Universität
Münster

Explicit Construction of Deep Neural Networks

Fabian Bremer

wissen.leben

Content

Universität Münster

Definition (DNN)

A **Deep Neural Network (DNN)** is a repeated concatenation of affine mappings and a specific non-linear map, called activation function $\sigma : \mathbb{R} \to \mathbb{R}$ of the DNN. Define $\sigma(x) = \max\{0, x\}, x \in \mathbb{R}$ as the rectified linear unit (ReLU) activation function. The DNN consists of a fixed number of hidden layers $L \in \mathbb{N}_0$ and numbers $N_\ell \in \mathbb{N}$ of so called *computation nodes* in layer $\ell \in \{1, \ldots, L+1\}$. N_0 is the input dimension of the DNN and N_{L+1} is the output dimension. The map $\Phi : \mathbb{R}^{N_0} \to \mathbb{R}^{N_{L+1}}$ is said to be realized by (and will therefore be called) a *feedforward* neural network, if for certain weights $A^\ell_{i,j} \in \mathbb{R}$, and biases $b^\ell_j \in \mathbb{R}$ it holds for all $x = (x_i)_{i=1}^{N_0}$, that:

$$w_j^1 := \sigma\left(\sum_{i=1}^{N_0} A^1_{i,j} x_i + b^1_j\right), \quad j \in \{1, \ldots, N_1\}$$

Definition (Contin.)

and

$$w_j^{\ell+1} := \sigma\left(\sum_{i=1}^{N_\ell} A_{i,j}^{\ell+1} w_i^\ell + b_j^{\ell+1}\right), \quad \ell \in \{1, \ldots, L-1\}, \quad j \in \{1, \ldots, N_{\ell+1}\}$$

and finally

$$\Phi(x) = \left(w_j^{L+1}\right)_{j=1}^{N_{L+1}} = \left(\sum_{i=1}^{N_L} A_{i,j}^{L+1} w_i^L + b_j^{L+1}\right)_{j=1}^{N_{L+1}}.$$

The number of hidden layers L of a DNN is referred to as its depth, denoted by $\mathrm{depth}(\Phi)$. If $L = 0$, then the previous equation holds with $w_i^0 := x_i$ for $i = 1, \ldots, N_0$. Such DNNs of depth 0 realize affine functions.

Definition (Contin.)

Define the total number of nonzero weights and biases as the size of the DNN, i.e.

$$\mathrm{size}(\Phi) := \left| \{(i,j,\ell) : A^{\ell}_{i,j} \neq 0\right\} \Big| + \Big| \left\{(j,\ell) : b^{\ell}_{j} \neq 0\right\} \Big| \,.$$

Let $\mathrm{size}_{\mathrm{in}}\,(\Phi)$ and $\mathrm{size}_{\mathrm{out}}\,(\Phi)$ be the number of nonzero weights and biases in the input resp. the output layer of Φ, i.e. $\mathrm{size}_{\mathrm{in}}(\Phi) := \left|\left\{(i,j) : A^{1}_{i,j} \neq 0\right\}\right| + \left|\left\{j : b^{1}_{j} \neq 0\right\}\right|$ and $\mathrm{size}_{\mathrm{out}}\,(\Phi) := \left|\left\{(i,j) : A^{L+1}_{i,j} \neq 0\right\}\right| + \left|\left\{j : b^{L+1}_{j} \neq 0\right\}\right|$, where A and b are the weights and biases.

For brevity say that $A \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ are the weights of layer $\ell \in \{1, \dots, L+1\}$ if $A_{ji} = A^{\ell}_{i,j}$. A DNN of depth 0 with weights A and bias b will be denoted by $((A, b))$.

Definition (Parallelization of DNNs)

Let Φ_1 be a DNN with input dimension n_1 and output dimension m_1 and Φ_2 be a DNN with input dimension n_2 and output dimension m_2. Let $L \in \mathbb{N}_0$ be the depth of both DNNs. The DNN $(\Phi_1, \Phi_2)_d$ is called **full parallelization of networks with distinct inputs** of Φ_1 and Φ_2:

$$(\Phi_1, \Phi_2)_d : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \to \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} : (x, x') \mapsto (\Phi_1(x), \Phi_2(x')).$$

It holds that

$$\operatorname{depth}((\Phi_1, \Phi_2)_d) = L, \quad \operatorname{size}((\Phi_1, \Phi_2)_d) = \operatorname{size}(\Phi_1) + \operatorname{size}(\Phi_2).$$

Definition (Concatenation of DNNs)

Let Φ_1 be a DNN with input dimension d and output dimension n and Φ_2 be a DNN with input dimension m and output dimension d. The **concatenation of Φ_1 and Φ_1** is defined by the map

$$\Phi_1 \circ \Phi_2 : \mathbb{R}^m \to \mathbb{R}^n : x \mapsto \Phi_1(\Phi_2(x)),$$

which is again a DNN with the properties

$$\operatorname{depth}(\Phi_1 \circ \Phi_2) = \operatorname{depth}(\Phi_1) + \operatorname{depth}(\Phi_2) - 1$$

and

$$\operatorname{size}(\Phi_1 \circ \Phi_2) \leq 2\operatorname{size}(\Phi_1) + 2\operatorname{size}(\Phi_2).$$

Definition (Identity Networks)

For all $n \in \mathbb{N}$ and $L \in \mathbb{N}_0$ the **identity network** $\Phi^{\mathrm{Id}}_{n,L}$ with

$$\Phi^{\mathrm{Id}}_{n,L} : \mathbb{R}^n \to \mathbb{R}^n : x \mapsto x$$

exists and has the properties

$$\operatorname{depth}(\Phi^{\mathrm{Id}}_{n,L}) = L, \quad \operatorname{size}(\Phi^{\mathrm{Id}}_{n,L}) \leq 2n(L+1).$$

Definition (Chebyshev Polynomials)

Let $n \in \mathbb{N}_0$. Define the univariate n-th **Chebyshev polynomial** of the first kind as T_n, such that

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

The recursive definition can be generalized to:

$$\forall m, n \in \mathbb{N}_0 : \quad T_{m+n} = 2T_m T_n - T_{|m-n|},$$

Definition (Chebyshev Polynomials)

Let $n \in \mathbb{N}_0$. Define the univariate n-th **Chebyshev polynomial** of the first kind as T_n, such that

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x).$$

The recursive definition can be generalized to:

$$\forall m, n \in \mathbb{N}_0 : \quad T_{m+n} = 2T_m T_n - T_{|m-n|},$$

For $K \in \mathbb{N}$ and $k = (k_j)_{j=1}^K \in \mathbb{N}_0^K$, denote **tensor product Chebyshev polynomials** by

$$T_k(x) := \prod_{j=1}^{K} T_{k_j}(x_j),$$

for $x = (x_j)_{j=1}^K \in [-1, 1]^K$.

Universität Münster

Notation

- *A superscript tilde (e.g. $\tilde{f}$) denotes a DNN. A superscript breve (e.g. $\breve{f}$) denotes a corrupted quantity. Typically, $\breve{f}$ denotes a numerical approximation of the map f, due to some measurement or due to some discretization error in approximating the map f.*
- *For finite index sets $\Lambda \subset \mathbb{N}_0^K$, denote the number of elements by $|\Lambda|$ and the maximum coordinatewise degree by $m_\infty(\Lambda) := \max_{k \in \Lambda} \|k\|_{\ell^\infty}$.*

Steps to construct DNNs approximating Chebyshev Polynomials

1. Construct DNNs that can multiply two numbers.
2. Construct DNNs that can multiply n numbers.
3. Construct DNNs that can approximate univariate Chebyshev polynomials using $T_{m+n} = 2T_mT_n - T_{|m-n|}$.
4. Construct DNNs that can approximate tensor product Chebyshev polynomials.

DNNs that emulate multiplication of two numbers

Lemma

For any $\delta \in (0,1)$ *and* $M \geq 1$ *there exists a ReLU DNN* $\tilde{\times}_{\delta,M} : [-M,M]^2 \to \mathbb{R}$ *such that*

$$\sup_{|a|,|b|\leq M} \left| ab - \tilde{\times}_{\delta,M}(a,b) \right| \leq \delta.$$

DNNs that emulate multiplication of two numbers

Lemma

For any $\delta \in (0,1)$ *and* $M \geq 1$ *there exists a ReLU DNN* $\tilde{\times}_{\delta,M} : [-M,M]^2 \to \mathbb{R}$ *such that*

$$\sup_{|a|,|b|\leq M} \left| ab - \tilde{\times}_{\delta,M}(a,b) \right| \leq \delta.$$

There exists a constant $C > 0$ *independent of* δ *and* M *such that* $\mathrm{size}_{in}\left(\tilde{\times}_{\delta,M}\right) \leq C$, $\mathrm{size}_{\mathrm{out}}\left(\tilde{\times}_{\delta,M}\right) \leq C$,

$$\mathrm{depth}\left(\tilde{\times}_{\delta,M}\right) \leq C\left(1 + \log_2(M/\delta)\right), \quad \mathrm{size}\left(\tilde{\times}_{\delta,M}\right) \leq C\left(1 + \log_2(M/\delta)\right).$$

Sketch of Proof

First define the sawtooth function $g : [0,1] \to [0,1]$ *as*

$$g(x) = \begin{cases} 2x & \text{if } x < \frac{1}{2} \\ 2(1-x) & \text{if } x \geq \frac{1}{2} \end{cases}$$

Sketch of Proof

First define the sawtooth function $g : [0,1] \to [0,1]$ *as*

$$g(x) = \begin{cases} 2x & \text{if } x < \frac{1}{2} \\ 2(1-x) & \text{if } x \geq \frac{1}{2} \end{cases}$$

and define the m-fold composition as $g_m = \underbrace{g \circ \cdots \circ g}_{m \text{ times}}$.

Sketch of Proof (Contin.)

Define the continuous, piecewise linear spline interpolation of x^2 *on* $[0, 1]$ *at the* $2^m + 1$ *uniformly distributed nodes* $j2^{-m}$ *for* $j = 0, \dots, 2^m \in \mathbb{N}_0$ *as* $f_m : [0, 1] \to [0, 1]$ *recursively with* $f_0(x) := x$ *and*

$$f_m(x) = f_{m-1}(x) - \frac{g_m(x)}{2^{2m}} \quad \forall m \geq 1.$$

It holds that the pointwise error is given by

$$\sup_{x \in [0,1]} \left| x^2 - f_m(x) \right| = 2^{-2m-2}.$$

Sketch of Proof (Contin.)

For any $M > 0$ *and* $a, b \in [-M, M]$ *it holds that*

$$
\begin{aligned}
& (a+b)^2 = a^2 + 2ab + b^2 \\
\Leftrightarrow \quad & ab = \frac{1}{2}\left((a+b)^2 - a^2 - b^2\right) \\
\Leftrightarrow \quad & ab = \frac{2M^2}{4M^2}\left(|a+b|^2 - |a|^2 - |b|^2\right) \\
\Leftrightarrow \quad & ab = 2M^2\left(\left(\frac{|a+b|}{2M}\right)^2 - \left(\frac{|a|}{2M}\right)^2 - \left(\frac{|b|}{2M}\right)^2\right).
\end{aligned}
$$

Universität Münster

Sketch of Proof (Contin.)

For any $M > 0$ *and* $a, b \in [-M, M]$ *it holds that*

$$\begin{aligned}
& (a+b)^2 = a^2 + 2ab + b^2 \\
\Leftrightarrow \quad & ab = \frac{1}{2}\left((a+b)^2 - a^2 - b^2\right) \\
\Leftrightarrow \quad & ab = \frac{2M^2}{4M^2}\left(|a+b|^2 - |a|^2 - |b|^2\right) \\
\Leftrightarrow \quad & ab = 2M^2\left(\left(\frac{|a+b|}{2M}\right)^2 - \left(\frac{|a|}{2M}\right)^2 - \left(\frac{|b|}{2M}\right)^2\right).
\end{aligned}$$

Define

$$\tilde{\times}_{\delta,M}(a,b) := 2M^2\left(f_m\left(\frac{|a+b|}{2M}\right) - f_m\left(\frac{|a|}{2M}\right) - f_m\left(\frac{|b|}{2M}\right)\right).$$

DNNs that emulate multiplication of n numbers

Proposition

For any $\delta \in (0,1)$, $n \in \mathbb{N}$ *and* $M \geq 1$ *there exists a ReLU DNN* $\tilde{\prod}^n_{\delta,M} : [-M,M]^n \to \mathbb{R}$ *such that*

$$\sup_{(x_i)_{i=1}^n \in [-M,M]^n} \left| \prod_{j=1}^n x_j - \tilde{\prod}^n_{\delta,M}(x_1,\ldots,x_n) \right| \leq \delta.$$

Universität Münster

DNNs that emulate multiplication of n numbers

Proposition

For any $\delta \in (0,1)$, $n \in \mathbb{N}$ and $M \geq 1$ there exists a ReLU DNN $\tilde{\prod}^n_{\delta,M} : [-M,M]^n \to \mathbb{R}$ such that

$$\sup_{(x_i)_{i=1}^n \in [-M,M]^n} \left| \prod_{j=1}^n x_j - \tilde{\prod}^n_{\delta,M}(x_1,\dots,x_n) \right| \leq \delta.$$

There exists a constant C independent of $\delta \in (0,1)$, $n \in \mathbb{N}$ and $M \geq 1$ such that

$$\operatorname{size}\left(\tilde{\prod}^n_{\delta,M}\right) \leq C\left(1 + n\log_2\left(nM^n/\delta\right)\right), \quad \operatorname{depth}\left(\tilde{\prod}^n_{\delta,M}\right) \leq C\left(1 + \log_2(n)\log_2\left(nM^n/\delta\right)\right).$$

Constructing univariate Chebyshev polynomial approximators

Lemma

There exists $C > 0$ such that for all $n \in \mathbb{N}$ and $\delta \in (0,1)$ there exist ReLU DNNs $\left\{\Phi_\delta^{\mathrm{Cheb},n}\right\}_{\delta\in(0,1)}$ with input dimension one and output dimension n which satisfy

$$\left\| T_\ell - \left(\Phi_\delta^{Cheb\,,n}\right)_\ell \right\|_{W^{1,\infty}([-1,1])} \le \delta, \quad \ell = 1, \dots, n$$

Universität Münster

Constructing univariate Chebyshev polynomial approximators

Lemma

There exists $C > 0$ such that for all $n \in \mathbb{N}$ and $\delta \in (0,1)$ there exist ReLU DNNs $\left\{\Phi_\delta^{\mathrm{Cheb},n}\right\}_{\delta\in(0,1)}$ with input dimension one and output dimension n which satisfy

$$\left\| T_\ell - \left(\Phi_\delta^{Cheb\,,n}\right)_\ell \right\|_{W^{1,\infty}([-1,1])} \leq \delta, \quad \ell = 1, \dots, n$$

and

$$\mathrm{depth}\left(\Phi_\delta^{\mathrm{Cheb},n}\right) \leq C(1+\log_2(n))\log_2(1/\delta) + C(1+\log_2(n))^3,$$

$$\mathrm{size}\left(\Phi_\delta^{\mathrm{Cheb},n}\right) \leq Cn\log_2(1/\delta) + Cn(1+\log_2(n)).$$

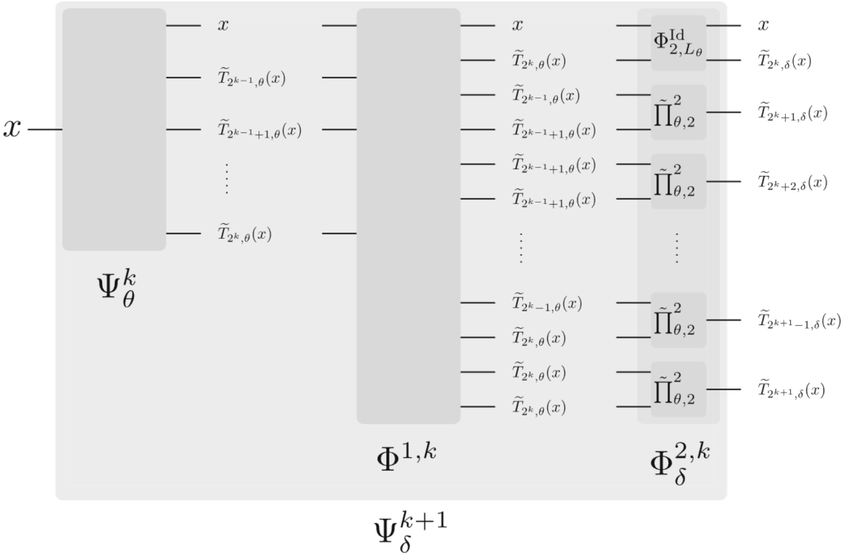
Sketch of Proof

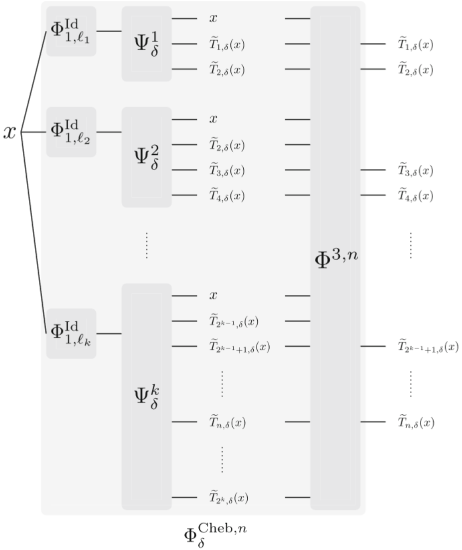
- *Inductively construct DNNs* Ψ_δ^k*, that take* x *as input and yield* x *and approximations for Chebyshev polynomials of degree* $2^k, \ldots, 2^{k+1}$ *of* x *with accuracy* δ *as output.*
- *For this use* $T_\ell = 2T_{\lceil \ell/2 \rceil} T_{\lfloor \ell/2 \rfloor} - T_{\lceil \ell/2 \rceil - \lfloor \ell/2 \rfloor}$*, which directly follows from* $T_{m+n} = 2T_m T_n - T_{|m-n|}$.
- *When constructing* Ψ_δ^{k+1} *use* Ψ_θ^k *with* $\theta = 2^{-2k-4}\delta$.
- *Parallelize* $\Psi_\delta^1, \ldots, \Psi_\delta^k$ *and filter the needed output.*

Sketch of Proof (Contin.)

For $k = 1$ *let* $\delta \in (0, 1)$ *and set* $L_1 := \operatorname{depth}\left(\tilde{\Pi}^2_{\delta/4,1}\right)$, *error bound* $\delta/4$ *and interval border* $M = 1$. *Let* A_i, b_i *for* $i = 1, \dots, L_1 + 1$ *be the weights and biases of* $\tilde{\Pi}^2_{\delta/4,1}$. *Furthermore let* $A := [1, 1]^T \in \mathbb{R}^{2\times 1}$ *and* $b := -1 \in \mathbb{R}$. *Define* Φ *as the DNN with weights* $A_1 A, A_2, \dots, A_{L_1}, 2A_{L_1+1}$ *and the biases* $b_1, \dots, b_{L_1}, b_{L_1+1} + b$. *Now define:*

$$\Psi^1_\delta := \left(\Phi^{\mathrm{Id}}_{1,L_1}, \Phi^{\mathrm{Id}}_{1,L_1}, \Phi\right),$$


x
Ψ_{θ}^{k}
x
$\widetilde{T}_{2^{k-1},\theta}(x)$
$\widetilde{T}_{2^{k-1}+1,\theta}(x)$
$\widetilde{T}_{2^{k},\theta}(x)$
$\Phi^{1,k}$
x
$\widetilde{T}_{2^{k},\theta}(x)$
$\widetilde{T}_{2^{k-1},\theta}(x)$
$\widetilde{T}_{2^{k-1}+1,\theta}(x)$
$\widetilde{T}_{2^{k-1}+1,\theta}(x)$
$\widetilde{T}_{2^{k-1}+1,\theta}(x)$
$\widetilde{T}_{2^{k}-1,\theta}(x)$
$\widetilde{T}_{2^{k},\theta}(x)$
$\widetilde{T}_{2^{k},\theta}(x)$
$\widetilde{T}_{2^{k},\theta}(x)$
$\Phi_{2,L_\theta}^{\mathrm{Id}}$
$\tilde{\prod}_{\theta,2}^{2}$
$\tilde{\prod}_{\theta,2}^{2}$
$\tilde{\prod}_{\theta,2}^{2}$
$\tilde{\prod}_{\theta,2}^{2}$
$\Phi_{\delta}^{2,k}$
x
$\widetilde{T}_{2^{k},\delta}(x)$
$\widetilde{T}_{2^{k}+1,\delta}(x)$
$\widetilde{T}_{2^{k}+2,\delta}(x)$
$\widetilde{T}_{2^{k+1}-1,\delta}(x)$
$\widetilde{T}_{2^{k+1},\delta}(x)$
Ψ_{δ}^{k+1}


x
$\Phi^{\mathrm{Id}}_{1,\ell_1}$
$\Phi^{\mathrm{Id}}_{1,\ell_2}$
$\Phi^{\mathrm{Id}}_{1,\ell_k}$
Ψ^1_δ
Ψ^2_δ
Ψ^k_δ
x
$\widetilde{T}_{1,\delta}(x)$
$\widetilde{T}_{2,\delta}(x)$
x
$\widetilde{T}_{2,\delta}(x)$
$\widetilde{T}_{3,\delta}(x)$
$\widetilde{T}_{4,\delta}(x)$
x
$\widetilde{T}_{2^{k-1},\delta}(x)$
$\widetilde{T}_{2^{k-1}+1,\delta}(x)$
$\widetilde{T}_{n,\delta}(x)$
$\widetilde{T}_{2^k,\delta}(x)$
$\Phi^{3,n}$
$\widetilde{T}_{1,\delta}(x)$
$\widetilde{T}_{2,\delta}(x)$
$\widetilde{T}_{3,\delta}(x)$
$\widetilde{T}_{4,\delta}(x)$
$\widetilde{T}_{2^{k-1}+1,\delta}(x)$
$\widetilde{T}_{n,\delta}(x)$
$\Phi^{\mathrm{Cheb},n}_\delta$

Approximation of Tensor Product Chebyshev Polynomials

Theorem

There exists a constant $C > 0$, such that for every $K \in \mathbb{N}$, every finite subset $\Lambda \subset \mathbb{N}_0^K$ and every $\delta \in (0,1)$ there exists a ReLU DNN $\Phi_{\Lambda,\delta}$ with input dimension K and output dimension $|\Lambda|$, such that the outputs of $\Phi_{\Lambda,\delta}$, which is denoted by $\left\{\tilde{T}_{k,\delta}\right\}_{k\in\Lambda}$, satisfy

$$\forall k \in \Lambda : \quad \left\|T_k - \tilde{T}_{k,\delta}\right\|_{W^{1,\infty}([-1,1]^K)} \leq \delta,$$

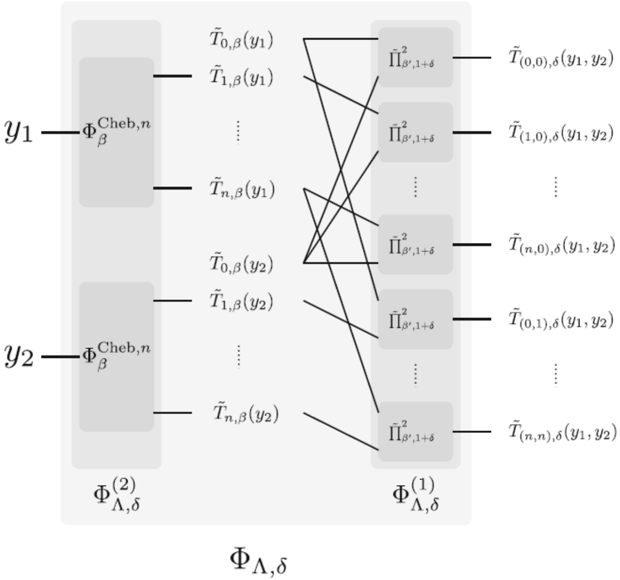
Approximation of Tensor Product Chebyshev Polynomials

Theorem

There exists a constant $C > 0$, such that for every $K \in \mathbb{N}$, every finite subset $\Lambda \subset \mathbb{N}_0^K$ and every $\delta \in (0,1)$ there exists a ReLU DNN $\Phi_{\Lambda,\delta}$ with input dimension K and output dimension $|\Lambda|$, such that the outputs of $\Phi_{\Lambda,\delta}$, which is denoted by $\{\tilde{T}_{k,\delta}\}_{k\in\Lambda}$, satisfy

$$\forall k \in \Lambda : \quad \left\| T_k - \tilde{T}_{k,\delta} \right\|_{W^{1,\infty}([-1,1]^K)} \leq \delta,$$

$$\begin{aligned}
\operatorname{depth}\left(\Phi_{\Lambda,\delta}\right) \leq & C\left(1+\log m_\infty(\Lambda)\right)^3 + C\left(1+\log(K)+\log m_\infty(\Lambda)\right)\log(1/\delta) \\
& + CK\log\left(m_\infty(\Lambda)\right) + CK\log K, \\
\operatorname{size}\left(\Phi_{\Lambda,\delta}\right) \leq & CK|\Lambda|\log\left(m_\infty(\Lambda)\right) + CK|\Lambda|\log(1/\delta) + CK^2|\Lambda| \\
& + CKm_\infty(\Lambda)\log\left(m_\infty(\Lambda)\right) + CKm_\infty(\Lambda)\log(1/\delta) + CK^2 m_\infty(\Lambda).
\end{aligned}$$


y_1
y_2
$\Phi_\beta^{\mathrm{Cheb},n}$
$\Phi_\beta^{\mathrm{Cheb},n}$
$\tilde{T}_{0,\beta}(y_1)$
$\tilde{T}_{1,\beta}(y_1)$
$\tilde{T}_{n,\beta}(y_1)$
$\tilde{T}_{0,\beta}(y_2)$
$\tilde{T}_{1,\beta}(y_2)$
$\tilde{T}_{n,\beta}(y_2)$
$\tilde{\Pi}^2_{\beta',1+\delta}$
$\tilde{T}_{(0,0),\delta}(y_1, y_2)$
$\tilde{T}_{(1,0),\delta}(y_1, y_2)$
$\tilde{T}_{(n,0),\delta}(y_1, y_2)$
$\tilde{T}_{(0,1),\delta}(y_1, y_2)$
$\tilde{T}_{(n,n),\delta}(y_1, y_2)$
$\Phi^{(2)}_{\Lambda,\delta}$
$\Phi^{(1)}_{\Lambda,\delta}$
$\Phi_{\Lambda,\delta}$

Lemma

Let $K \in \mathbb{N}$, $f : [-1, 1]^K \to \mathbb{R}$ a map which admits a holomorphic complex extension to the isotropic Bernstein polyellipse $\mathcal{E}_\varrho$ with $\varrho = (\rho, \ldots, \rho) \in (1, \infty)^K$ for some $\rho > 1$. Assume that an approximation $\check{f}$ of f is available and an upper bound on $\|f - \check{f}\|_{L^\infty([-1,1]^K)}$ exists. Then, for every $\rho' \in (1, \rho)$ and every $s \in \mathbb{N}$, there exists a constant $C'(s, \rho, \rho') > 0$ such that for all $n \in \mathbb{N}$:

$$\|f - \check{p}_{f,n}\|_{L^\infty([-1,1]^K)}$$

$$\leq C(n, K) \left(\frac{2\rho}{\rho - 1} \right)^K \max_{z \in \mathcal{E}_\varrho} |f(z)| \rho^{-n-1} + C(n, K) \|f - \check{f}\|_{L^\infty([-1,1]^K)},$$

$$\|f - \check{p}_{f,n}\|_{W^{s,\infty}([-1,1]^K)}$$

$$\leq C(n, K) \left(\frac{2C'\rho'}{\rho' - 1} \right)^K \max_{z \in \mathcal{E}_\varrho} |f(z)| \rho'^{-n-1} + C(n, K) \|f - \check{f}\|_{L^\infty([-1,1]^K)}.$$

Theorem

Let the assumptions of previous Lemma be true. Then there exists a ReLU DNN $\Phi_n^{\tilde{f}}$ for all $n \in \mathbb{N}$ with K-dimensional input and one-dimensional output, such that for all $\rho' \in (1, \rho)$, there exists a constant $C(K, \rho, \rho') > 0$ and a constant $c(\rho) > 0$, such that:

$$\left\| f - \Phi_n^{\tilde{f}} \right\|_{L^\infty([-1,1]^K)} \leq C \left(\max_{z \in \mathcal{E}_\varrho} |f(z)| + \|f - \check{f}\|_{L^\infty([-1,1]^K)} \right) \rho'^{-n} + \left(\frac{2}{\pi} \log(n+1) + 1 \right)^K \|f - \check{f}\|_{L^\infty([-1,1]^K)},$$

Theorem

Let the assumptions of previous Lemma be true. Then there exists a ReLU DNN $\Phi_n^{\tilde{f}}$ for all $n \in \mathbb{N}$ with K-dimensional input and one-dimensional output, such that for all $\rho' \in (1, \rho)$, there exists a constant $C(K, \rho, \rho') > 0$ and a constant $c(\rho) > 0$, such that:

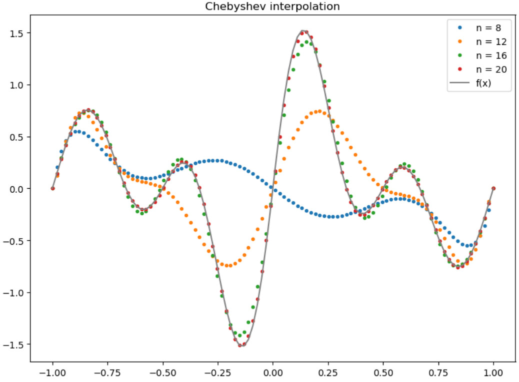
$$\begin{aligned}
\left\| f - \Phi_n^{\tilde{f}} \right\|_{L^\infty([-1,1]^K)} \leq & C \left(\max_{z \in \mathcal{E}_\varrho} |f(z)| + \|f - \check{f}\|_{L^\infty([-1,1]^K)} \right) \rho'^{-n} \\
& + \left(\frac{2}{\pi} \log(n+1) + 1 \right)^K \|f - \check{f}\|_{L^\infty([-1,1]^K)}, \\
\left\| f - \Phi_n^{\tilde{f}} \right\|_{W^{1,\infty}([-1,1]^K)} \leq & C \left(\max_{z \in \mathcal{E}_\varrho} |f(z)| + \|f - \check{f}\|_{L^\infty([-1,1]^K)} \right) \rho'^{-n} \\
& + n^2 \left(\frac{2}{\pi} \log(n+1) + 1 \right)^K \|f - \check{f}\|_{L^\infty([-1,1]^K)},
\end{aligned}$$

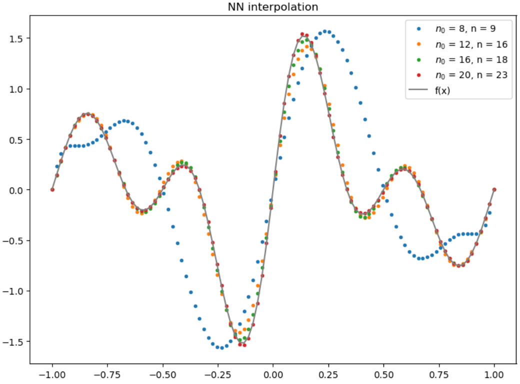
Theorem

Let the assumptions of previous Lemma be true. Then there exists a ReLU DNN $\Phi_n^{\tilde{f}}$ for all $n \in \mathbb{N}$ with K-dimensional input and one-dimensional output, such that for all $\rho' \in (1, \rho)$, there exists a constant $C(K, \rho, \rho') > 0$ and a constant $c(\rho) > 0$, such that:

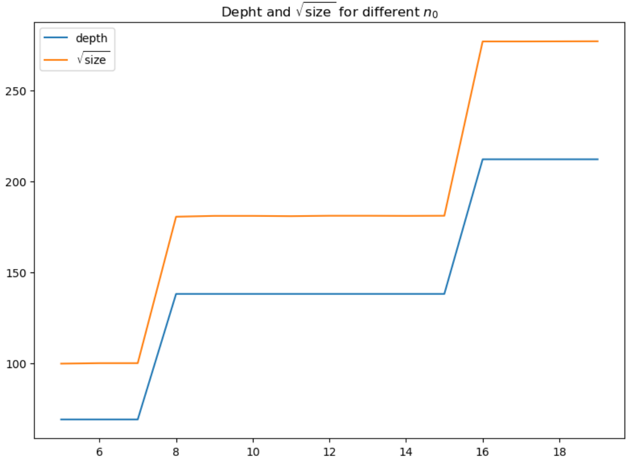
$$\begin{aligned}
\left\| f - \Phi_n^{\tilde{f}} \right\|_{L^\infty([-1,1]^K)} \leq & C \left(\max_{z \in \mathcal{E}_\varrho} |f(z)| + \|f - \check{f}\|_{L^\infty([-1,1]^K)} \right) \rho'^{-n} \\
& + \left(\frac{2}{\pi} \log(n+1) + 1 \right)^K \|f - \check{f}\|_{L^\infty([-1,1]^K)}, \\
\left\| f - \Phi_n^{\tilde{f}} \right\|_{W^{1,\infty}([-1,1]^K)} \leq & C \left(\max_{z \in \mathcal{E}_\varrho} |f(z)| + \|f - \check{f}\|_{L^\infty([-1,1]^K)} \right) \rho'^{-n} \\
& + n^2 \left(\frac{2}{\pi} \log(n+1) + 1 \right)^K \|f - \check{f}\|_{L^\infty([-1,1]^K)}, \\
\operatorname{depth}\left(\Phi_n^{\tilde{f}} \right) \leq & cKn(1 + \log(Kn)), \quad \operatorname{size}\left(\Phi_n^{\tilde{f}} \right) \leq cK^2(n+1)^{K+1}.
\end{aligned}$$

Approximation of the function $f(x) = e^{-|x|}(\sin(4\pi x) + \cos(2\pi(x - 1/4)))$ for $x \in [-1, 1]$

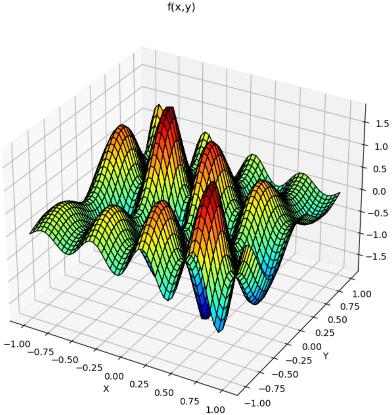




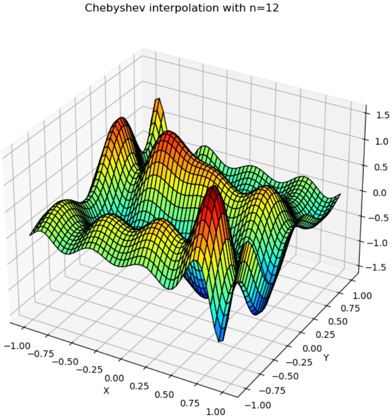
Universität
Münster

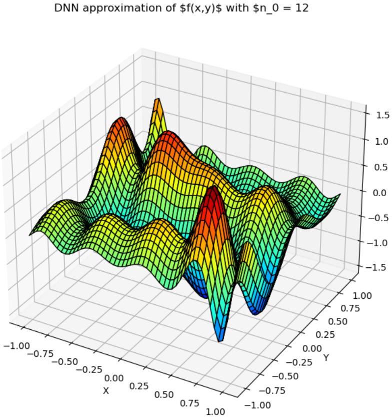


Approximation of the function
$f(x, y) = e^{-|x+y|}(\sin(\pi x/4) + \cos(2\pi(y - 1/4)))$

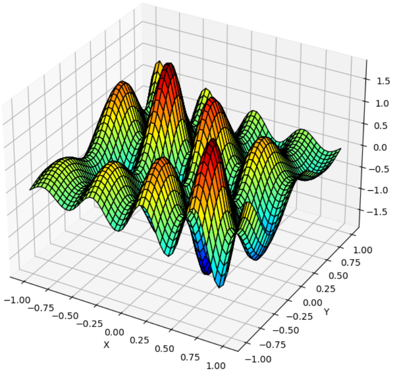


Universität Münster

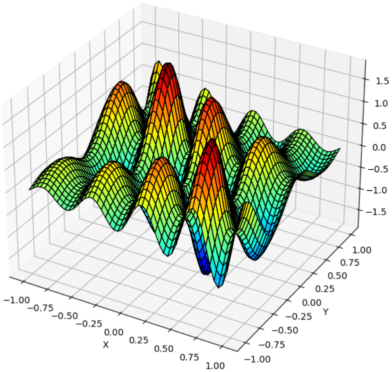




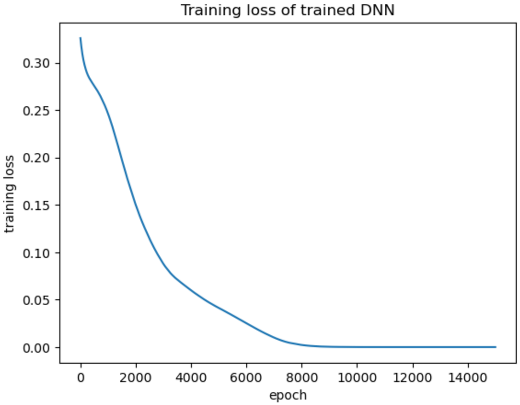
Chebyshev interpolation with n=20

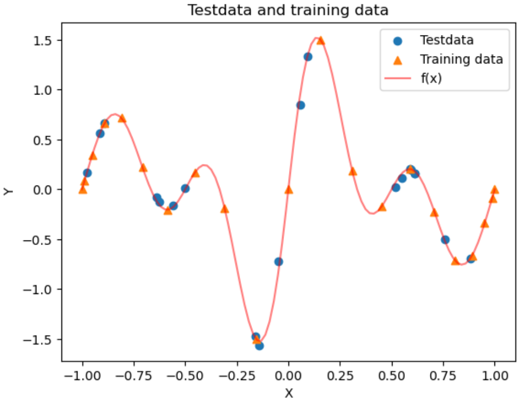


DNN approximation of $f(x,y)$ with $n_0 = 20

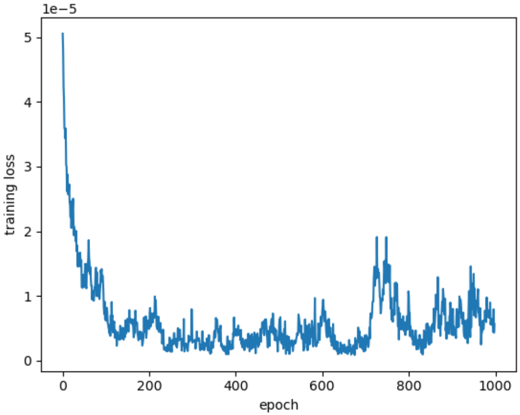


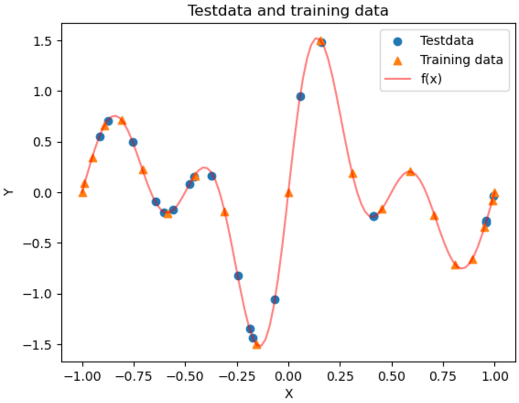
Using trained DNN, 3 Layers, 1024 Nodes per layer





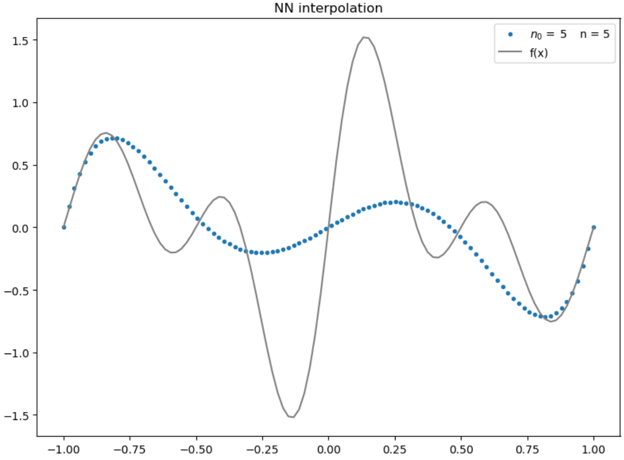
Training constructed DNN with 20 Layers



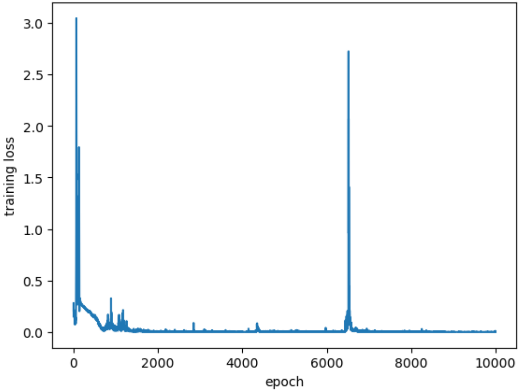


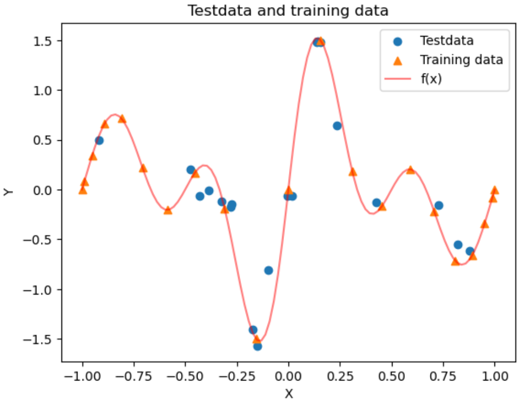
Universität Münster

Constructed DNN with 5 Layers



Training constructed DNN with 5 Layers





Conclusion

- Constructed DNNs can be implemented to validate the theoretical results. They are however due to their architecture not suitable for use in practical applications.

Conclusion

- Constructed DNNs can be implemented to validate the theoretical results. They are however due to their architecture not suitable for use in practical applications.
- The interesting behaviour when training DNNs with starting values from constructed DNNs invites for further studies.

Conclusion

- Constructed DNNs can be implemented to validate the theoretical results. They are however due to their architecture not suitable for use in practical applications.
- The interesting behaviour when training DNNs with starting values from constructed DNNs invites for further studies.
- Why use

$$2\left(\left(\frac{|a+b|}{2}\right)^2 - \left(\frac{|a|}{2}\right)^2 - \left(\frac{|b|}{2}\right)^2\right) = ab$$

  and not

$$\left(\left(\frac{|a+b|}{2}\right)^2 - \left(\frac{|a-b|}{2}\right)^2\right) = ab?$$

Thank you!

https://github.com/FaBremer/constructed_DNNs

Work based on:
Herrmann, L., Opschoor, J.A.A. & Schwab, C. Constructive Deep ReLU Neural Network Approximation. J Sci Comput 90, 75 (2022). https://doi.org/10.1007/s10915-021-01718-2

Figures 1-3 were used from this source.