin}} \sum_{i=1}^{n} (y_i - \bar{f}(x_i))^2$$

($\bar{f}$ is the *clipping* of f: $\bar{f} = \min\{\max\{f, -R\}, R\}$; realizable by ReLU)

Theorem

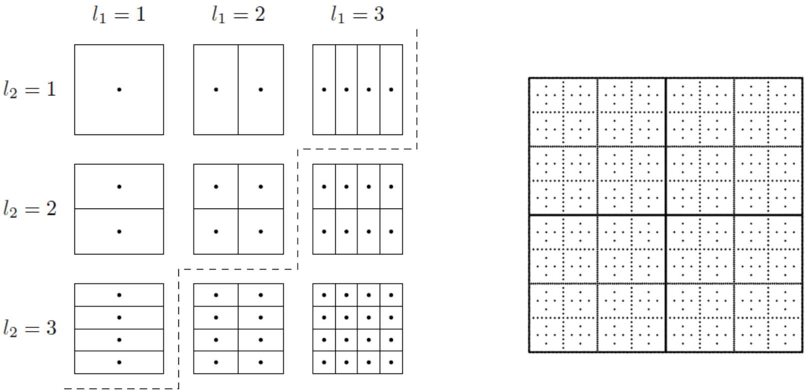
Suppose that $0 < p, q \leq \infty$ *and* $\beta > (1/p - 1/2)_+$. *For all* $f^{\mathrm{o}} \in MB^{\beta}_{p,q}([0,1]^d)$ *s.t.* $\|f^{\mathrm{o}}\|_{MB^{\beta}_{p,q}([0,1]^d)} \leq 1$, *by letting* $u = (1 - \frac{1}{q})_+ \ (p \geq 2),\ (\frac{1}{2} - \frac{1}{q})_+ \ (p < 2)$,

$$\|f^{\mathrm{o}} - \hat{f}\|^2_{L^2(P)} \leq \begin{cases} n^{-\frac{2\beta}{2\beta+1}} \log(n)^{\frac{2\beta+2u}{1+2\beta}(d-1)} \log(n)^3 & (\textit{every time}), \\ n^{-\frac{2\beta}{2\beta+1+\log_2(e)}} \log(n)^3 & (u = 0). \end{cases}$$

Besov space $B^{\beta}_{p,q}([0,1]^d)$: $\tilde{O}(n^{-\frac{2\beta}{2\beta+d}})$.

$\rightarrow$ **effect of dimensionality is eased.**

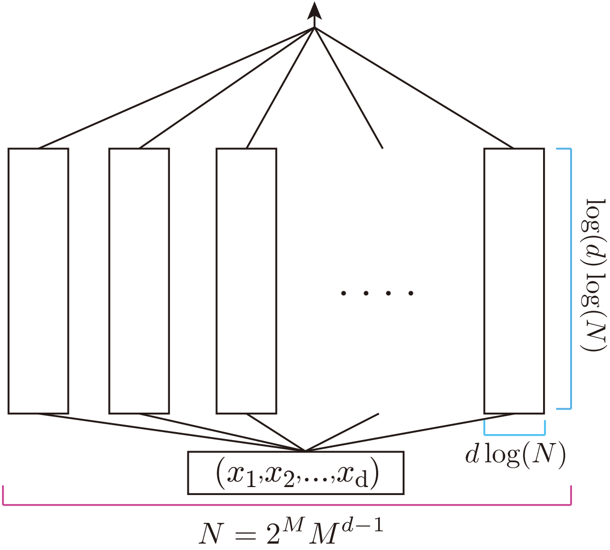
Sparse grid



(figure is borrowed from (Montanelli & Du, 2017))

Number of points in sparse grid: $N = 2^M M^{d-1}$.
Dense grid: $N = 2^{Md}$.

NN-structure



Applications

- Additive model:

$$f(x_1, \dots, x_d) = \sum_{j=1}^{d} f_r(x_r).$$

- Tensor product form:

$$f(x_1, \dots, x_d) = \sum_{r=1}^{R} \prod_{k=1}^{d} f_{r,k}(x_k).$$

- **Dimensionality reduction:**

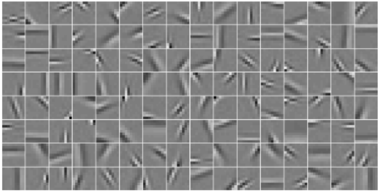
$$f^{\mathrm{o}} = g \circ F$$

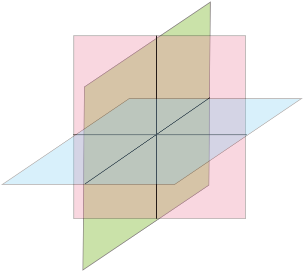
where $F : \mathbb{R}^d \to \mathbb{R}^D$ such that $D \ll d$ and $F_i \in MB^s_{p,q}$, and $g \in B^{\gamma}_{p,q}(\mathbb{R}^D)$:

$$\tilde{O}(n^{-\frac{2s}{2s+1+\log_2(e)}} + n^{-\frac{2\gamma}{2\gamma+D}}).$$

(F is a nonlinear dimensionality reduction into a low dimensional space (e.g., low dimensional manifold embedding).)
(see also Bölcskei et al. (2017))

Sparse input

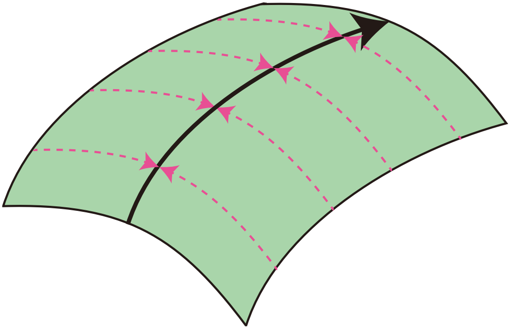




Input x is **sparse** (its number of non-zero elements is small).

$$\|x\|_0 \leq k \quad \Rightarrow \quad n^{-\frac{2\gamma}{2\gamma+k}}$$

Low dimensional manifold



$f(x)$ only depends on D-dimensional quotient-manifold:

$$n^{-\frac{2\gamma}{2\gamma+D}}$$

Conclusion

Adaptivity of deep learning

- It was shown that the ReLU-DNN has a high adaptivity to the shape of the target functions (discontinuity and spatial inhomogeneous smoothness).

$$\|\hat{f} - f^{\mathrm{o}}\|_{L_2(P)}^2 = \tilde{O}(n^{-2s/(2s+d)})$$

  - DNN outperforms a non-adaptive method.

$$\text{(DNN)} \quad n^{-2s/(2s+d)} \ll n^{-\frac{2(s-d(1/p-1/2))}{2s+d-2d(1/p-1/2)}} \quad \text{(linear method)}$$

- The ReLU-DNN can ease the *curse of dimensionality* to estimate the *mixed-smooth* Besov spaces.

$$\text{(Besov)}\ \tilde{O}(n^{-2s/(2s+d)}) \quad \to \quad \text{(m-Besov)}\ \tilde{O}(n^{-2s/(2s+1)} \log(n)^{\frac{2\beta+2u}{1+2\beta}(d-1)})$$

Better than fixed basis methods: high adaptivity to sparsity.

Arora, S., Ge, R., Neyshabur, B., & Zhang, Y. (2018). Stronger generalization bounds for deep nets via a compression approach. Proceedings of the 35th International Conference on Machine Learning (pp. 254–263). PMLR.

Barron, A. R. (1993). Universal approximation bounds for superpositions of a sigmoidal function. IEEE Transactions on Information theory, 39, 930–945.

Bianchini, M., & Scarselli, F. (2014). On the complexity of neural network classifiers: A comparison between shallow and deep architectures. IEEE transactions on neural networks and learning systems, 25, 1553–1565.

Bölcskei, H., Grohs, P., Kutyniok, G., & Petersen, P. (2017). Optimal approximation with sparsely connected deep neural networks. arXiv preprint arXiv:1705.01714.

Cohen, N., Sharir, O., & Shashua, A. (2016). On the expressive power of deep learning: A tensor analysis. The 29th Annual Conference on Learning Theory (pp. 698–728).

Cohen, N., & Shashua, A. (2016). Convolutional rectifier networks as generalized tensor decompositions. Proceedings of the 33th International Conference on Machine Learning (pp. 955–963).

DeVore, R. A., & Popov, V. A. (1988). Interpolation of besov spaces. Transactions of the American Mathematical Society, 305, 397–414.

Donoho, D. L., & Johnstone, I. M. (1998). Minimax estimation via wavelet shrinkage. The Annals of Statistics, 26, 879–921.

Dũng, D. (2011a). B-spline quasi-interpolant representations and sampling recovery of functions with mixed smoothness. Journal of Complexity, 27, 541–567.

Dũng, D. (2011b). Optimal adaptive sampling recovery. Advances in Computational Mathematics, 34, 1–41.

Eldan, R., & Shamir, O. (2016). The power of depth for feedforward neural networks. Proceedings of The 29th Annual Conference on Learning Theory (pp. 907–940).

Imaizumi, M., & Fukumizu, K. (2018). Deep neural networks learn non-smooth functions effectively. arXiv preprint arXiv:1802.04474.

Liang, S., & Srikant, R. (2016). Why deep neural networks for function approximation? arXiv preprint arXiv:1610.04161. ICLR2017.

Light, W., & Cheney, E. (1985). Approximation theory in tensor product spaces. Lecture notes in mathematics. Springer-Verlag.

Montanelli, H., & Du, Q. (2017). Deep relu networks lessen the curse of dimensionality. arXiv preprint arXiv:1712.08688.

Montufar, G. F., Pascanu, R., Cho, K., & Bengio, Y. (2014). On the number of linear regions of deep neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Weinberger (Eds.), Advances in neural information processing systems 27, 2924–2932. Curran Associates, Inc.

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., & Ganguli, S. (2016). Exponential expressivity in deep neural networks through transient chaos. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett (Eds.), Advances in neural information processing systems 29, 3360–3368. Curran Associates, Inc.

Schmidt-Hieber, J. (2017). Nonparametric regression using deep neural networks with ReLU activation function. ArXiv e-prints.

Sickel, W., & Ullrich, T. (2009). Tensor products of Sobolev–Besov spaces and applications to approximation from the hyperbolic cross. Journal of Approximation Theory, 161, 748–786.

Smolyak, S. (1963). Quadrature and interpolation formulas for tensor products of certain classes of functions. Soviet Math. Dokl. (pp. 240–243).

Sonoda, S., & Murata, N. (2015). Neural network with unbounded activation functions is universal approximator. Applied and Computational Harmonic Analysis.

Suzuki, T., Abe, H., Murata, T., Horiuchi, S., Ito, K., Wachi, T., Hirai, S., Yukishima, M., & Nishimura, T. (2018). Spectral-Pruning: Compressing deep neural network via spectral analysis. arXiv e-prints, arXiv:1808.08558.

Yarotsky, D. (2016). Error bounds for approximations with deep relu networks. CoRR, abs/1610.01145.

Zhang, S., Wong, M.-Y., & Zheng, Z. (2002). Wavelet threshold estimation of a regression function with random design. Journal of multivariate analysis, 80, 256–284.