A Technical results and complementary experiments

A.1 Proof of Lemma 1

We prove the result for $\mathscr{D}_2$. The result for $\mathscr{D}_k$ holds following a similar argument.

Fix $D_{2,\alpha} \in \mathscr{D}_2$, $\alpha \in \Lambda$. According to (1), we have, for $x \in \mathbb{R}^d$, $D_{2,\alpha}(x) = f_q \circ \cdots \circ f_1(x)$, where $f_i(t) = \sigma_2(V_i t + c_i)$ for $i = 1, \ldots, q-1$ (σ_2 is applied on pairs of components), and $f_q(t) = V_q t + c_q$. Therefore, for $(x,y) \in (\mathbb{R}^d)^2$,

$$\begin{aligned}
\|f_1(x) - f_1(y)\|_\infty &\leqslant \|V_1 x - V_1 y\|_\infty \\
&\quad \text{(since } \sigma_2 \text{ is 1-Lipschitz)} \\
&= \|V_1(x-y)\|_\infty \\
&\leqslant \|V_1\|_{2,\infty} \, \|x-y\| \\
&\leqslant \|x-y\| \\
&\quad \text{(by Assumption 1).}
\end{aligned}$$

Thus,

$$\begin{aligned}
\|f_2 \circ f_1(x) - f_2 \circ f_1(y)\|_\infty &\leqslant \|V_2 f_1(x) - V_2 f_1(y)\|_\infty \\
&\quad \text{(since } \sigma_2 \text{ is 1-Lipschitz)} \\
&\leqslant \|V_2\|_\infty \, \|f_1(x) - f_1(y)\|_\infty \\
&\leqslant \|f_1(x) - f_1(y)\|_\infty \\
&\quad \text{(by Assumption 1)} \\
&\leqslant \|x-y\|.
\end{aligned}$$

Repeating this, we conclude that, for each $\alpha \in \Lambda$ and all $(x,y) \in (\mathbb{R}^d)^2$, $|D_{2,\alpha}(x) - D_{2,\alpha}(y)| \leqslant \|x-y\|$, which is the desired result.

A.2 Proof of Lemma 2

Recall that $m_f \geqslant 2$. Throughout the proof, we let $\cdot$ refer to the dot product in $\mathbb{R}^d$. Let $(i,j) \in \{1, \ldots, m_f\}^2$, $i \neq j$. There exist $(a_i, b_i) \in \mathbb{R}^d \times \mathbb{R}$ and $(a_j, b_j) \in \mathbb{R}^d \times \mathbb{R}$ such that $\ell_i = a_i \cdot x + b_i$ and $\ell_j = a_j \cdot x + b_j$. Therefore,

$$\ell_i(x) - \ell_j(x) \leqslant 0 \iff x \cdot (a_i - a_j) \leqslant b_j - b_i.$$

So, there exist two subdomains $\tilde{\Omega}_1$ and $\tilde{\Omega}_2$, separated by an affine hyperplane, in which $\ell_i - \ell_j$ does not change sign. By repeating this operation for the $m_f(m_f-1)/2$ different pairs (ℓ_i, ℓ_j), we get that the number M_f of subdomains on which any pair $\ell_i - \ell_j$ does not change sign is smaller than the maximal number of arrangements of $m_f(m_f-1)/2$ hyperplanes.

Denoting by $C_{n,d}$ the maximal number of arrangements of n hyperplanes in $\mathbb{R}^d$, we know that when $d > n$ then $C_{n,d} = 2^n$, whereas if $n > d$ the upper bound $C_{n,d} \leqslant (1+n)^d$ becomes preferable (Devroye et al., 1996, Chapter 30). Thus, we have

$$m_f \leqslant M_f \leqslant \min\big(2^{m_f^2/2}, (m_f/\sqrt{2})^{2d}\big).$$

A.3 Proof of Proposition 1

We prove the first part of the proposition by using an induction on n. The case where $n = 1$ and thus $m = 2^1$ is clear since the function $f = \max(f_1, f_2)$ can be represented by a neural network of the form (1) with depth $q+1$ and size $s_1 + s_2 + 1$. Now, let $m = 2^n$ with $n > 1$. We have that $m/2 = 2^{n-1}$. By the induction hypothesis, $g_1 = \max(f_1, \ldots, f_{m/2})$ and $g_2 = \max(f_{m/2+1}, \ldots, f_m)$ can be represented by neural networks of the form (1) with depths $q+n-1$, and sizes at most $s_1 + \cdots + s_{m/2} + m/2 - 1$ and $s_{m/2+1} + \cdots + s_m + m/2 - 1$, respectively. Consequently, the function $G(x) = (g_1(x), g_2(x))$ can be implemented by a neural network of the form (1) with depth $q+n-1$ and size $s_1 + \cdots + s_m + m - 2$. Finally, by concatenating a one neuron layer, we have that the function $f = \max(g_1, g_2)$ can be represented by a neural network of the form (1) with depth $q + n = q + \log_2(m)$ and size at most $s_1 + \cdots + s_m + m - 1$.

Now, let us prove the case where m is arbitrary. Let $f_1, \ldots, f_m : \mathbb{R}^d \to \mathbb{R}$ be a collection of functions ($m \geqslant 2$), each represented by a neural network of the form (1) with depth q and size s_i, $i = 1, \ldots, m$. We prove below by an induction on n

that there exists a neural network of the form (1) with depth $q+\lceil \log_2(m)\rceil$, a final layer of width $v_{q-1}=2$, and a size at most $s_1+\cdots+s_m+2^{\lceil \log_2(m)\rceil}-1$ that represents the functions $f=\max(f_1,\ldots,f_m)$ and $g=\min(f_1,\ldots,f_m)$ (the symbol $\lceil\cdot\rceil$ stands for the ceiling function and the symbol $\lfloor\cdot\rfloor$ stands for the integer function).

The base case $m=2$ is clear using the GroupSort activation and $v_1=2$. For $m>2$, let $n\geqslant 2$ be such that $2^{n-1}\leqslant m<2^n$. Let $g_1=\max(f_1,\ldots,f_{2^{n-1}})$ and $g_2=\max(f_{2^{n-1}+1},\ldots,f_m)$. From the first part of the proof, we know that g_1 can be represented by a neural network of the form (1) with depth $q_1=q+\lfloor\log_2 m\rfloor=q+n-1$ and size $s_1+\cdots+s_{2^{n-1}}+2^{n-1}-1$. Also, by the induction hypothesis, g_2 can be represented by a neural network of the form (1) with depth $q_2=q+\lceil\log_2(m-2^{n-1})\rceil$ and size at most $s_{2^{n-1}+1}+\cdots+s_m+2^{\lceil\log_2(m-2^{n-1})\rceil}-1$. Therefore, by padding identity matrices with two neurons (recall that $v_{q_2-1}=2$) on layers from $q+\lceil\log_2(m-2^{n-1})\rceil$ to $q+n-1$, we have:

$$\begin{aligned}2^{\lceil\log_2(m-2^{n-1})\rceil}-1+2(n-2-\lceil\log_2(m-2^{n-1})\rceil) &= \sum_{k=0}^{k=\lceil\log_2(m-2^{n-1})\rceil-1} 2^k+\sum_{k=\lceil\log_2(m-2^{n-1})\rceil}^{k=n-2} 2^1\\ &\leqslant \sum_{k=0}^{k=n-2} 2^k = 2^{n-1}-1.\end{aligned}$$

Thus, g_2 can be represented by a neural network of the form (1) with depth $q_2=q+\lfloor\log_2 m\rfloor$ and size at most $s_{2^{n-1}+1}+\cdots+s_m+2^{n-1}-1$. Now, the bivariate function $G(x)=(g_1(x),g_2(x))$ can be implemented by a neural network of the form (1) with depth $q+\lfloor\log_2(m)\rfloor$ and size s such that

$$s\leqslant s_1+\cdots+s_m+2(2^{n-1}-1)=s_1+\cdots+s_m+2^n-2.$$

By concatenating a one neuron layer, we have that the function $f=\max(g_1,g_2)$ can be represented by a neural network of the form (1) with depth $q+\lceil\log_2(m)\rceil$ and size at most $s_1+\cdots+s_m+2^n-1=s_1+\cdots+s_m+2^{\lceil\log_2 m\rceil}-1$. The conclusion follows using the inequality $2^{\lceil\log_2 m\rceil}\leqslant 2m$.

A.4 Proof of Theorem 1

Let $f\in\mathrm{Lip}_1(\mathbb{R}^d)$ that is also m_f-piecewise linear. We know that each linear function can be represented by a 1-neuron neural network verifying Assumption 1 (no need for hidden layers). It is easy to see, using a small variant of Proposition 1, that any collection of $\tilde{m}$ linear functions with $\tilde{m}\leqslant m$ can be represented by a neural network of depth $\lceil\log_2(m)\rceil+1$ and size at most $3m-1$. Thus, combining (2) with Proposition 1, for each $k\in\{1,\ldots,M_f\}$ there exists a neural network of the form (1), verifying Assumption 1 and representing the function $\min_{i\in S_k}\ \ell_i$, with depth equal to $\lceil\log_2(m_f)\rceil+1$ (since $|S_k|\leqslant m_f$) and size at most $3m_f-1$.

Using again Proposition 1, we conclude that there exists a neural network of the form (1), verifying Assumption 1 and representing f, with depth$\lceil\log_2(M_f)\rceil+\lceil\log_2(m_f)\rceil+1$ and size at most $3m_fM_f+M_f-1$.

A.5 Proof of Corollary 1

According to He et al. (2018, Theorem A.1), the function f can be written as

$$f=\max_{1\leqslant k\leqslant m_f}\ \min_{i\in S_k}\ell_i,$$

where $|S_k|\leqslant m_f$. Using the same technique of proof as for Theorem 1, we find that there exists a neural network of the form (1), verifying Assumption 1 and representing f, with depth equal to $2\lceil\log_2(m_f)\rceil+1$ and size at most $3m_f^2+m_f-1$.

A.6 Proof of Proposition 2

Let $f\in\mathrm{Lip}_1(\mathbb{R})$ that is also m_f-piecewise linear. The proof of the first statement is an immediate consequence of Corollary 1 since connected subsets of $\mathbb{R}$ are also convex.

As for the second claim of the proposition, considering the case where f is convex, we know from He et al. (2018, Theorem A.1) that f can be written as

$$f=\max_{1\leqslant k\leqslant m_f}\ \ell_k.$$

Each function ℓ_k, $k = 1,\dots,m_f$, can be represented by a 1-neuron neural network verifying Assumption 1. Hence, by Proposition 1, there exists a neural network of the form (1), verifying Assumption 1 and representing f, with depth $\lceil \log_2(m_f)\rceil + 1$ and size at most $3m_f - 1$.

The last claim of the proposition for $m = 2^n$ is clear using Proposition 1.

A.7 Proof of Lemma 3

The result is proved by induction on q. To begin with, in the case $q = 2$ we have a neural network with one hidden layer. When applying the GroupSort function with a grouping size 2, every activation node is defined as the max or min between two different linear functions. The maximum number of breakpoints is equal to the maximum number of intersections, that is $\nu_1/2$. Thus, there is at most $\nu_1/2+1$ pieces.

Now, let us assume that the property is true for a given $q \geqslant 3$. Consider a neural network with depth q and widths $\nu_1,\dots,\nu_{q-1}$. Observe that the input to any node in the last layer is the output of a $\mathbb{R} \to \mathbb{R}$ GroupSort neural network with depth $(q-1)$ and widths $\nu_1,\dots,\nu_{q-2}$. Using the induction hypothesis, the input to this node is a function from $\mathbb{R} \to \mathbb{R}$ with at most $2^{q-3} \times (\nu_1/2+1) \times \cdots \times \nu_{q-2}$ pieces. Thus, after applying the GroupSort function with a grouping size 2, each node output is a function with at most $2 \times (2^{q-3} \times (\nu_1/2+1) \times \nu_2 \times \cdots \times \nu_{q-2})$. With the final layer, we take an affine combination of ν_{q-1} functions, each with at most $2^{q-2} \times (\nu_1/2+1) \times \nu_2 \times \cdots \times \nu_{q-2}$ pieces. In all, we therefore get at most $2^{q-2} \times (\nu_1/2+1) \times \nu_2 \times \cdots \times \nu_{q-1}$ pieces. The induction step is completed.

A.8 Proof of Corollary 2

Let f be an m_f-piecewise linear function. For a neural network of depth q and widths $\nu_1,\dots,\nu_q$ representing f, we have, by Lemma 3,

$$2^{q-1} \times (\nu_1/2+1) \times \cdots \times \nu_{q-1} \geqslant m_f.$$

By the inequality of arithmetic and geometric means, minimizing the size $s = \nu_1/2 + \cdots + \nu_k$ subject to this constraint, means setting $\nu_1/2+1 = \nu_2 = \cdots = \nu_k$. This implies that $s \geqslant \frac{1}{2}(q-1)m_f^{1/(q-1)}$.

A.9 Proof of Theorem 2

The proof follows the one from Cooper (1995, Theorem 3). Tesselate $[0,1]^d$ by cubes of side $s = \varepsilon/(2\sqrt{d})$ and denote by $n = (\lceil 1/s \rceil)^d$ the number of cubes in the tesselation. Choose n data points, one in each different cube. Then any Delaunay sphere will have a radius $R < \varepsilon/2M_f$. Now, construct $\tilde{f}$ by linearly interpolating between values of f over the Delaunay simplices. According to Seidel (1995), the number m_f of subdomains is $O(n^{d/2})$ and each of them is convex. Besides, by Cooper (1995, Lemma 2), $\tilde{f}$ guarantees an approximation error $\|f - \tilde{f}\|_\infty \leqslant \varepsilon$.

Using Corollary 1, we know that there exists a neural network of the form (1) verifying Assumption 1 and representing $\tilde{f}$. Besides, its depth is $2\lceil \log_2(m_f)\rceil + 1$ and its size is at most $3m_f^2 + m_f - 1$. Consequently, we have that the depth of the neural network is $2\lceil \log_2(m_f)\rceil + 1 = O(d^2 \log_2(\frac{2\sqrt{d}}{\varepsilon}))$ and the size at most $O(m^2) = O((\frac{2\sqrt{d}}{\varepsilon})^{d^2})$.

A.10 Proof of Proposition 3

Let $f \in \mathrm{Lip}_1([0,1])$ and f_m be the piecewise linear interpolation of f with the following $2^m + 1$ breakpoints: $k/2^m$, $k = 0,\dots,2^m$. We know that the function f_m approximates f with an error $\varepsilon_m \leqslant 2^{-m}$. In particular, for any $m \geqslant \log_2(1/\varepsilon)$, we have $\varepsilon_m \leqslant \varepsilon$. Besides, for any m, f_m is a 1-Lipschitz function defined on $[0,1]$, piecewise linear on 2^m subdomains. Thus, according to Proposition 2, there exists a neural network of the form (1), verifying Assumption 1 and representing f_m, with depth $2m+1$ and size at most $3 \times 2^{2m} + 2^m - 1$. Taking $m = \lceil \log_2(1/\varepsilon)\rceil$ shows the desired result.

Let $\varepsilon > 0$, let f be a convex (or concave) function in $\mathrm{Lip}_1([0,1])$, and let f_m be the piecewise linear interpolation of f with the following $2^m + 1$ breakpoints: $k/2^m$, $k = 0,\dots,2^m$. The function f_m approximates f with an error $\varepsilon_m = 2^{-m}$. In particular, for any $m \geqslant \log_2(1/\varepsilon)$, we have $\varepsilon_m \leqslant \varepsilon$. Besides, for any m, f_m is a 2^m-piecewise linear convex function defined on $[0,1]$. Hence, by Proposition 2, there exists a neural network of the form (1), verifying Assumption 1 and representing f_m, with depth $m+1$ and size at most $2 \times 2^m - 1$. Taking $m = \lceil \log_2(1/\varepsilon)\rceil$ leads to the desired result.

A.11 Proof of Proposition 4

We prove the result by using an induction on n. The case where $n = 1$ and thus $m = k^1$ is true since the function $f = \max(f_1,\dots,f_k)$ can be represented by a neural network of the form (1) with grouping size k, depth $q+1$, and size $s_1+\cdots+s_k+1$. Now, let $m = k^n$ with $n > 1$. We have that $\lfloor m/k \rfloor = \lceil m/k \rceil = m/k = k^{n-1}$. Let $g_1 = \max(f_1,\dots,f_{m/k}), g_2 = \max(f_{m/k+1},\dots,f_{2m/k}),\dots,g_k = \max(f_{((k-1)m/k)+1},\dots,f_m)$. By the induction hypothesis, $g_1,\dots,g_k$ can all be represented by neural networks of the form (1) with grouping size k, width depths equal to $q+n-1$ and sizes at most $s_1+\cdots+s_{m/k}+\frac{k^{n-1}-1}{k-1},\dots,s_{(k-1)m/k+1}+\cdots+s_m+\frac{k^{n-1}-1}{k-1}$, respectively.

Consequently, the function $G(x) = (g_1(x),\dots,g_k(x))$ can be implemented by a neural network of the form (1) with grouping size k, depth $q+n-1$, and size at most $s_1+\cdots+s_m+m-2$. Finally, by concatenating a one neuron layer, we see that the function $f = \max(g_1,\dots,g_k)$ can be represented by a neural network of the form (1) with depth $q+n = q+\log_k(m)$ and size at most

$$s_1+\cdots+s_m+k\Big(\frac{k^{n-1}-1}{k-1}\Big)+1 = s_1+\cdots+s_m+\frac{k^n-1}{k-1} = s_1+\cdots+s_m+\frac{m-1}{k-1}.$$

A.12 Proof of Corollary 3

According to He et al. (2018, Theorem A.1), the function f can be written as

$$f = \max_{1\leqslant k\leqslant m_f} \min_{i\in S_k} \ell_i,$$

where $|S_k| \leqslant m_f$ and $m_f = k^n$ for some $n \geqslant 1$. It is easy to see, using a small variant of Proposition 4, that any collection of $\tilde{m}$ linear functions with $\tilde{m} \leqslant m_f$ can be represented by a neural network of depth $\log_k(m)+1$ and size at most $\frac{m_f-1}{k-1}$. Therefore, by Proposition 4, there exists a neural network verifying Assumption 1 with grouping size k representing $\min\limits_{i\in S_k} \ell_i$ with depth $\log_k(m)+1$ and size at most $\frac{m_f-1}{k-1}$.

Using again Proposition 4, we find that there exists a neural network, verifying Assumption 1, with grouping size k, representing f with depth $2\log_k(m_f)+1$ and size at most

$$m_f\Big(\frac{m_f-1}{k-1}\Big)+\frac{m_f-1}{k-1} = \frac{m_f^2-1}{k-1}.$$

A.13 Proof of Lemma 4

The result is proved by induction on q. To begin with, in the case $q = 2$ we have a neural network with one hidden layer. When applying the GroupSort function with a grouping size k, the maximum number of breakpoints is equal to the maximum number of intersections of linear functions. In each group of k functions, there are at most $\frac{k(k-1)}{2}$ intersections. Thus, there are at most $\frac{k(k-1)}{2} \times \frac{v_1}{k} = \frac{(k-1)v_1}{2}$ breakpoints, that is $\frac{(k-1)v_1}{2}+1$ pieces.

Now, let us assume that the property is true for a given $q \geqslant 3$. Consider a neural network with depth q and widths $v_1,\dots,v_{q-1}$. Observe that the input to any node in the last layer is the output of a $\mathbb{R}\to\mathbb{R}$ GroupSort neural network with depth $(q-1)$ and widths $v_1,\dots,v_{q-2}$. Using the induction hypothesis, the input to this node is a function from $\mathbb{R}\to\mathbb{R}$ with at most $k^{q-3}\times(\frac{(k-1)v_1}{2}+1)\times\cdots\times v_{q-2}$ pieces. Thus, after applying the GroupSort function with a grouping size k, each node output is a function with at most $k\times(k^{q-3}\times(\frac{(k-1)v_1}{2}+1)\times v_2\times\cdots\times v_{q-2})$. With the final layer, we take an affine combination of v_{q-1} functions, each with at most $k^{q-2}\times(\frac{(k-1)v_1}{2}+1)\times v_2\times\cdots\times v_{q-2}$ pieces. In all, we therefore get at most $k^{q-2}\times(\frac{(k-1)v_1}{2}+1)\times v_2\times\cdots\times v_{q-1}$ pieces. The induction step is completed.

A.14 Proof of Theorem 3

The proof of Theorem 3 is straightforward and follows the one of Theorem 2 combined with the result obtained in Corollary 3.

A.15 Proof of Proposition 5

Let $f \in \mathrm{Lip}_1([0,1])$ and f_m be the piecewise linear interpolation of f with the following k^n+1 breakpoints: i/k^n, $k=0,\dots,k^n$. We know that the function f_m approximates f with an error $\varepsilon_m \leqslant k^{-n}$. In particular, for any $n \geqslant \log_k(1/\varepsilon)$, we have $\varepsilon_n \leqslant \varepsilon$. Besides, for any n, f_{k^n} is a 1-Lipschitz function defined on $[0,1]$, piecewise linear on k^n subdomains. Thus, according to Corollary 3, there exists a neural network of the form (1), verifying Assumption 1 and representing f_{k^n}, with grouping size k, depth $2n+1$, and size at most $\frac{k^{2n}-1}{k-1}$. Taking $n = \lceil \log_k(1/\varepsilon) \rceil$ shows the desired result.

B Experiments: Extended comparison between GroupSort and ReLU networks

We provide in this section further results and details on the experiments ran in Section 5.

B.1 Task 1: Approximating functions

Piecewise linear functions. We complete the experiments of Section 5 by estimating the 6-piecewise linear function f in the model $Y=f(X)$ (noiseless case, see Figure 7 and Figure 8) and in the model $Y=f(X)+\varepsilon$ (noisy case, see Figure 9 and Figure 10). Recall that in both cases, X follows a uniform distribution on $[-1.5,1.5]$ and the sample size is $n=100$.

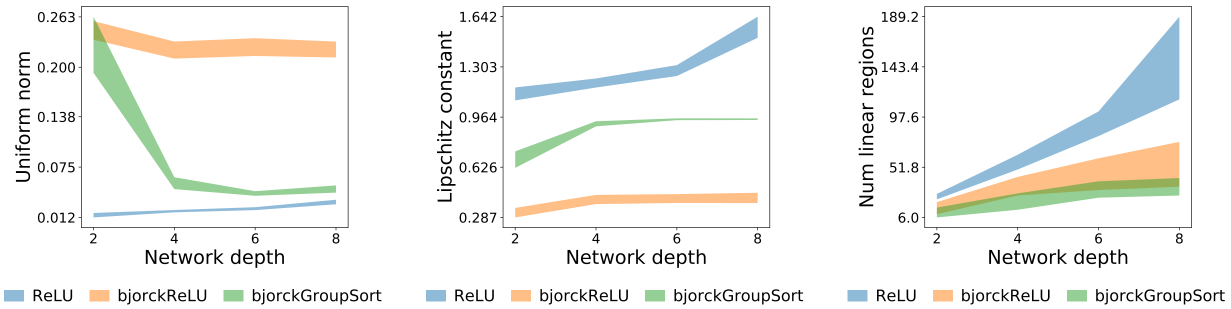
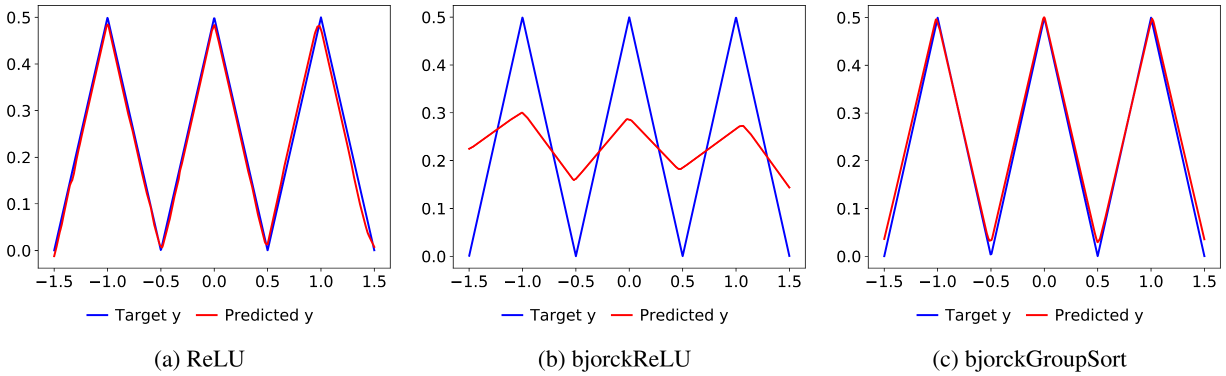


Figure 7: Estimating the 6-piecewise linear function in the model $Y=f(X)$, with a dataset of size $n=100$ (the thickness of the line represents a 95%-confidence interval).



(a) ReLU (b) bjorckReLU (c) bjorckGroupSort

Figure 8: Reconstructing the 6-piecewise linear function in the model $Y=f(X)$, with a dataset of size $n=100$.

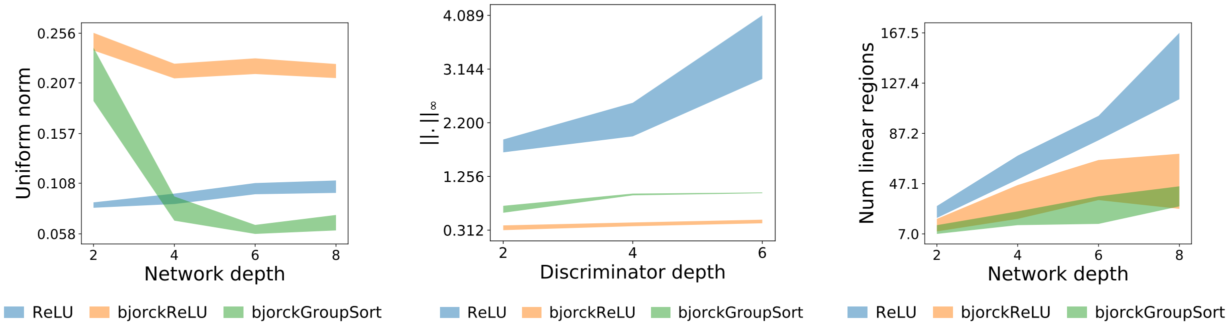
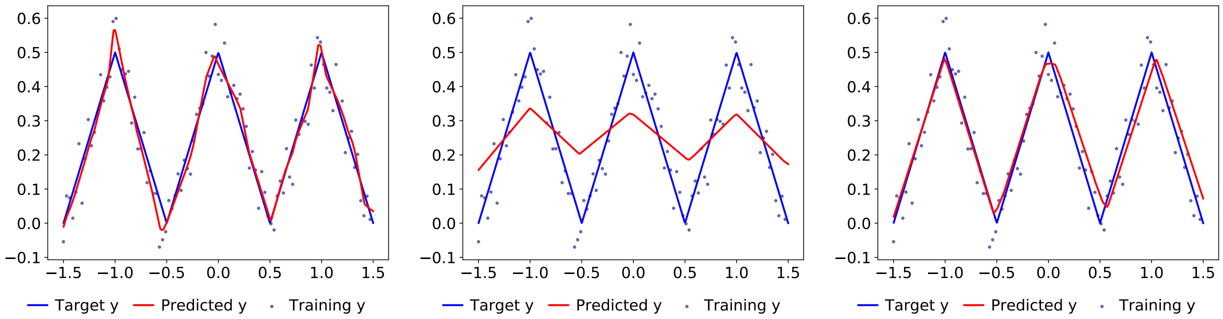


Figure 9: Estimating the 6-piecewise linear function in the model $Y = f(X) + \varepsilon$, with a dataset of size $n = 100$ (the thickness of the line represents a 95%-confidence interval).



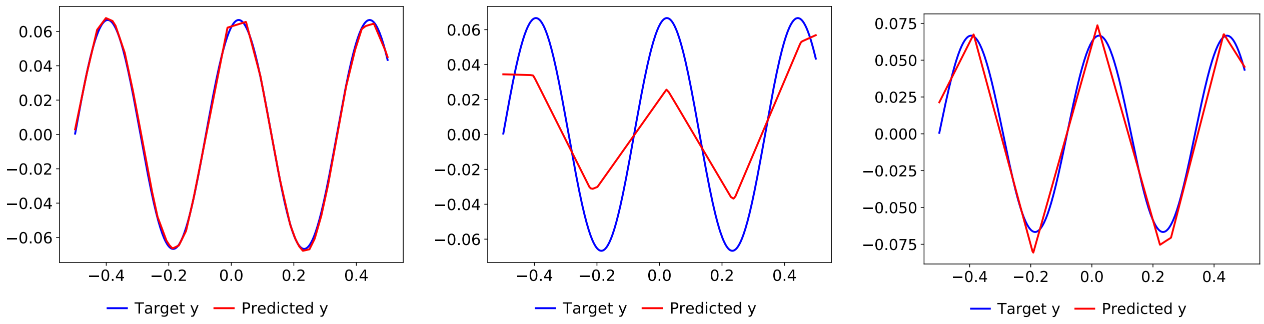
(a) ReLU

(b) bjorckReLU

(c) bjorckGroupSort

Figure 10: Reconstructing the 6-piecewise linear function in the model $Y = f(X) + \varepsilon$, with a dataset of size $n = 100$.

The sinus function. We provide in this subsection additional details for the learning of the sinus function $f(x) = (1/15)\sin(15x)$ defined on $[0,1]$ (see Section 5). Figure 11 is the case without noise while Figure 12 is the case with noise.

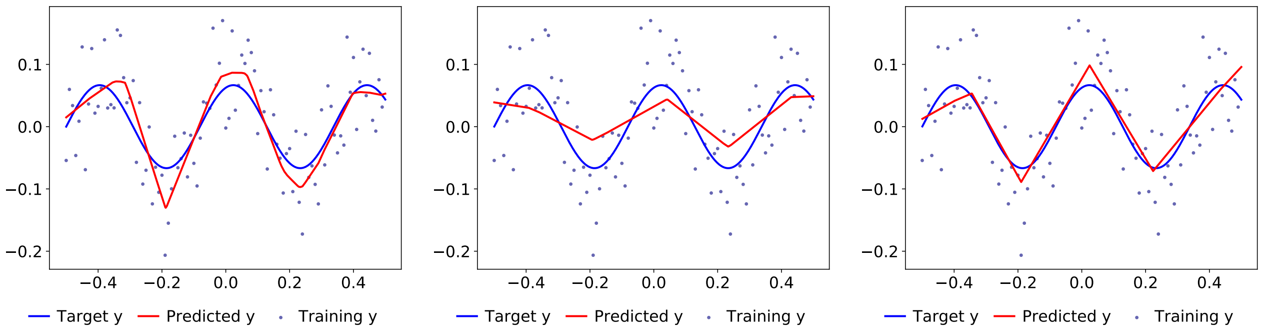


(a) ReLU

(b) bjorckReLU

(c) bjorckGroupSort

Figure 11: Reconstructing the function $f(x) = (1/15)\sin(15x)$ in the model $Y = f(X)$, with a dataset of size $n = 100$.



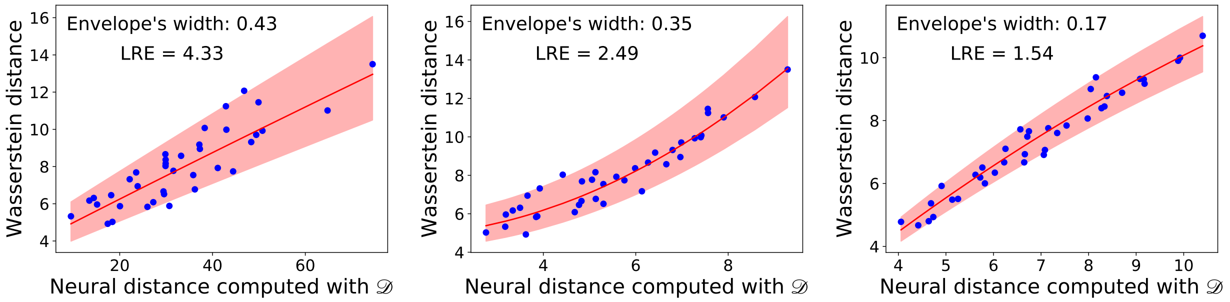
(a) ReLU

(b) bjorckReLU

(c) bjorckGroupSort

Figure 12: Reconstructing the function $f(x) = (1/15)\sin(15x)$ in the model $Y = f(X) + \varepsilon$, with a dataset of size $n = 100$.

B.2 Task 2: Calculating Wasserstein distances



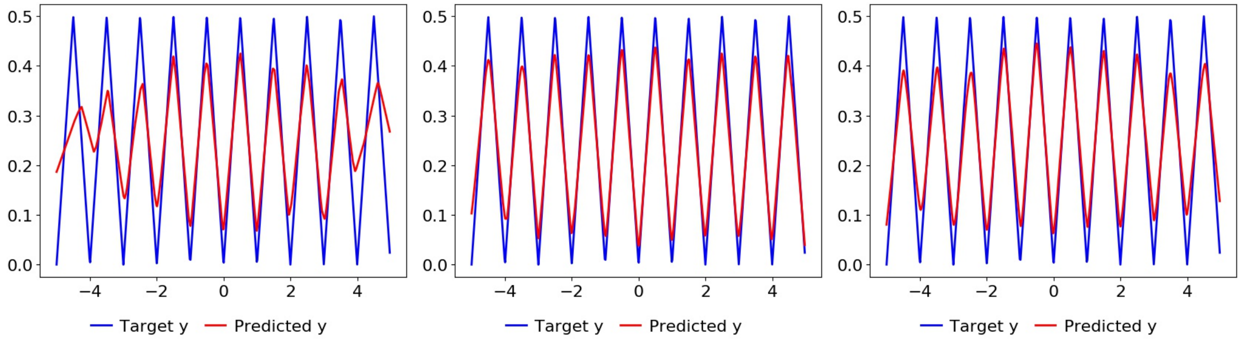
(a) $\mathscr{D}$ = ReLU network

(b) $\mathscr{D}$ = bjorckReLU network

(c) $\mathscr{D}$ = bjorckGroupSort network

Figure 13: Scatter plots of 40 pairs of Wasserstein and neural distances, for $q = 2$. The underlying distributions are bivariate Gaussian distributions with 4 components. The red curve is the optimal parabolic fitting and LRE refers to the Least Relative Error. The red zone is the envelope obtained by stretching the optimal curve.

C Study of increasing group sizes for GroupSort networks

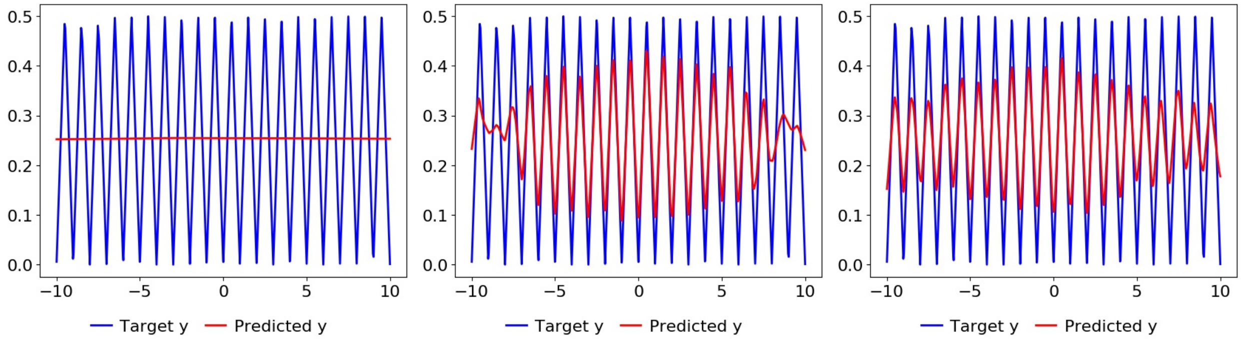


(a) Grouping size = 2

(b) Grouping size = 5

(c) Grouping size = 10

Figure 14: Reconstruction of a 20-piecewise linear function with varying grouping sizes ($k = 2, 5, 10$).



(a) Grouping size = 2

(b) Grouping size = 5

(c) Grouping size = 10

Figure 15: Reconstruction of a 40-piecewise linear function with varying grouping sizes ($k = 2, 5, 10$).

D Shared architecture for both GroupSort and ReLU networks

Operation	Feature Maps	Activation
$D(x)$		
Fully connected - q layers	width w	$\{$GroupSort, ReLU$\}$
Width w	$\{50\}$	
Depth q	$\{2, 4, 6, 8\}$	
Batch size	256	
Learning rate	0.0025	
Optimizer	Adam: $\beta_1 = 0.5$	$\beta_2 = 0.5$

Table 2: Hyperparameters used for the training of all neural networks