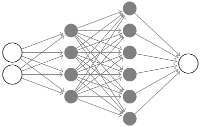
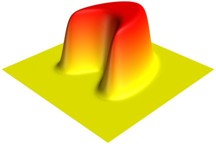
Learning ReLU networks to high uniform accuracy is intractable

Julius Berner
Department of Computing and Mathematical Sciences
California Institute of Technology

Caltech

$\Big\|$  $-$  $\Big\|_{L^\infty} \geq \varepsilon$

Collaborators

Philipp Grohs

University of Vienna, Austrian Academy of Sciences

Felix Voigtlaender

Catholic University of Eichstätt-Ingolstadt

universität wien

Katholische Universität Eichstätt-Ingolstadt

ÖAW RICAM

Motivation: Instabilities in Deep Learning

Undesired outputs of trained neural networks,
even for inputs within the training distribution.

Motivation: Instabilities in Deep Learning

Adversarial examples

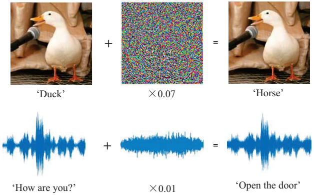
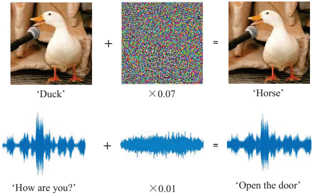


Fig. 1: Y. Gong and C. Poellabauer. Protecting voice controlled systems using sound source identification based on acoustic cues. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9. IEEE, 2018

Motivation: Instabilities in Deep Learning

Adversarial examples



Hallucinations

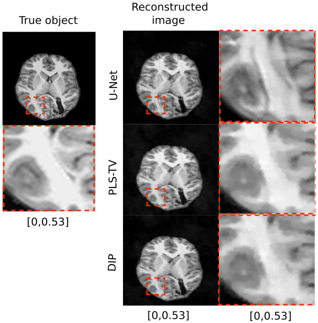
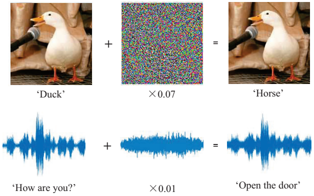


Fig. 2: S. Bhadra, V. A. Kelkar, F. J. Brooks, and M. A. Anastasio. On hallucinations in tomographic image reconstruction. *IEEE transactions on medical imaging*, 40(11):3249–3260, 2021

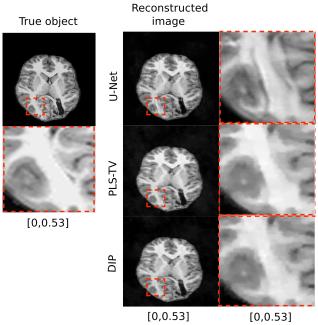
Fig. 1: Y. Gong and C. Poellabauer. Protecting voice controlled systems using sound source identification based on acoustic cues. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9. IEEE, 2018

Motivation: Instabilities in Deep Learning

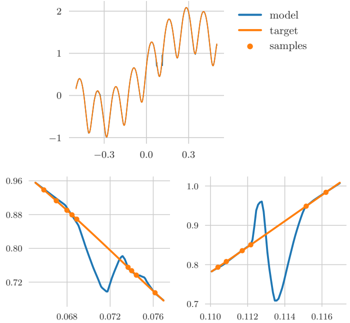
Adversarial examples



Hallucinations



Function approximation



See also: B. Adcock and N. Dexter. The gap between theory and practice in function approximation with deep neural networks. *SIAM Journal on Mathematics of Data Science*, 3(2):624–655, 2021

Fig. 2: S. Bhadra, V. A. Kelkar, F. J. Brooks, and M. A. Anastasio. On hallucinations in tomographic image reconstruction. *IEEE transactions on medical imaging*, 40(11):3249–3260, 2021

Fig. 1: Y. Gong and C. Poellabauer. Protecting voice controlled systems using sound source identification based on acoustic cues. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9. IEEE, 2018

Motivation: Instabilities in Deep Learning

Undesired outputs of trained neural networks,
even for inputs within the training distribution,
despite theoretical guarantees.

Theoretical Guarantees

Approximation

Bounds on the **number of parameters** of neural networks $\mathcal{N}$ to approximate function classes U in the sense of

$$\sup_{u \in U} \inf_{\phi \in \mathcal{N}} \|\phi - u\|_{L^\infty} \leq \varepsilon.$$

Theoretical Guarantees

Approximation

Bounds on the **number of parameters** of neural networks $\mathcal{N}$ to approximate function classes U in the sense of

$$\sup_{u \in U} \inf_{\phi \in \mathcal{N}} \|\phi - u\|_{L^\infty} \leq \varepsilon.$$

👍 Neural networks can optimally approximate many classical function classes (by emulating optimal dictionaries)!

See, e.g.: D. Elbrächter, D. Perekrestenko, P. Grohs, and H. Bölcskei. Deep neural network approximation theory. *IEEE Transactions on Information Theory*, 67(5):2581–2623, 2021

Theoretical Guarantees

Approximation

Bounds on the **number of parameters** of neural networks $\mathcal{N}$ to approximate function classes U in the sense of

$$\sup_{u \in U} \inf_{\phi \in \mathcal{N}} \|\phi - u\|_{L^\infty} \leq \varepsilon.$$

👍 Neural networks can optimally approximate many classical function classes (by emulating optimal dictionaries)!

Generalization

Bounds on the **number of samples** m required for the empirical risk minimizer

$$\hat{\phi} \in \arg\min_{\phi \in \mathcal{N}} \sum_{i=1}^m (\phi(x_i) - y_i)^2$$

to approximate the optimal neural network $\phi^* \in \mathcal{N}$, i.e.,

$$\|\hat{\phi} - \phi^*\|_{L^2} \leq \varepsilon.$$

See, e.g.: M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations.* Cambridge University Press, 1999

See, e.g.: D. Elbrächter, D. Perekrestenko, P. Grohs, and H. Bölcskei. Deep neural network approximation theory. *IEEE Transactions on Information Theory*, 67(5):2581–2623, 2021

Theoretical Guarantees

Approximation

Bounds on the **number of parameters** of neural networks $\mathcal{N}$ to approximate function classes U in the sense of

$$\sup_{u \in U} \inf_{\phi \in \mathcal{N}} \|\phi - u\|_{L^\infty} \leq \varepsilon.$$

👍 Neural networks can optimally approximate many classical function classes (by emulating optimal dictionaries)!

Generalization

Bounds on the **number of samples** m required for the empirical risk minimizer

$$\hat{\phi} \in \arg\min_{\phi \in \mathcal{N}} \sum_{i=1}^{m} (\phi(x_i) - y_i)^2$$

to approximate the optimal neural network $\phi^* \in \mathcal{N}$, i.e.,

$$\|\hat{\phi} - \phi^*\|_{L^2} \leq \varepsilon.$$

👍 Often scale only polynomially in the underlying dimension d!

See, e.g.: J. Berner, P. Grohs, and A. Jentzen. Analysis of the generalization error [. . .]. *SIAM Journal on Mathematics of Data Science*, 2(3):631–657, 2020

See, e.g.: M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999

See, e.g.: D. Elbrächter, D. Perekrestenko, P. Grohs, and H. Bölcskei. Deep neural network approximation theory. *IEEE Transactions on Information Theory*, 67(5):2581–2623, 2021

Theoretical Guarantees

Approximation

Bounds on the **number of parameters** of neural networks $\mathcal{N}$ to approximate function classes U in the sense of

$$\sup_{u \in U} \inf_{\phi \in \mathcal{N}} \|\phi - u\|_{L^\infty} \leq \varepsilon.$$

👍 Neural networks can optimally approximate many classical function classes (by emulating optimal dictionaries)!

Generalization

Bounds on the **number of samples** m required for the empirical risk minimizer

$$\hat{\phi} \in \arg\min_{\phi \in \mathcal{N}} \sum_{i=1}^{m} (\phi(x_i) - y_i)^2$$

to approximate the optimal neural network $\phi^* \in \mathcal{N}$, i.e.,

$$\|\hat{\phi} - \phi^*\|_{L^2} \leq \varepsilon.$$

👍 Often scale only polynomially in the underlying dimension d!

👎 Generalization results only provide **guarantees in an average sense** (w.r.t. the L^2-norm).

Example: Neural PDE Solver

For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

Example: Neural PDE Solver

For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

- Neural networks can approximate the solution up to error ε in the L^∞-norm,

Example: Neural PDE Solver

For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

- Neural networks can approximate the solution up to error ε in the L^∞-norm,
- Empirical risk minimizer approximates the solution up to error ε in the L^2-norm,

See, e.g.: D. Elbrächter, P. Grohs, A. Jentzen, and C. Schwab. Dnn expression rate analysis of high-dimensional pdes: Application to option pricing. *Constructive Approximation*, 55(1):3–71, 2022

Example: Neural PDE Solver

For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

- Neural networks can approximate the solution up to error ε in the L^∞-norm,
- Empirical risk minimizer approximates the solution up to error ε in the L^2-norm,

where the number of parameters and samples only **scale polynomially** in the **dimension** d and ε^{-1}.

See, e.g.: J. Berner, P. Grohs, and A. Jentzen. Analysis of the generalization error [. . .]. *SIAM Journal on Mathematics of Data Science*, 2(3):631–657, 2020

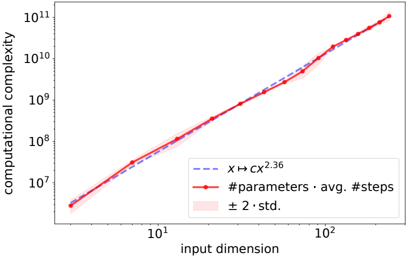
See, e.g.: D. Elbrächter, P. Grohs, A. Jentzen, and C. Schwab. Dnn expression rate analysis of high-dimensional pdes: Application to option pricing. *Constructive Approximation*, 55(1):3–71, 2022

Example: Neural PDE Solver

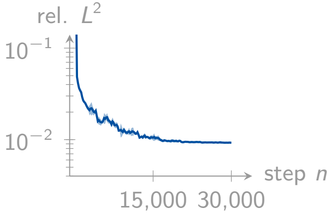
For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

- Neural networks can approximate the solution up to error ε in the L^∞-norm,
- Empirical risk minimizer approximates the solution up to error ε in the L^2-norm,

where the number of parameters and samples only **scale polynomially** in the **dimension** d and ε^{-1}.



Computational complexity to solve heat equations of varying dimensions up to a rel. L^1-error of 10^{-2}.



See, e.g.: J. Berner, P. Grohs, and A. Jentzen. Analysis of the generalization error [. . .]. *SIAM Journal on Mathematics of Data Science*, 2(3):631–657, 2020

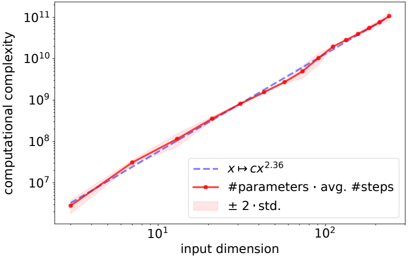
See, e.g.: D. Elbrächter, P. Grohs, A. Jentzen, and C. Schwab. Dnn expression rate analysis of high-dimensional pdes: Application to option pricing. *Constructive Approximation*, 55(1):3–71, 2022

Example: Neural PDE Solver

For the numerical solution of certain PDEs, deep learning **overcomes the curse of dimensionality**:

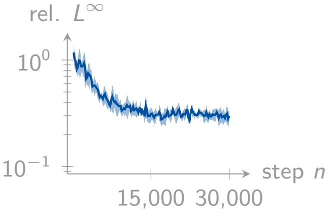
- Neural networks can approximate the solution up to error ε in the L^∞-norm,
- Empirical risk minimizer approximates the solution up to error ε in the L^2-norm,

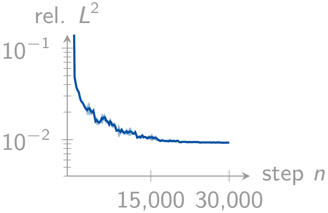
where the number of parameters and samples only **scale polynomially** in the **dimension** d and ε^{-1}.



Computational complexity to solve heat equations of varying dimensions up to a rel. L^1-error of 10^{-2}.

⚠ Only low uniform accuracies.





See, e.g.: J. Berner, P. Grohs, and A. Jentzen. Analysis of the generalization error [...]. *SIAM Journal on Mathematics of Data Science*, 2(3):631–657, 2020

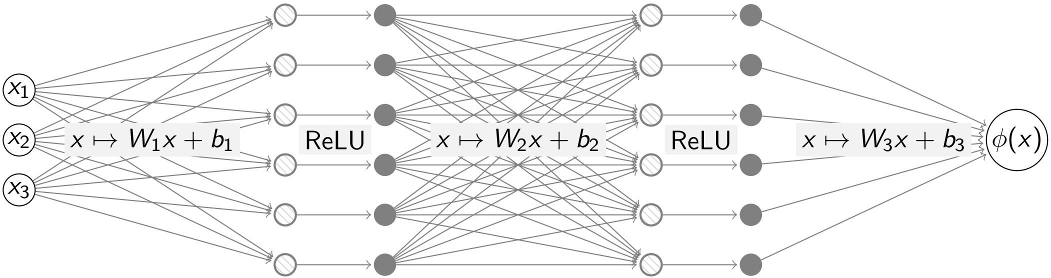
See, e.g.: D. Elbrächter, P. Grohs, A. Jentzen, and C. Schwab. Dnn expression rate analysis of high-dimensional pdes: Application to option pricing. *Constructive Approximation*, 55(1):3–71, 2022

Our results

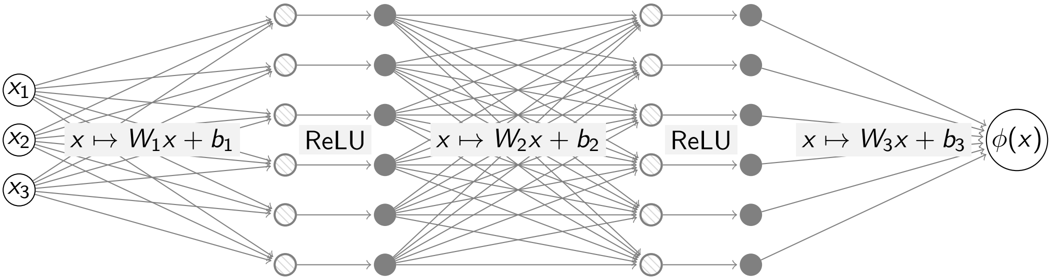
Undesired outputs of trained neural networks,
even for inputs within the training distribution,
despite theoretical guarantees.

Our results: Learning ReLU networks from samples with **uniform accuracy** (in the $\|\cdot\|_{L^\infty}$-norm) requires an **intractable number of samples**!

Setting: ReLU Networks



Setting: ReLU Networks



We consider sets $\mathcal{N} \subset C([0,1]^d)$ of feedforward networks with activation $\mathrm{ReLU}(x) = \max\{x, 0\}$, depth $L \in \mathbb{N}$, width $B \in \mathbb{N}$, and parameters $(W_\ell, b_\ell)_{\ell=1}^L$ with ℓ^q-regularization

$$\max_{1 \le \ell \le L} \max\{\|W_\ell\|_q, \|b_\ell\|_q\} \le c.$$

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

This includes:

✓ all variants of (S)GD

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

This includes:

- ✓ all variants of (S)GD
- ✓ adaptive algorithms (e.g., active learning),

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

This includes:

- ✓ all variants of (S)GD
- ✓ adaptive algorithms (e.g., active learning),
- ✓ randomized algorithms (e.g., MC algorithms),

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

This includes:

- ✓ all variants of (S)GD
- ✓ adaptive algorithms (e.g., active learning),
- ✓ randomized algorithms (e.g., MC algorithms),
- ✓ intractable algorithms (e.g., empirical risk minimization),

Setting: Learning Algorithms

We consider all learning algorithms $\mathcal{A}\colon U \to L^\infty([0,1]^d)$ that only operate on samples

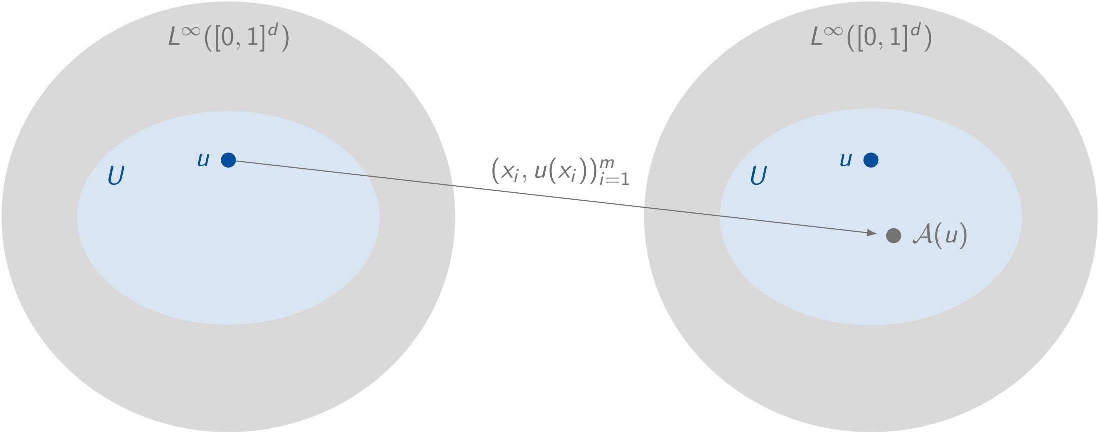
$$(x_i, u(x_i))_{i=1}^m$$

of functions $u \in U \subset C([0,1]^d)$.

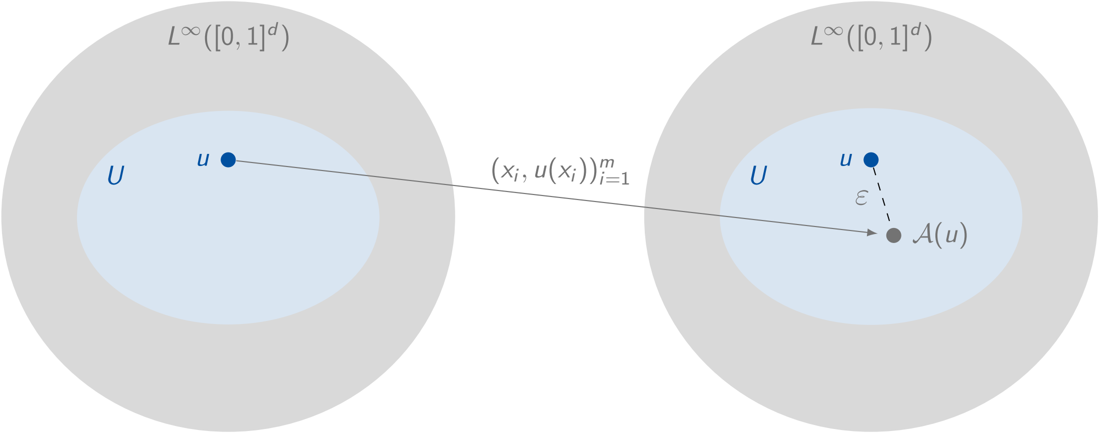
This includes:

- ✓ all variants of (S)GD
- ✓ adaptive algorithms (e.g., active learning),
- ✓ randomized algorithms (e.g., MC algorithms),
- ✓ intractable algorithms (e.g., empirical risk minimization),
- ✓ evaluations of a local operator instead of point samples (e.g., differential operator in the context of PINNs).

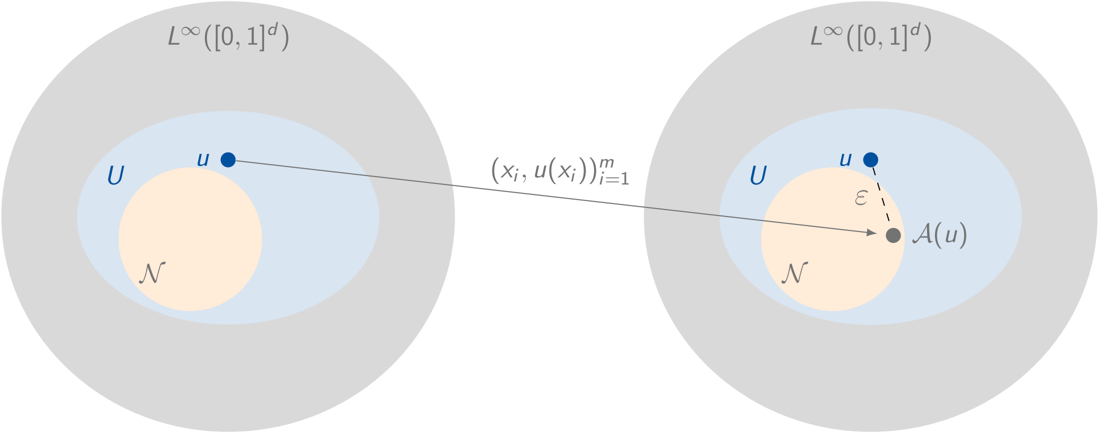
Setting: Learning Algorithms



Setting: Learning Algorithms



Setting: Learning Algorithms



Lower Bound

Lower Bound

Let $\mathcal{N} \subset U$ consist of ReLU networks with input dimension d, $L \geq 3$ layers, width $3d$, and parameters bounded by c. Any algorithm $\mathcal{A}$ satisfying $\sup_{u \in U} \mathbb{E}[\|\mathcal{A}(u) - u\|_{L^\infty}] \leq \varepsilon$ requires

$$m \geq c^{dL}(3d)^{d(L-2)} \left(\tfrac{1}{2^7 \varepsilon}\right)^d$$

samples on average.

Lower Bound

Lower Bound

Let $\mathcal{N} \subset U$ consist of ReLU networks with input dimension d, $L \geq 3$ layers, width $3d$, and parameters bounded by c. Any algorithm $\mathcal{A}$ satisfying $\sup_{u \in U} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^\infty}\right] \leq \varepsilon$ requires

$$m \geq c^{dL} (3d)^{d(L-2)} \left(\frac{1}{2^7 \varepsilon}\right)^d$$

samples on average.

Lower Bound

Lower Bound

Let $\mathcal{N} \subset U$ consist of ReLU networks with input dimension d, $L \geq 3$ layers, width $3d$, and parameters bounded by c. Any algorithm $\mathcal{A}$ satisfying $\sup_{u \in U} \mathbb{E}[\|\mathcal{A}(u) - u\|_{L^\infty}] \leq \varepsilon$ requires

$$m \geq c^{dL} (3d)^{d(L-2)} \left(\frac{1}{2^7 \varepsilon}\right)^{d}$$

samples on average.

👎 Number of samples m required to achieve high uniform accuracy ε scales **exponentially** with the **underlying dimension** d and the **depth** L of the ReLU networks $\mathcal{N}$. For instance, for $d = 15$, $c = 2$, $L = 7$, and $\varepsilon = \frac{1}{256}$, the sample size m exceeds the estimated number of atoms in the universe.

Lower Bound

Lower Bound

Let $\mathcal{N} \subset U$ consist of ReLU networks with input dimension d, $L \geq 3$ layers, width $3d$, and parameters bounded by c. Any algorithm $\mathcal{A}$ satisfying $\sup_{u \in U} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^\infty}\right] \leq \varepsilon$ requires

$$m \geq c^{dL} (3d)^{d(L-2)} \left(\frac{1}{2^7 \varepsilon}\right)^{d}$$

samples on average.

- 👎 Number of samples m required to achieve high uniform accuracy ε scales **exponentially** with the **underlying dimension** d and the **depth** L of the ReLU networks $\mathcal{N}$. For instance, for $d = 15$, $c = 2$, $L = 7$, and $\varepsilon = \frac{1}{256}$, the sample size m exceeds the estimated number of atoms in the universe.
- ⚠ Different from other hypothesis sets (e.g., polynomials or certain RKHS), m can **significantly exceed the number of parameters** defining the class $\mathcal{N}$.

Lower Bound: Proof Sketch

Proof Idea: Construction of **localized bumps** with regularized ReLU networks.

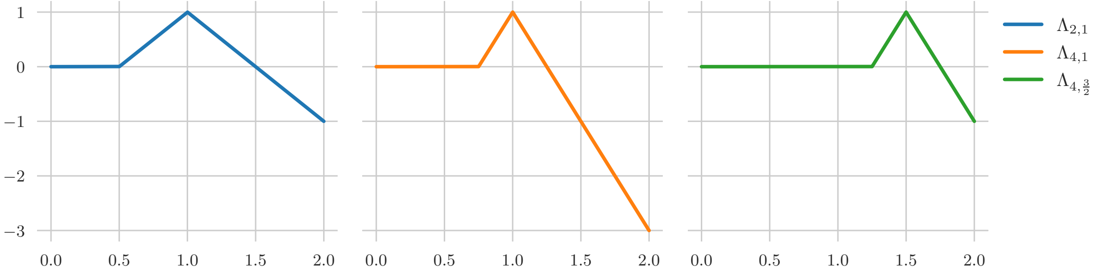
Lower Bound: Proof Sketch

Proof Idea: Construction of **localized bumps** with regularized ReLU networks.

Let us define

$$f_y(x) = \mathrm{ReLU}\left(1 - d + \textstyle\sum_{i=1}^{d} \Lambda_{y_i}(x_i)\right),$$

where Λ_{y_i} are hat functions with $\Lambda_{y_i}(y_i) = 1$
and $\Lambda_{y_i} \leq 0$ outside of $y_i + [-\frac{1}{M}, \frac{1}{M}]$.



Lower Bound: Proof Sketch

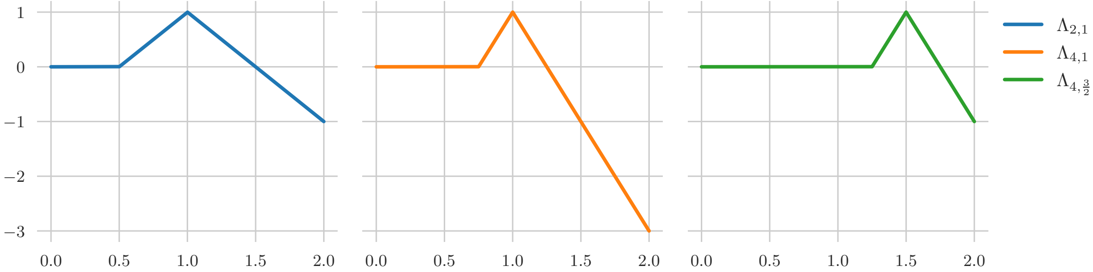
Proof Idea: Construction of **localized bumps** with regularized ReLU networks.

Let us define

$$f_y(x) = \mathrm{ReLU}\left(1 - d + \textstyle\sum_{i=1}^d \Lambda_{y_i}(x_i)\right),$$

where Λ_{y_i} are hat functions with $\Lambda_{y_i}(y_i) = 1$ and $\Lambda_{y_i} \le 0$ outside of $y_i + [-\frac{1}{M}, \frac{1}{M}]$.

1. f_y is supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$.



Lower Bound: Proof Sketch

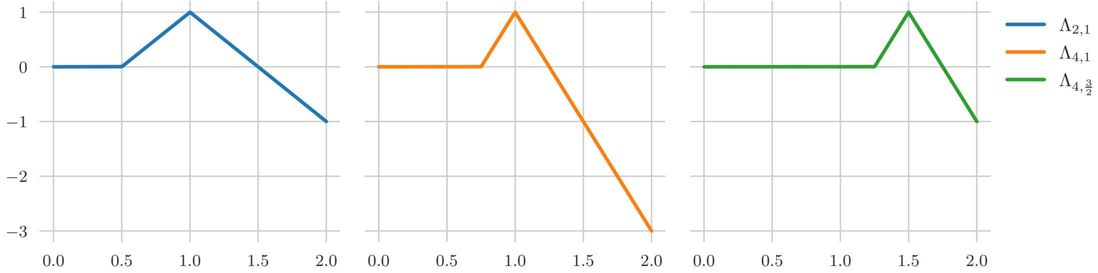
Proof Idea: Construction of **localized bumps** with regularized ReLU networks.

Let us define

$$f_y(x) = \mathrm{ReLU}\left(1 - d + \textstyle\sum_{i=1}^d \Lambda_{y_i}(x_i)\right),$$

where Λ_{y_i} are hat functions with $\Lambda_{y_i}(y_i) = 1$ and $\Lambda_{y_i} \leq 0$ outside of $y_i + [-\frac{1}{M}, \frac{1}{M}]$.

1. f_y is supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$.
2. It holds that $\|f_y\|_{L^p([0,1]^d)} \asymp M^{-d/p}$.



Lower Bound: Proof Sketch

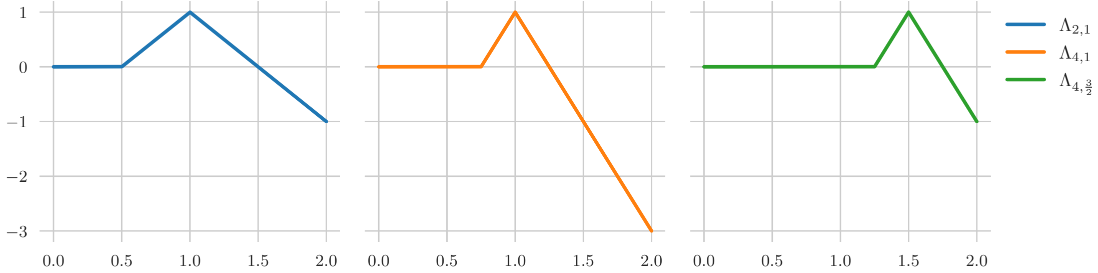
Proof Idea: Construction of **localized bumps** with regularized ReLU networks.

Let us define

$$f_y(x) = \mathrm{ReLU}\left(1 - d + \sum_{i=1}^d \Lambda_{y_i}(x_i)\right),$$

where Λ_{y_i} are hat functions with $\Lambda_{y_i}(y_i) = 1$ and $\Lambda_{y_i} \leq 0$ outside of $y_i + [-\frac{1}{M}, \frac{1}{M}]$.

1. f_y is supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$.
2. It holds that $\|f_y\|_{L^p([0,1]^d)} \asymp M^{-d/p}$.
3. f_y can be represented by a ReLU network with depth $L \geq 3$.

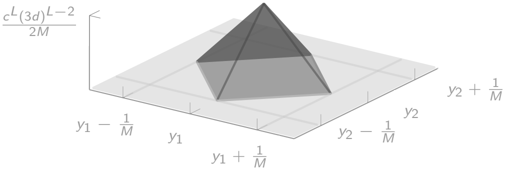


Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

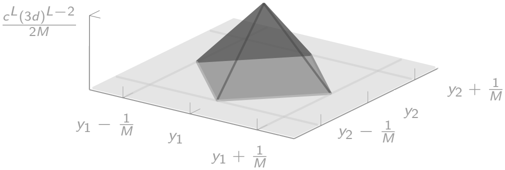


1. The previous construction yields ReLU networks $\phi_y \in \mathcal{N}$ supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$ with

$$\|\phi_y\|_{L^\infty} \geq c^L(3d)^{L-2}(2M)^{-1}.$$

Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

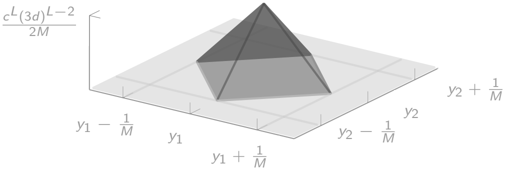


Bound: $m \geq c^{dL}(3d)^{d(L-2)} \left(\frac{1}{2^7 \varepsilon}\right)^d$

1. The previous construction yields ReLU networks $\phi_y \in \mathcal{N}$ supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$ with
$$\|\phi_y\|_{L^\infty} \geq c^L(3d)^{L-2}(2M)^{-1}.$$
2. Any $\mathcal{A}$ using m samples *on average*, will use at most $2m$ samples with probability at least $\frac{1}{2}$.

Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

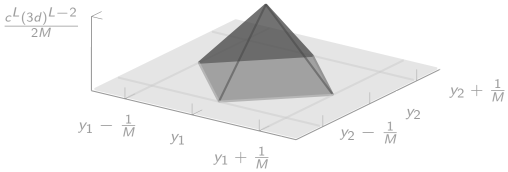


Bound: $m \geq c^{dL}(3d)^{d(L-2)} \left(\frac{1}{2^7\varepsilon}\right)^d$

1. The previous construction yields ReLU networks $\phi_y \in \mathcal{N}$ supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$ with
$$\|\phi_y\|_{L^\infty} \geq c^L(3d)^{L-2}(2M)^{-1}.$$
2. Any $\mathcal{A}$ using m samples *on average*, will use at most $2m$ samples with probability at least $\frac{1}{2}$.
3. Set $M = 8\lceil m^{1/d} \rceil$ and define $(y_\ell)_\ell$ as the nodes of a uniform grid on $[0,1]^d$ with width $2/M$.

Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

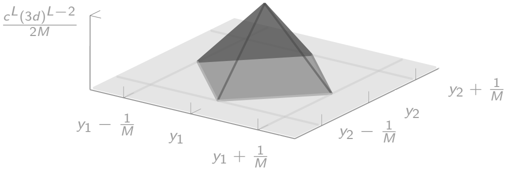


Bound: $m \geq c^{dL}(3d)^{d(L-2)} \left(\frac{1}{2^7\varepsilon}\right)^d$

1. The previous construction yields ReLU networks $\phi_y \in \mathcal{N}$ supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$ with
$$\|\phi_y\|_{L^\infty} \geq c^L(3d)^{L-2}(2M)^{-1}.$$
2. Any $\mathcal{A}$ using m samples *on average*, will use at most $2m$ samples with probability at least $\frac{1}{2}$.
3. Set $M = 8\lceil m^{1/d} \rceil$ and define $(y_\ell)_\ell$ as the nodes of a uniform grid on $[0,1]^d$ with width $2/M$.
4. Given sampling points $(x_i)_{i=1}^{2m}$, at least half of the indices ℓ satisfy $\phi_{y_\ell}(x_i) = 0$ for *all* i, i.e., $\mathcal{A}(\pm\phi_{y_\ell}) = \mathcal{A}(0)$.

Lower Bound: Proof Sketch

Let us focus on $p = \infty$ and ReLU networks in $\mathcal{N} \subset U$, i.e., with width $3d$ and parameters bounded by c.

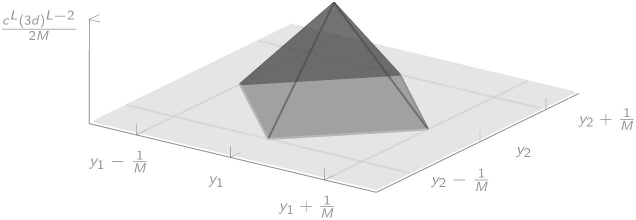


Bound: $m \geq c^{dL}(3d)^{d(L-2)} \left(\frac{1}{2^7\varepsilon}\right)^d$

1. The previous construction yields ReLU networks $\phi_y \in \mathcal{N}$ supported on $y + [-\frac{1}{M}, \frac{1}{M}]^d$ with
$$\|\phi_y\|_{L^\infty} \geq c^L(3d)^{L-2}(2M)^{-1}.$$
2. Any $\mathcal{A}$ using m samples *on average*, will use at most $2m$ samples with probability at least $\frac{1}{2}$.
3. Set $M = 8\lceil m^{1/d} \rceil$ and define $(y_\ell)_\ell$ as the nodes of a uniform grid on $[0,1]^d$ with width $2/M$.
4. Given sampling points $(x_i)_{i=1}^{2m}$, at least half of the indices ℓ satisfy $\phi_{y_\ell}(x_i) = 0$ for *all* i, i.e., $\mathcal{A}(\pm\phi_{y_\ell}) = \mathcal{A}(0)$.
5. Any $\mathcal{A}$ will thus make an expected error of $\varepsilon = \|\phi_y\|_{L^\infty}/4$ on average w.r.t. ℓ.

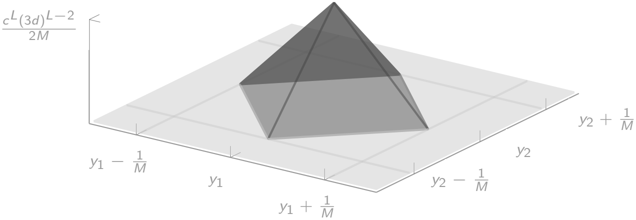
Lower Bound: Theory vs. Practice

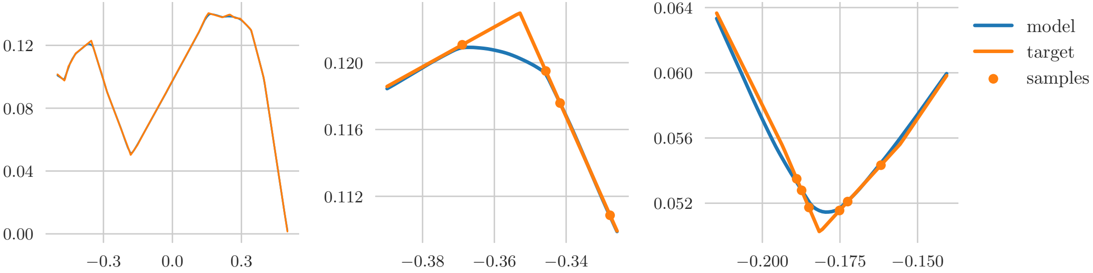
Similar bumps prevent high uniform accuracies in teacher-student settings.



Lower Bound: Theory vs. Practice

Similar bumps prevent high uniform accuracies in teacher-student settings.





Upper Bound

Upper Bound

There exists an algorithm $\mathcal{A}$ that satisfies $\sup_{u \in \mathcal{N}} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^\infty}\right] \leq \varepsilon$ using

$$m \leq c^{dL} (3d)^{d(L-2)} \left(\frac{3d^2}{\varepsilon} \right)^d$$

samples.

Upper Bound

Upper Bound

There exists an algorithm $\mathcal{A}$ that satisfies $\sup_{u \in \mathcal{N}} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^\infty}\right] \leq \varepsilon$ using

$$m \leq c^{dL}(3d)^{d(L-2)} \left(\frac{3d^2}{\varepsilon}\right)^d$$

samples.

Recall the lower bound: $m \geq c^{dL}(3d)^{d(L-2)} \left(\frac{1}{2^7 \varepsilon}\right)^d$

Upper Bound

Upper Bound

There exists an algorithm $\mathcal{A}$ that satisfies $\sup_{u\in\mathcal{N}} \mathbb{E}\left[\|\mathcal{A}(u)-u\|_{L^\infty}\right] \leq \varepsilon$ using

$$m \leq c^{dL}(3d)^{d(L-2)}\left(\frac{3d^2}{\varepsilon}\right)^d$$

samples.

Recall the lower bound: $m \geq c^{dL}(3d)^{d(L-2)}\left(\frac{1}{2^7\varepsilon}\right)^d$

Our bounds are **asymptotically sharp**.

Upper Bound: Proof Sketch

ReLU networks $\mathcal{N}$: input dimension d, L layers, width B, and parameters bounded by c

1. Induction over the number of layers L shows that

$$\mathrm{Lip}_{\ell^2}(\phi) \leq c^L \cdot \sqrt{d} \cdot B^{L-1}$$

for every $\phi \in \mathcal{N}$.

Upper Bound: Proof Sketch

ReLU networks $\mathcal{N}$: input dimension d, L layers, width B, and parameters bounded by c

1. Induction over the number of layers L shows that

$$\mathrm{Lip}_{\ell^2}(\phi) \leq c^L \cdot \sqrt{d} \cdot B^{L-1}$$

   for every $\phi \in \mathcal{N}$.

2. A piecewise constant interpolation algorithm $\mathcal{A}^*$ achieves that

$$\varepsilon = \left\| \mathcal{A}^*(u) - u \right\|_{L^\infty} \leq \mathrm{Lip}_{\ell^2}(u) \cdot \sqrt{d} \cdot m^{-1/d}.$$

Upper Bound: Proof Sketch

ReLU networks $\mathcal{N}$: input dimension d, L layers, width B, and parameters bounded by c

1. Induction over the number of layers L shows that

$$\mathrm{Lip}_{\ell^2}(\phi) \leq c^L \cdot \sqrt{d} \cdot B^{L-1}$$

for every $\phi \in \mathcal{N}$.

2. A piecewise constant interpolation algorithm $\mathcal{A}^*$ achieves that

$$\varepsilon = \left\| \mathcal{A}^*(u) - u \right\|_{L^\infty} \leq \mathrm{Lip}_{\ell^2}(u) \cdot \sqrt{d} \cdot m^{-1/d}.$$

3. Upper bound (for $B = 3d$): $m \leq c^{dL}(3d)^{d(L-2)} \left(\frac{3d^2}{\varepsilon} \right)^d$

General Lower Bound

Fully explicit bounds w.r.t. L^p-norm, architecture, and ℓ^q-norm on the parameters.

General Lower Bound

Fully explicit bounds w.r.t. L^p-norm, architecture, and ℓ^q-norm on the parameters.

General Lower Bound

Let $p, q \in [1, \infty]$. Assume that $\mathcal{N} \subset U$, where $\mathcal{N}$ is the set of ReLU networks with input dimension d, $L \geq 3$ layers of width B and parameters bounded by c in the ℓ^q-norm. Then, for any algorithm $\mathcal{A}$ and $s \leq \min\left\{\frac{B}{3}, d\right\}$, we have

$$\sup_{u \in U} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^p}\right] \geq \Omega \cdot (32s)^{-1-\frac{s}{p}} \cdot m^{-\frac{1}{p}-\frac{1}{s}},$$

where $\Omega = \frac{1}{8 \cdot 3^{2/q}} \cdot c^L \cdot s^{1-\frac{2}{q}}$ if $q < 2$ and $\Omega = \frac{1}{48} \cdot c^L \cdot B^{(L-1)(1-\frac{2}{q})}$ else.

General Lower Bound

Fully explicit bounds w.r.t. L^p-norm, architecture, and ℓ^q-norm on the parameters.

General Lower Bound

Let $p, q \in [1, \infty]$. Assume that $\mathcal{N} \subset U$, where $\mathcal{N}$ is the set of ReLU networks with input dimension d, $L \geq 3$ layers of width B and parameters bounded by c in the ℓ^q-norm. Then, for any algorithm $\mathcal{A}$ and $s \leq \min\left\{\frac{B}{3}, d\right\}$, we have

$$\sup_{u \in U} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^p}\right] \geq \Omega \cdot (32s)^{-1-\frac{s}{p}} \cdot m^{-\frac{1}{p}-\frac{1}{s}},$$

where $\Omega = \frac{1}{8 \cdot 3^{2/q}} \cdot c^L \cdot s^{1-\frac{2}{q}}$ if $q < 2$ and $\Omega = \frac{1}{48} \cdot c^L \cdot B^{(L-1)(1-\frac{2}{q})}$ else.

⚠ Strong regularizer (small q): exponential scaling is only visible for smaller ε.

General Lower Bound

Fully explicit bounds w.r.t. L^p-norm, architecture, and ℓ^q-norm on the parameters.

General Lower Bound

Let $p, q \in [1, \infty]$. Assume that $\mathcal{N} \subset U$, where $\mathcal{N}$ is the set of ReLU networks with input dimension d, $L \geq 3$ layers of width B and parameters bounded by c in the ℓ^q-norm. Then, for any algorithm $\mathcal{A}$ and $s \leq \min\left\{\frac{B}{3}, d\right\}$, we have

$$\sup_{u \in U} \mathbb{E}\left[\|\mathcal{A}(u) - u\|_{L^p}\right] \geq \Omega \cdot (32s)^{-1-\frac{s}{p}} \cdot m^{-\frac{1}{p}-\frac{1}{s}},$$

where $\Omega = \frac{1}{8 \cdot 3^{2/q}} \cdot c^L \cdot s^{1-\frac{2}{q}}$ if $q < 2$ and $\Omega = \frac{1}{48} \cdot c^L \cdot B^{(L-1)(1-\frac{2}{q})}$ else.

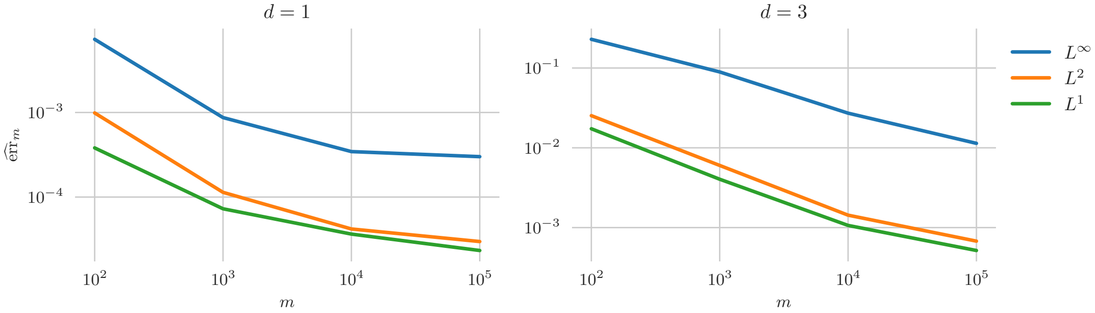
⚠ Strong regularizer (small q): exponential scaling is only visible for smaller ε.

⚠ $p \ll \infty$: tractable bounds in line with statistical learning theory and ε-entropy numbers scaling linearly in the depth L and the number of parameters, and logarithmically in ε^{-1}.

Experiments

Theoretical results are **validated in student-teacher settings**.

✓ Gap between uniform and average errors:



Min-max error over various ReLU networks (students), each trained using Adam on m samples from 40 teacher networks with $B = 32$, $L = 5$, and uniform weights in $[-0.5, 0.5]$.

Connections

Statistical learning theory

Connections

Statistical learning theory

- Bounded hypothesis classes with finite pseudodimension D can be learned to squared L^2-loss ε with $\mathcal{O}(D\varepsilon^{-2})$ samples.

P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *The Journal of Machine Learning Research*, 20(1):2285–2301, 2019

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2018

M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999

V. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999

Connections

Statistical learning theory

- Bounded hypothesis classes with finite pseudodimension D can be learned to squared L^2-loss ε with $\mathcal{O}(D\varepsilon^{-2})$ samples.
- We consider "realizable case" of PAC learning, where the target function is contained in the hypothesis class.

P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *The Journal of Machine Learning Research*, 20(1):2285–2301, 2019

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2018

M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999

V. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999

Connections

Statistical learning theory

- Bounded hypothesis classes with finite pseudodimension D can be learned to squared L^2-loss ε with $\mathcal{O}(D\varepsilon^{-2})$ samples.
- We consider "realizable case" of PAC learning, where the target function is contained in the hypothesis class.
- We provide fully explicit upper and lower bounds on the sample complexity of regularized neural network hypothesis classes (without IID assumption; for all L^p-norms).

P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *The Journal of Machine Learning Research*, 20(1):2285–2301, 2019

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2018

M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999

V. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999

Connections

Neural network identification

Connections

Neural network identification

- Robust identification of the parameters from samples is sufficient to guarantee uniform accuracy.

D. Rolnick and K. Kording. Reverse-engineering deep relu networks. In *International Conference on Machine Learning*, pages 8178–8187, 2020

C. Fiedler, M. Fornasier, T. Klock, and M. Rauchensteiner. Stable recovery of entangled weights: Towards robust identification of deep neural networks from minimal samples. *Applied and Computational Harmonic Analysis*, 62:123–172, 2023

Connections

Neural network identification

- Robust identification of the parameters from samples is sufficient to guarantee uniform accuracy.
- It is not a necessary condition, since proximity of input-output maps does not imply proximity of the neural network parameters.

J. Berner, D. M. Elbrächter, and P. Grohs. How degenerate is the parametrization of neural networks with the relu activation function? *Advances in Neural Information Processing Systems*, 32, 2019

D. Rolnick and K. Kording. Reverse-engineering deep relu networks. In *International Conference on Machine Learning*, pages 8178–8187, 2020

C. Fiedler, M. Fornasier, T. Klock, and M. Rauchensteiner. Stable recovery of entangled weights: Towards robust identification of deep neural networks from minimal samples. *Applied and Computational Harmonic Analysis*, 62:123–172, 2023

Connections

Neural network identification

- Robust identification of the parameters from samples is sufficient to guarantee uniform accuracy.
- It is not a necessary condition, since proximity of input-output maps does not imply proximity of the neural network parameters.
- Our results show that efficient identification from samples requires further prior information (as is done in related works).

J. Berner, D. M. Elbrächter, and P. Grohs. How degenerate is the parametrization of neural networks with the relu activation function? *Advances in Neural Information Processing Systems*, 32, 2019

D. Rolnick and K. Kording. Reverse-engineering deep relu networks. In *International Conference on Machine Learning*, pages 8178–8187, 2020

C. Fiedler, M. Fornasier, T. Klock, and M. Rauchensteiner. Stable recovery of entangled weights: Towards robust identification of deep neural networks from minimal samples. *Applied and Computational Harmonic Analysis*, 62:123–172, 2023

Connections

Hardness results on training and runtime

Connections

Hardness results on training and runtime

- For certain architectures neural network training is known to be NP-complete.

A. L. Blum and R. L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1):117–127, 1992

V. Vu. On the infeasibility of training neural networks with small mean-squared error. *IEEE Transactions on Information Theory*, 44(7):2892–2900, 1998

Connections

Hardness results on training and runtime

- For certain architectures neural network training is known to be NP-complete.
- Statistical query (SQ) algorithms: ReLU neural networks with constant depth and polynomial size constraints are not efficiently learnable up to a small squared loss with respect to a Gaussian distribution (in terms of runtime; contingent on difficult and unproven conjectures from cryptography)

S. Chen, A. Gollakota, A. R. Klivans, and R. Meka. Hardness of noise-free learning for two-hidden-layer neural networks. *arXiv preprint arXiv:2202.05258*, 2022

A. L. Blum and R. L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1):117–127, 1992

V. Vu. On the infeasibility of training neural networks with small mean-squared error. *IEEE Transactions on Information Theory*, 44(7):2892–2900, 1998

Connections

Hardness results on training and runtime

- For certain architectures neural network training is known to be NP-complete.
- Statistical query (SQ) algorithms: ReLU neural networks with constant depth and polynomial size constraints are not efficiently learnable up to a small squared loss with respect to a Gaussian distribution (in terms of runtime; contingent on difficult and unproven conjectures from cryptography)
- We show that the considered problem is information-theoretically hard, not just computationally (even if it were possible to efficiently learn a neural network from samples, the necessary number of data points would be intractable).

S. Chen, A. Gollakota, A. R. Klivans, and R. Meka. Hardness of noise-free learning for two-hidden-layer neural networks. *arXiv preprint arXiv:2202.05258*, 2022

A. L. Blum and R. L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1):117–127, 1992

V. Vu. On the infeasibility of training neural networks with small mean-squared error. *IEEE Transactions on Information Theory*, 44(7):2892–2900, 1998

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- Asymptotically matching **upper bounds**.

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- Asymptotically matching **upper bounds**.
- **Empirical validation** of our results in teacher-student settings.

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- Asymptotically matching **upper bounds**.
- **Empirical validation** of our results in teacher-student settings.
- Connections to **statistical learning theory**, **statistical query algorithms** and **neural network identification**.

Summary and Open Questions

- 🚀 **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- 🚀 In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- 🚀 Asymptotically matching **upper bounds**.
- 🚀 **Empirical validation** of our results in teacher-student settings.
- 🚀 Connections to **statistical learning theory**, **statistical query algorithms** and **neural network identification**.

? Overcome lower bounds by **incorporating additional information** about u into the learning problem.

Summary and Open Questions

- **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- Asymptotically matching **upper bounds**.
- **Empirical validation** of our results in teacher-student settings.
- Connections to **statistical learning theory**, **statistical query algorithms** and **neural network identification**.

? Overcome lower bounds by **incorporating additional information** about u into the learning problem.

? **Worst-case analysis**: for any $\mathcal{A}$ there exists at least one u on which $\mathcal{A}$ performs poorly. Is this poor behavior is actually generic?

Summary and Open Questions

- 🚀 **General lower bounds** for learning ReLU networks w.r.t. different L^p-norms and parameter regularizations.
- 🚀 In typical settings, learning ReLU networks needs an **intractable number of samples** for $p \gg 1$.
- 🚀 Asymptotically matching **upper bounds**.
- 🚀 **Empirical validation** of our results in teacher-student settings.
- 🚀 Connections to **statistical learning theory**, **statistical query algorithms** and **neural network identification**.

- ? Overcome lower bounds by **incorporating additional information** about u into the learning problem.
- ? **Worst-case analysis**: for any $\mathcal{A}$ there exists at least one u on which $\mathcal{A}$ performs poorly. Is this poor behavior is actually generic?
- ? Extension to **other architectures** and **activation functions**.

Thank you for your attention!

arxiv.org/abs/2205.13531
github.com/juliusberner/theory2practice
mail@jberner.info