Neural Networks 175 (2024) 106297

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

Expressive power of ReLU and step networks under floating-point operations

Yeachan Park [a,1], Geonho Hwang [a,1], Wonyeol Lee [b], Sejun Park [c,*]

[a] Korea Institute for Advanced Study, Seoul, 02455, Republic of Korea
[b] Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, United States
[c] Department of Artificial Intelligence, Korea University, Seoul, 02841, Republic of Korea

ARTICLE INFO

Keywords:
Neural networks
Universal approximation
Memorization
Floating-point arithmetic

ABSTRACT

The study of the expressive power of neural networks has investigated the fundamental limits of neural networks. Most existing results assume real-valued inputs and parameters as well as exact operations during the evaluation of neural networks. However, neural networks are typically executed on computers that can only represent a tiny subset of the reals and apply inexact operations, i.e., most existing results do not apply to neural networks used in practice. In this work, we analyze the expressive power of neural networks under a more realistic setup: when we use floating-point numbers and operations as in practice. Our first set of results assumes floating-point operations where the significand of a float is represented by finite bits but its exponent can take any integer value. Under this setup, we show that neural networks using a binary threshold unit or ReLU can memorize any finite input/output pairs and can approximate any continuous function within an arbitrary error. In particular, the number of parameters in our constructions for universal approximation and memorization coincides with that in classical results assuming exact mathematical operations. We also show similar results on memorization and universal approximation when floating-point operations use finite bits for both significand and exponent; these results are applicable to many popular floating-point formats such as those defined in the IEEE 754 standard (e.g., 32-bit single-precision format) and bfloat16.

1. Introduction

Identifying the expressive power of neural networks is an important problem in the theory of deep learning. Theoretical results represented by the universal approximation theorem have shown that neural networks with sufficiently large width and depth are able to approximate a continuous function on a compact domain within an arbitrary error (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989; Lu, Pu, Wang, Hu, & Wang, 2017; Park, Yun, Lee & Shin, 2021; Pinkus, 1999). Another line of work on memory capacity has studied regression problems and showed that shallow networks with $O(n)$ parameters can fit arbitrary n input/output pairs (Baum, 1988; Huang & Babri, 1998; Vershynin, 2020; Yun, Sra, & Jadbabaie, 2019), while $o(n)$ parameters are sufficient for deep ones with $\omega(1)$ layers (Park, Lee, Yun & Shin, 2021; Vardi, Yehudai, & Shamir, 2022). However, since these results assume exact mathematical operations, they do not apply to neural networks executed on computers that can only represent a tiny subset of the reals (e.g., floating-point numbers) and perform inexact operations (e.g., floating-point operations) (Puheim, Nyulászi, Madarász, & Gašpar, 2014; Wray & Green, 1995).

Several works studied the expressive power under machine-representable parameters. For example, Ding, Liu, Xiong, and Shi (2019) showed that for approximating a continuous function within ε error using quantized weights, $O(K \log^5 \varepsilon)$ parameters are sufficient where K denotes the number of parameters for approximation using real-valued parameters. The memory capacity of networks with quantized weights was also studied in Park, Lee, et al. (2021), Vardi et al. (2022). However, these works also assume exact mathematical operations, and hence, what neural networks can/cannot do on actual computers has remained unknown.

Most neural networks used in practice operate under floating-point arithmetic. Typically, a floating-point number x can be written as $x = s_x \times a_x \times 2^{e_x}$, where s_x, a_x, and e_x are called the *sign*, *significand*, and *exponent* of x, respectively. Here, to express x using a finite memory, s_x, a_x, and e_x must have finite-bit representations: e.g., in 32-bit single-precision floats, s_x uses 1 bit, a_x uses 23 bits, and e_x uses 8 bits (IEEE Computer Society, 2019). If we apply floating-point addition/multiplication to two floating-point numbers, the result is rounded to a nearest[2] floating-point number; hence, floating-point operations may incur some rounding error.

* Corresponding author.
E-mail address: sejun.park000@gmail.com (S. Park).

[1] Equal contribution.

[2] There are other rounding modes (e.g., "round towards 0") as well, but this paper assumes the "round to nearest (ties to even)" mode since it is the most commonly used one (e.g., it is the default rounding mode in the IEEE 754 standard (IEEE Computer Society, 2019)).

https://doi.org/10.1016/j.neunet.2024.106297
Received 15 January 2024; Received in revised form 4 April 2024; Accepted 7 April 2024
Available online 9 April 2024

Although the relative error of each floating-point operation is usually small, the relative error incurred by a composition of such operations can be arbitrarily large, even with a small number of operations. For example, let δ be some small floating-point number such that $1/\delta$ is also a floating-point number, and consider the expression $(1/\delta)\times(1+\delta-1)$. If floating-point operations are applied to this expression and δ is small enough, then 1 is returned for the expression $1+\delta$. This implies that 0 is returned for $(1+\delta-1)$ as well as for the entire expression. Thus, the final output 0 has a relative error one compared to the true output 1 (which is computed under exact operations).

1.1. Summary of contribution

Unlike prior works considering exact mathematical operations that typical neural network implementations do not perform, in this work, we investigate the expressive power of neural networks under floating-point operations, with the following activation functions: $\mathrm{Step}(x) = \mathbb{1}[x \geq 0]$ and $\mathrm{ReLU}(x) = \max\{x, 0\}$. To our knowledge, this is the first work that considers practical neural networks typically implemented by computers using floating-point numbers (e.g., inputs, parameters, and intermediate values) and operations (e.g., addition and multiplication).

Our first set of results is for $\mathbb{F}_p$, a set of floating-point numbers with *p-bit significand and unbounded exponent* (i.e., the exponent can be any integer), and an input dimension $d \leq 2^p$. We note that $\mathbb{F}_p$ have been widely used in the floating-point literature since floating-point operations in $\mathbb{F}_p$ do not incur any overflow and underflow (Boldo, 2015; Boldo & Melquiond, 2011; Jeannerod, 2015; Jeannerod, Louvet, Muller, & Plet, 2016; Jeannerod & Rump, 2018), i.e., $\mathbb{F}_p$ is often easier to analyze compared to the bounded exponent case.

- As in the classical results, we show for Step networks that $O(n)$ parameters are sufficient for memorizing arbitrary n input/output pairs. Theorem 2 states that there exists a Step network $f_\theta : \mathbb{F}_p^d \to \mathbb{F}_p$ of $O(n)$ parameters that is parameterized by θ, uses only floating-point operations, and satisfies the following: for any $D = \{(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_n, y_n)\} \subset \mathbb{F}_p^d \times \mathbb{F}_p$ such that $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D satisfying
$$f_{\theta_D}(\mathbf{z}_i) = y_i, \text{ for all } i.$$
- We next show that Step networks can also universally approximate continuous functions on a unit cube. Theorem 3 states that for any continuous $f^* : [0,1]^d \to \mathbb{R}$ and $\varepsilon > 0$, there exists a floating-point Step network $f : [0,1]^d \cap \mathbb{F}_p^d \to \mathbb{F}_p$ such that
$$|f(\mathbf{x}) - f^*(\mathbf{x})| \leq |f^*(\mathbf{x}) - \lceil f^*(\mathbf{x}) \rfloor| + \varepsilon, \text{ for all } \mathbf{x} \in [0,1]^d \cap \mathbb{F}_p^d,$$
where $\lceil f^*(\mathbf{x}) \rfloor$ denotes a floating-point number in $\mathbb{F}_p$ closest to $f^*(\mathbf{x})$. The first term in the error bound denotes an intrinsic error arising from the nature of floating-point arithmetic, i.e., one cannot achieve a smaller error than this term.
- By carefully analyzing floating-point operations, we further extend these results to ReLU networks in Theorems 5 and 6.

Our next set of results considers a more realistic class of floating-point numbers $\mathbb{F}_{p,q}$ with *p-bit significand and q-bit exponent*, i.e., each number in $\mathbb{F}_{p,q}$ can be represented by a finite number of bits. In particular, we focus on p, q satisfying $q \geq 5$ and $4 \leq p \leq 2^{q-2}+2$. This condition on p, q is met by many practical floating-point formats such as those defined in the IEEE 754 standard (e.g., 32-bit single-precision format) (IEEE Computer Society, 2019) and bfloat16 (Abadi et al., 2016); see Section 2.2 for details. We note that operations in $\mathbb{F}_{p,q}$ may incur overflow or underflow unlike in $\mathbb{F}_p$.

- Theorems 8 and 9 show that memorization and universal approximation using Step networks are possible under $\mathbb{F}_{p,q}$ with a similar number of parameters for the $\mathbb{F}_p$ case. This shows that Step networks are expressive even under realistic floating-point operations.
- Showing similar memorization and universal approximation results for ReLU networks is more challenging since the output of an activation function can be very large unlike in Step networks; this can incur overflow if the output is multiplied by a large weight in the next layer. Nevertheless, by carefully analyzing floating-point operations under $\mathbb{F}_{p,q}$, we show in Theorem 10 that $O(n)$ parameters are sufficient for memorizing arbitrary n input/output pairs $(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_n, y_n)$, under a mild condition on inputs. In addition, Theorem 11 shows that universal approximation using ReLU networks is possible under $\mathbb{F}_{p,q}$.

We lastly note that the numbers of parameters used in our results to show memorization and universal approximation have the same order compared to the necessary and sufficient numbers of parameters shown in corresponding prior results under exact operations.

1.2. Organization

We introduce our problem setup and notations in Section 2. We then formally describe our results on the expressive power of Step networks and ReLU networks under $\mathbb{F}_p$ and $\mathbb{F}_{p,q}$ in Sections 3 and 4. We present the proofs of these results in Sections 5 and 6, and conclude the paper in Section 8.

2. Problem setup and notations

2.1. Notations

We first introduce the notations used in this paper. For $n \in \mathbb{N}$, we use $[n] \triangleq \{1, \dots, n\}$. We often use lower-case alphabets $a, b, c, \dots$ for denoting scalar values and bold lower-case alphabets $\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots$ for denoting column vectors where f, g, h are reserved for denoting a scalar-valued functions. For $n \in \mathbb{N}$, we use $\mathbf{1}_n$ to denote the n-dimensional vector consisting of ones. For a vector $\mathbf{x}$, we use x_i to denote its ith coordinate. For $\mathcal{S} \subset \mathbb{R}$ and $x \in \mathbb{R}$, we use $x^{(\geq;\mathcal{S})} \triangleq \inf_{v \in \mathcal{S}: v \geq x} v$, $x^{(\leq;\mathcal{S})} \triangleq \sup_{v \in \mathcal{S}: v \leq x} v$, $x^{(>;\mathcal{S})} \triangleq \inf_{v \in \mathcal{S}: v > x} v$, and $x^{(<;\mathcal{S})} \triangleq \sup_{v \in \mathcal{S}: v < x} v$. Likewise, for $\mathbf{x} = (x_1, \dots, x_d)$, we define $\mathbf{x}^{(\leq;\mathcal{S})} \triangleq (x_1^{(\leq;\mathcal{S})}, \dots, x_d^{(\leq;\mathcal{S})})$ and use $\mathbf{x}^{(\geq;\mathcal{S})}$, $\mathbf{x}^{(<;\mathcal{S})}$, and $\mathbf{x}^{(>;\mathcal{S})}$ similarly. For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we use $[\mathbf{x}, \mathbf{y}] \triangleq [x_1, y_1] \times \cdots \times [x_d, y_d]$; we also use $[\mathbf{x}, \mathbf{y})$, $(\mathbf{x}, \mathbf{y}]$, and $(\mathbf{x}, \mathbf{y})$ similarly. We use $\dot\bigcup$ to denote the disjoint union.

We use $C(\mathcal{X}, \mathcal{Y})$ to denote the set of all continuous functions from $\mathcal{X}$ to $\mathcal{Y}$. Given a continuous function $f : \mathcal{X} \to \mathbb{R}$ for some compact $\mathcal{X} \subset \mathbb{R}^d$, we use $\omega_f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ to denote the modulus of continuity of f, defined as
$$\omega_f(\delta) \triangleq \sup_{\mathbf{x}, \mathbf{x}' \in \mathcal{X}: \|\mathbf{x} - \mathbf{x}'\|_\infty \leq \delta} |f(\mathbf{x}) - f(\mathbf{x}')|,$$
and we use $\omega_f^{-1} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ to denote its inverse, defined as
$$\omega_f^{-1}(\varepsilon) \triangleq \sup\{\delta \geq 0 : \omega_f(\delta) \leq \varepsilon\}.$$
Throughout this paper, we treat the input dimension d as a constant and often hide it in the big-O notation $O(\cdot)$.

2.2. Floating-point numbers

We consider two types of floating-point numbers:
$$\begin{aligned}\mathbb{F}_p &\triangleq \left\{s \times (1.a_1 \cdots a_p) \times 2^b : s \in \{-1, 1\}, a_1, \dots, a_p \in \{0, 1\}, b \in \mathbb{Z}\right\} \cup \left\{0\right\},\\ \mathbb{F}_{p,q} &\triangleq \left\{s \times (1.a_1 \cdots a_p) \times 2^{b-1+e_{min}} : s \in \{-1, 1\}, a_1, \dots, a_p \in \{0, 1\}, b \in [2^q - 2]\right\}\\ &\quad \cup \left\{s \times (0.a_1 \cdots a_p) \times 2^{e_{min}} : s \in \{-1, 1\}, a_1, \dots, a_p \in \{0, 1\}\right\}, \qquad (1)\end{aligned}$$
where $1.a_1 \cdots a_p$ and $0.a_1 \cdots a_p$ denote binary representations and $e_{min} \triangleq -2^{q-1}+2$. We also use $e_{max} \triangleq 2^{q-1}-1$ for $\mathbb{F}_{p,q}$. In Eq. (1), the first factor (i.e., s) of each floating-point number is called the *sign*, the second factor (e.g., $1.a_1 \cdots a_p$) called the *significand*, and $\log_2$ of the third factor (e.g., b) called the *exponent*. For instance, the sign, significand, and

exponent of any $x \in \mathbb{F}_{p,q}$ are always in $\{-1, 1\}$, $[0, 2)$, and $[e_{min}, e_{max}]$, respectively. Many floating-point formats used in practice can also represent three special values: $+\infty$, $-\infty$, and NaN (not-a-number). We do not include these values in $\mathbb{F}_p$ and $\mathbb{F}_{p,q}$, yet we do consider them in our results (including proofs). For $\mathbb{F}_{p,q}$, we assume that p and q satisfy $q \geq 5$ and $4 \leq p \leq 2^{q-2} + 2$.

As we introduced in Section 1.1, floating-point numbers with unbounded exponent $\mathbb{F}_p$ have been widely used since it does not incur any underflow and overflow (i.e., rounding to 0 or $\pm\infty$) unlike those over $\mathbb{F}_{p,q}$ (Boldo, 2015; Boldo & Melquiond, 2011; Jeannerod, 2015; Jeannerod et al., 2016; Jeannerod & Rump, 2018). On the other hand, each number in $\mathbb{F}_{p,q}$ can be represented by a finite number of bits, and $\mathbb{F}_{p,q}$ covers many practical floating-point formats. For example, the following floating-point formats defined in the IEEE 754 standard (IEEE Computer Society, 2019) are all instances of $\mathbb{F}_{p,q}$: the 16-bit half-precision format has $(p, q) = (10, 5)$; the 32-bit single-precision format has $(p, q) = (23, 8)$; the 64-bit double-precision format has $(p, q) = (52, 11)$; and the 128-bit quadruple-precision format has $(p, q) = (112, 15)$. Also, the bfloat16 format Abadi et al. (2016), frequently used in machine learning these days, is also an instance of $\mathbb{F}_{p,q}$ with $(p, q) = (7, 8)$.

We say a non-zero number $x \in \mathbb{F}_{p,q}$ is *normal* if $|x| \geq 2^{e_{min}}$ (i.e., significand is at least 1), and is *subnormal* if $|x| < 2^{e_{min}}$ (i.e., significand is smaller than 1). In $\mathbb{F}_{p,q}$, we denote the unit round-off as $u \triangleq 2^{-p}$, the largest number as $\Omega \triangleq (2-u) \times 2^{e_{max}}$, and the smallest positive number as $\eta \triangleq 2^{-p+e_{min}}$. We also use $u \triangleq 2^{-p}$ for $\mathbb{F}_p$. We often use $\mathbb{F}$ to denote either $\mathbb{F}_p$ or $\mathbb{F}_{p,q}$.

2.3. Floating-point operations

For both $\mathbb{F}_p$ and $\mathbb{F}_{p,q}$, we consider the rounding mode called "round to nearest (ties to even)". Roughly, we round $x \in \mathbb{R}$ to a floating-point number $y \in \mathbb{F}$ that is closest to x, where ties are broken by choosing a floating-point number that has 0 as the pth binary digit of its significand (i.e., $a_p = 0$ in Eq. (1)). Formally, the rounding of $x \in \mathbb{R}$ in $\mathbb{F}$ is defined as

$$\lceil x \rfloor_{\mathbb{F}_p} \triangleq \operatorname*{arg\,min}_{y \in \mathbb{F}_p} |x - y|, \lceil x \rfloor_{\mathbb{F}_{p,q}} \triangleq \begin{cases} \arg\min_{y \in \mathbb{F}_{p,q}} |x - y| & \text{if } x \in (-\Omega', \Omega'), \\ +\infty & \text{if } x \in [\Omega', \infty), \\ -\infty & \text{if } x \in (-\infty, -\Omega'], \end{cases}$$

where ties are broken as explained above and $\Omega' = \Omega + 2^{e_{max}-p-1}$ (Boldo, Jeannerod, Melquiond, & Muller, 2023, Chapter 2.1). If $\mathbb{F}$ is clear from context, we use $\lceil x \rfloor$ to denote $\lceil x \rfloor_{\mathbb{F}}$. For $x \in \mathbb{F}$, we use $x^- \triangleq \sup_{z \in \mathbb{F}: z < x} z$ and $x^+ \triangleq \inf_{z \in \mathbb{F}: z > x} z$ if exist: e.g., 0^+ and 0^- do not exist in $\mathbb{F}_p$. We note that all network constructions in our results do not generate infinities and NaN during all operations including intermediate ones.

We consider the following floating-point operations for $\mathbb{F}$: $a \oplus_{\mathbb{F}} b \triangleq \lceil a + b \rfloor_{\mathbb{F}}$, $a \ominus_{\mathbb{F}} b \triangleq a \oplus (-b)$, and $a \otimes_{\mathbb{F}} b \triangleq \lceil a \times b \rfloor_{\mathbb{F}}$. We use $a \oplus b$, $a \ominus b$, $a \otimes b$ to denote $a \oplus_{\mathbb{F}} b$, $a \oplus_{\mathbb{F}} b$, $a \otimes_{\mathbb{F}} b$ if $\mathbb{F}$ is clear from the context. Since the addition between floating-point numbers is not associative, the ordering of a summation becomes important. We define the summation operation $\bigoplus$ of a sequence of floating-point numbers $x_1, x_2, \ldots$ inductively as follows: for $\bigoplus_{i=1}^{1} x_i \triangleq x_1$,

$$\bigoplus_{i=1}^{n} x_i \triangleq \left(\bigoplus_{i=1}^{n-1} x_i \right) \oplus x_n.$$

For $y, x_1, x_2, \ldots, \in \mathbb{F}$, we also define the order of operation as

$$y \oplus \bigoplus_{i=1}^{n} x_i \triangleq \left(y \oplus \bigoplus_{i=1}^{n-1} x_i \right) \oplus x_n,$$

and

$$y \ominus \bigoplus_{i=1}^{n} x_i \triangleq \left(y \ominus \bigoplus_{i=1}^{n-1} x_i \right) \ominus x_n.$$

Using $\bigoplus$, we define an affine transformation under floating-point arithmetic as follows: for $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{F}^n$, $\mathbf{w} = (w_1, \ldots, w_k) \in \mathbb{F}^k$, and $\mathcal{I} = \{i_1, \ldots, i_k\} \subset [n+1]$ with $i_1 < \cdots < i_k$,

$$\mathrm{aff}_{\mathbb{F}}(\mathbf{x}, \mathbf{w}, \mathcal{I}) \triangleq \bigoplus_{j=1}^{k} (w_j \otimes z_{i_j}),$$

where $(z_1, \ldots, z_{n+1}) = (x_1, \ldots, x_n, 1)$.

2.4. Neural networks

Let $\mathbb{F}$ be a set of floating-point numbers, $L \in \mathbb{N}$ be the number of layers, $N_0 = d$ and $N_L = 1$ be the input and output dimensions, $N_1, \ldots, N_{L-1} \in \mathbb{N}$ be numbers of hidden neurons, and $\mathcal{I}_{l,i} = \{\iota_{l,i,1}, \ldots, \iota_{l,i,|\mathcal{I}_{l,i}|}\} \subset [N_{l-1} + 1]$ be a set of indices that characterizes the affine map used for computing an input to the ith hidden neuron in the layer l. Let $N = \sum_{i=1}^{L-1} N_l$ be the total number of hidden neurons and $I = \sum_{l=1}^{L} \sum_{i=1}^{N_l} |\mathcal{I}_{l,i}|$ be the total number of free parameters. Under this setup, we define a neural network $f_{\theta,\mathcal{I}}(\cdot; \mathbb{F}) : \mathbb{F}^d \to \mathbb{F}$ parameterized by $\theta \in \mathbb{F}^I$ with $\mathcal{I} \triangleq (\mathcal{I}_{1,1}, \ldots, \mathcal{I}_{1,N_l}, \ldots, \mathcal{I}_{L,1}, \ldots, \mathcal{I}_{1,N_L})$ via the following recursive relationship: for each $l \in [L]$,

$$\begin{aligned} & f_{\theta,\mathcal{I}}(\mathbf{x}; \mathbb{F}) \triangleq \phi_L(\mathbf{x}; \mathbb{F}), \\ & \phi_l(\mathbf{x}; \mathbb{F}) \triangleq \left(\phi_{l,1}(\mathbf{x}; \mathbb{F}), \ldots, \phi_{l,N_l}(\mathbf{x}; \mathbb{F}) \right), \\ & \phi_{l,i}(\mathbf{x}; \mathbb{F}) \triangleq \mathrm{aff}_{\mathbb{F}}\left(\psi_{l-1}(\mathbf{x}; \mathbb{F}), \mathbf{w}_{l,i}, \mathcal{I}_{l,i} \right) = \bigoplus_{j=1}^{|\mathcal{I}_{l,i}|} (w_{l,i,j} \otimes \psi_{l-1}(\mathbf{x}; \mathbb{F})_{\iota_{l,i,j}}), \\ & \psi_l(\mathbf{x}; \mathbb{F}) \triangleq \left(\sigma(\phi_{l,1}(\mathbf{x}; \mathbb{F})), \ldots, \sigma(\phi_{l,N_l}(\mathbf{x}; \mathbb{F})) \right), \end{aligned} \tag{2}$$

where $\psi_0(\mathbf{x}; \mathbb{F}) \triangleq \mathbf{x}$, $\sigma : \mathbb{F} \to \mathbb{F}$ denotes a pointwise activation function, and θ denote the concatenations of all $w_{l,i,k}$, i.e., $\theta \in \mathbb{F}^I$. In the definition of neural networks, we note that $\phi_{l,i}(\mathbf{x}; \mathbb{F})$ denotes the mapping of the input $\mathbf{x}$ to the feature of the ith neuron at the lth layer *before* applying the activation function of the lth layer, and $\psi_{l,i}(\mathbf{x}; \mathbb{F})$ denotes the mapping of the input $\mathbf{x}$ to the feature of the ith neuron at the lth layer *after* applying the activation function of the lth layer. See Fig. 1 for an illustration of the neural network example.

The network defined in Eq. (2) has L layers, N (hidden) neurons, and I parameters, where each layer l has $\sum_{i=1}^{N_l} |\mathcal{I}_{l,i}|$ free parameters. Note that if $\mathcal{I}_{l,i} = [N_{l-1} + 1]$ for all l, i, then the network is fully-connected. For notational simplicity, we often omit $\mathcal{I}$ and use f_θ. We say a neural network f_θ is a σ *network* if f_θ uses σ as its activation function in Eq. (2) for all layers $l \in [L]$. Note that within our network architecture, every input, output, parameter of the network, and number occurring during the intermediate calculation is in $\mathbb{F}$. Furthermore, all operations performed adhere to floating-point arithmetic.

2.5. Memorization

For a set of floating-point numbers $\mathbb{F}$, $\mathcal{Z} \subset \mathbb{F}^d$, and $\mathcal{Y} \subset \mathbb{F}$, we say a neural network $f_\theta(\cdot; \mathbb{F}) : \mathbb{F}^d \to \mathbb{F}$ of I parameters can *memorize any set of n pairs in* $\mathcal{Z} \times \mathcal{Y}$ if for any $\{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_n, y_n)\} \subset \mathcal{Z} \times \mathcal{Y}$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists a parameter configuration $\theta \in \mathbb{F}^I$ such that $f_\theta(\mathbf{z}_i; \mathbb{F}) = y_i$ for all $i \in [n]$.

2.6. Universal approximation

For a set of floating-point numbers $\mathbb{F}$, a compact set $\mathcal{X} \subset \mathbb{R}^d$, and $\mathcal{Y} \subset \mathbb{R}$, we say σ networks can *universally approximate* $C(\mathcal{X}, \mathcal{Y})$ under $\mathbb{F}$ if for each $f^* \in C(\mathcal{X}, \mathcal{Y})$ and $\varepsilon > 0$, there exists a σ network $f_\theta(\cdot; \mathbb{F}) : \mathbb{F}^d \to \mathbb{F}$ such that

$$|f_\theta(\mathbf{x}; \mathbb{F}) - f^*(\mathbf{x})| \leq |f^*(\mathbf{x}) - \lceil f^*(\mathbf{x}) \rfloor_{\mathbb{F}}| + \varepsilon,$$

for all $\mathbf{x} \in \mathcal{X} \cap \mathbb{F}^d$. As we described in Section 1.1, the term $|f^*(\mathbf{x}) - \lceil f^*(\mathbf{x}) \rfloor_{\mathbb{F}}|$ is an intrinsic error from the representation of floating-point numbers; one cannot obtain a smaller error than this.

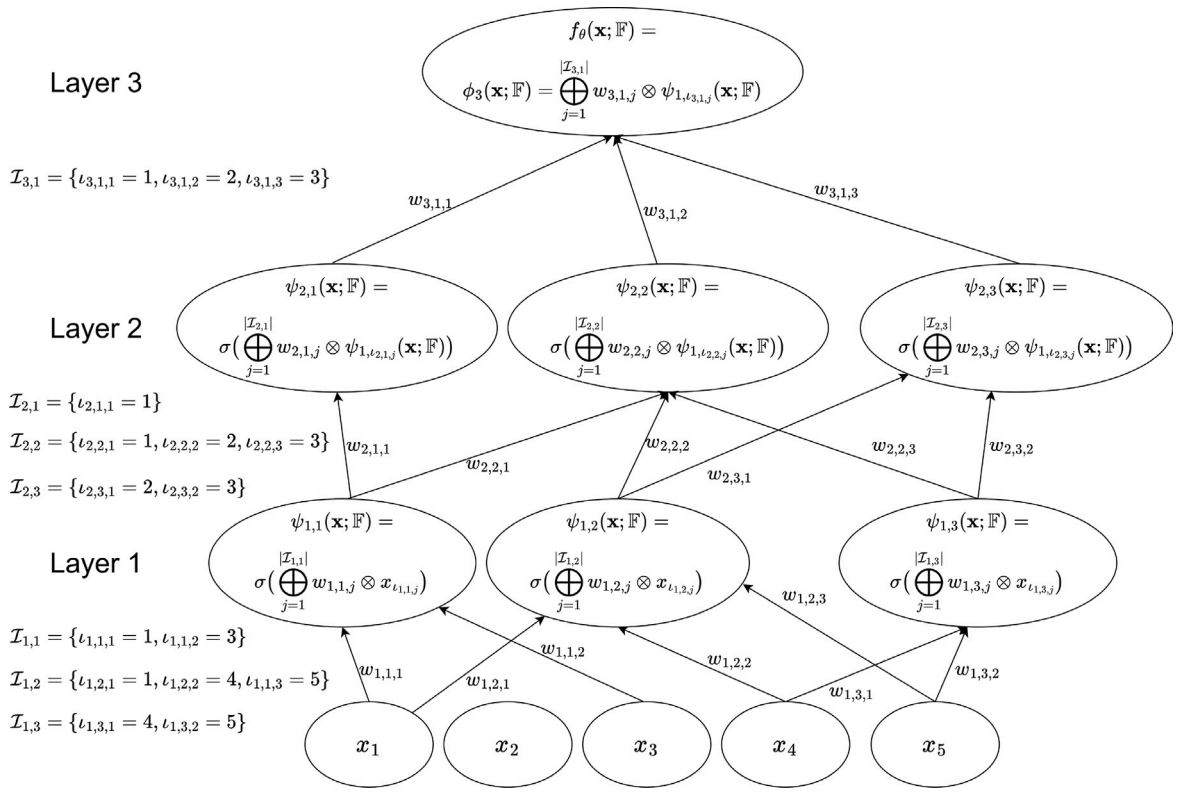

Fig. 1. An illustration of a neural network is presented in Eq. (2). In this example, we set the parameters as follows: $d = 5$, $L = 3$, and $N_1 = N_2 = 3$.

3. Expressive power of neural networks under $\mathbb{F}_p$

We are now ready to present our main results on the universal approximation and memorization properties of neural networks under floating-point inputs, parameters, and operations. In particular, we first introduce our results for $\mathbb{F}_p$ in Section 3, and then extend these results to $\mathbb{F}_{p,q}$ in Section 4. We defer the proofs of all results in this section to Section 5.

3.1. Step *network results*

To show our memorization and universal approximation results for Step networks under $\mathbb{F}_p$, we explicitly construct Step networks that can memorize arbitrary n pairs or universally approximate a target function. Here, our Step network constructions consist of indicator functions for d-dimensional cubes and affine transformations, where such multi-dimensional indicator functions can be implemented by a single Step network architecture as in the following lemma. The proof of Lemma 1 is presented in Section 5.2.

Lemma 1. *For any $p \in \mathbb{N}$ and $d \in [2^p]$, there exists a* Step *network $f_\theta(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 3 layers and $6d+2$ parameters that satisfies the following: for any $\alpha = (\alpha_1, \dots, \alpha_d), \beta = (\beta_1, \dots, \beta_d) \in \mathbb{F}_p^d$ with $\alpha_i < \beta_i$ for all $i \in [d]$, there exists $\theta_{\alpha,\beta} \in \mathbb{F}_p^{6d+2}$ such that*

$$f_{\theta_{\alpha,\beta}}(\mathbf{x};\mathbb{F}_p) = \mathbb{1}\left[\mathbf{x} \in \prod_{i=1}^{d} [\alpha_i, \beta_i]\right],$$

for all $\mathbf{x} \in [0,1]^d$.

Lemma 1 shows that if the input dimension d smaller than or equal to 2^p, a three-layer Step network of $O(1)$ parameters can represent any indicator function for a d-dimensional cube under $\mathbb{F}_p$. We note that the parameter $\theta_{\alpha,\beta}$ of this network is a function of the cube $\Pi_{i=1}^d [\alpha_i, \beta_i]$ in the indicator function.

Using Lemma 1, we show the existence of a Step network $f_\theta(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ such that for any $D = \{(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_n, y_n)\} \subset \mathbb{F}_p^d \times \mathbb{F}_p$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D satisfying

$$f_{\theta_D}(\mathbf{x};\mathbb{F}_p) = \sum_{i=1}^{n} y_i \otimes \mathbb{1}\left[\mathbf{x} \in \prod_{j=1}^{d} [z_{ij}, z_{ij}]\right].$$

Such a result is formally stated in the following theorem; its proof is presented in Section 5.3.

Theorem 2. *For any $p \in \mathbb{N}$ and $d \in [2^p]$, there exists a* Step *network $f_\theta(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 3 layers and $6dn + 2n$ parameters satisfying the following: for any $D = \{(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_n, y_n)\} \subset \mathbb{F}_p^d \times \mathbb{F}_p$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D such that*

$$f_{\theta_D}(\mathbf{z}_i) = y_i, \quad \forall i \in [n].$$

Theorem 2 shows that even under floating-point operations, Step networks can successfully memorize finite datasets represented by floating-point numbers in $\mathbb{F}_p$. Furthermore, it states that $O(n)$ parameters are sufficient for Step networks to memorize arbitrary n pairs under $\mathbb{F}_p$; this coincides with the real-valued parameters and exact operation case (Baum, 1988; Huang & Babri, 1998; Vershynin, 2020; Yun et al., 2019) that are known to be tight up to a logarithmic multiplicative factor for networks of $O(1)$ layers using piecewise linear activation functions (Bartlett, Harvey, Liaw, & Mehrabian, 2019). Namely, the required number of parameters for memorization under $\mathbb{F}_p$ does not decrease compared to existing results assuming real operations.

We next show that Step networks can universally approximate $C([0,1]^d, \mathbb{R})$ under $\mathbb{F}_p$. See Section 5.5 for the proof of Theorem 3.

Theorem 3. *For any $p \in \mathbb{N}$, $d \in [2^p]$, $f^* \in C([0,1]^d, \mathbb{R})$, and $\varepsilon > 0$, there exists a* Step *network $f_\theta(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 3 layers and at most $(6d+2)K^d$ parameters where $K = \min\{k \in \mathbb{N} : (\omega_{f^*}^{-1}(\varepsilon))^{-1} \leq k\}$ such that*

$$|f_\theta(\mathbf{x};\mathbb{F}_p) - f^*(x)| \leq |f^*(\mathbf{x};\mathbb{F}_p) - \lceil f^*(x) \rfloor| + \varepsilon, \quad \forall \mathbf{x} \in [0,1]^d \cap \mathbb{F}_p^d.$$

Theorem 3 states that given a target continuous function $f^* : [0,1]^d \to \mathbb{R}$ and $\varepsilon > 0$, $O(\omega_{f^*}^{-1}(\varepsilon)^{-d})$ parameters are sufficient for approximating f^* in $|f^*(x;\mathbb{F}_p) - \lceil f^*(x)\rfloor| + \varepsilon$ error; this result easily generalizes to an arbitrary compact domain in $\mathbb{R}^d$ instead of $[0,1]^d$. The number of parameters in the network in Theorem 3 is similar to existing results under real-valued parameters and the exact operations for ReLU networks of $O(1)$ layers: $(\omega_{f^*}^{-1}(\Theta(\varepsilon)))^{-d}$ parameters are necessary and sufficient for approximation in ε error (Yarotsky, 2018).

The error bound in Theorem 3 contains an additional term $|f^*(x;\mathbb{F}_p) - \lceil f^*(x)\rfloor|$ compared to the classical universal approximation results. This term corresponds to an intrinsic error arising from the floating-point representation; one cannot achieve a smaller error than this. However, if the domain of the target function contains only finite numbers in $\mathbb{F}_p^d$, then we can approximate this function in the intrinsic error without an additional error term ε as in the following corollary of our memorization result (Theorem 2). We also note that the domain $[0,1]^d \cap \mathbb{F}_p^d$ considered in Theorem 3 is an infinite set, and hence, cannot directly apply this corollary.

Corollary 4. *For any $p \in \mathbb{N}$, $d \in [2^p]$, a compact set $\mathcal{K} \subset \mathbb{R}^d$ such that $|\mathcal{K} \cap \mathbb{F}_p^d| < \infty$, $f^* \in C(\mathcal{K},\mathbb{R})$, and $\varepsilon > 0$, there exists a* Step *network $f(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 3 layers and $6d|\mathcal{K} \cap \mathbb{F}_p^d| + 2|\mathcal{K} \cap \mathbb{F}_p^d|$ parameters such that*

$$|f_\theta(\mathbf{x};\mathbb{F}_p) - f^*(\mathbf{x})| = |f^*(\mathbf{x};\mathbb{F}_p) - \lceil f^*(\mathbf{x})\rfloor|, \quad \forall \mathbf{x} \in [0,1]^d \cap \mathbb{F}_p^d.$$

3.2. ReLU *network results*

We next analyze the expressive power of ReLU networks under $\mathbb{F}_p$. Given the Step network results, a naïve approach can be implementing Step function using two ReLU activation functions. For example, for any two consecutive numbers $z^- < z$ in $\mathbb{F}_p$,

$$\mathbb{1}[x \geq z] = \mathrm{ReLU}\left(\frac{1}{z - z^-}(x - z^-)\right) - \mathrm{ReLU}\left(\frac{1}{z - z^-}(x - z)\right), \tag{3}$$

for all $x \in \mathbb{F}_p$ under *exact operations*. However, if we consider floating-point operations in $\mathbb{F}_p$, RHS in Eq. (3) does not represent the indicator function anymore. For example, consider the indicator function $\mathbb{1}[x \geq 1]$, which can be exactly implemented under the exact mathematical operations for all inputs $x \in \mathbb{F}_p$ by

$$f(x;\mathbb{R}) = \mathrm{ReLU}\big((x \times 2^p) - (1-u) \times 2^p\big) - \mathrm{ReLU}\big((x \times 2^p) - 2^p\big),$$

where $u = 2^{-p}$. Under operations in $\mathbb{F}_p$, however, $f(x;\mathbb{F}_p)$ with the following form cannot represent $\mathbb{1}[x \geq 1]$ for some $x \in \mathbb{F}_p$:

$$f(x;\mathbb{F}_p) = \mathrm{ReLU}\big((x \times 2^p) \ominus ((1-u) \times 2^p)\big) \ominus \mathrm{ReLU}\big((x \times 2^p) \ominus 2^p\big).$$

Specifically, the output values of $f(x;\mathbb{F}_p)$ can be exactly characterized as follows: for $x \in \mathbb{F}_p$,

$$f(x;\mathbb{F}_p) = \begin{cases} 0 & \text{if } x < 1, \\ 1 & \text{if } 1 \leq x < 1.1 \times 2^1, \\ 1 + (-1)^{n_x+1} & \text{if } 1.1 \times 2^1 \leq x < 1.01 \times 2^2, \\ 0 & \text{if } x \geq 1.01 \times 2^2, \end{cases} \tag{4}$$

where $n_x = (x - 1.1 \times 2^1) \times 2^{p-1} \in \mathbb{N}$ for $1.1 \times 2^1 \leq x < 1.01 \times 2^2$. Here, one can observe that $f(x;\mathbb{F}_p)$ becomes 0 for $x \geq 1.01 \times 2^2$; for $1.1 \times 2^1 \leq x < 1.01 \times 2^2$, it oscillates between 0 and 2. Hence, unlike the exact operation case, deriving the ReLU network results from Step network results is non-trivial under floating-point operations. We present the formal derivation of Eq. (4) in Section 5.4.

Nevertheless, we observe that Eq. (3) under $\mathbb{F}_p$ behaves like an indicator function for x close to z. Using this property, we implement an indicator function as follows: using a ReLU network, we first map an input x to x' in some local neighborhood of z such that $x' \geq z$ if and only if $x \geq z$; we then apply Eq. (3) to x'. This construction of an indicator function via ReLU networks enables us to show memorization and universal approximation properties of ReLU networks as in the following theorems. The proofs of Theorems 5 and 6 are presented in Sections 5.6 and 5.7, respectively.

Theorem 5. *For any $p \in \mathbb{N}$ and $d \in [2^p]$, there exists a* ReLU *network $f_\theta(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 4 layers and $20dn + 2n$ parameters satisfying the following: for any $D = \{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_n, y_n)\} \subset \mathbb{F}_p^d \times \mathbb{F}_p$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D such that*

$$f_{\theta_D}(\mathbf{z}_i) = y_i, \quad \forall i \in [n].$$

Theorem 6. *For any $p \in \mathbb{N}$, $d \in [2^p]$, $f^* \in C([0,1]^d,\mathbb{R})$, and $\varepsilon > 0$, there exists a* Step *network $f(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 4 layers and $(20d+2)K^d$ parameters where $K = \min\{k \in \mathbb{N} : (\omega_{f^*}^{-1}(\varepsilon))^{-1} \leq k\}$ such that*

$$|f_\theta(x;\mathbb{F}_p) - f^*(x)| \leq |f^*(x;\mathbb{F}_p) - \lceil f^*(x)\rfloor| + \varepsilon.$$

We note that the numbers of parameters in networks in Theorems 5 and 6 coincide with that of Step network results (Theorems 2 and 3) up to constant multiplicative factors, i.e., they also correspond to the necessary and sufficient number of parameters in classical results under exact operations. Further, the following analogue of Corollary 4 also holds for ReLU networks.

Corollary 7. *For any $p \in \mathbb{N}$, $d \in [2^p]$, a compact set $\mathcal{K} \subset \mathbb{R}^d$ such that $|\mathcal{K} \cap \mathbb{F}_p^d| < \infty$, $f^* \in C(\mathcal{K},\mathbb{R})$, and $\varepsilon > 0$, there exists a* ReLU *network $f(\cdot;\mathbb{F}_p) : \mathbb{F}_p^d \to \mathbb{F}_p$ of 4 layers and $20d|\mathcal{K} \cap \mathbb{F}_p^d| + 2|\mathcal{K} \cap \mathbb{F}_p^d|$ parameters such that*

$$|f_\theta(\mathbf{x};\mathbb{F}_p) - f^*(\mathbf{x})| = |f^*(\mathbf{x};\mathbb{F}_p) - \lceil f^*(\mathbf{x})\rfloor|, \quad \forall \mathbf{x} \in [0,1]^d \cap \mathbb{F}_p^d.$$

4. Expressive power of neural networks under $\mathbb{F}_{p,q}$

We now consider a more practical setup: when all computations are done in $\mathbb{F}_{p,q}$ (i.e., the set of floating-point numbers with p-bit significand and q-bit exponent), where $p, q \in \mathbb{N}$ satisfy $q \geq 5$ and $4 \leq p \leq 2^{q-2} + 2$. As noted in Section 2.2, this class of $\mathbb{F}_{p,q}$ covers many popular floating-point formats such as those defined in the IEEE 754 standard (IEEE Computer Society, 2019) and bfloat16 (Abadi et al., 2016). Specifically, we demonstrate that Step networks and ReLU networks under $\mathbb{F}_{p,q}$ also exhibit the memorization and universal approximation properties with the same number of parameters as in the $\mathbb{F}_p$ case. The proofs of all results in this section are presented in Section 6.

4.1. Step *network results*

For Step networks, we can prove similar results as in the $\mathbb{F}_p$ case; we construct an indicator function for a high-dimensional cube and derive the following memorization and universal approximation results. Namely, Step networks are expressive under realistic floating-point operations. The proofs of Theorems 8 and 9 are presented in Sections 6.3 and 6.4, respectively.

Theorem 8. *For any $d \in [2^p]$, there exists a* Step *network $f_\theta(\cdot;\mathbb{F}_{p,q}) : \mathbb{F}_{p,q}^d \to \mathbb{F}_{p,q}$ of 3 layers and $6dn + 2n$ parameters satisfying the following: for any $D = \{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_n, y_n)\} \subset \mathbb{F}_{p,q}^d \times \mathbb{F}_{p,q}$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D such that*

$$f_{\theta_D}(\mathbf{z}_i) = y_i, \quad \forall i \in [n].$$

Theorem 9. *For any $d \in [2^p]$, $f^* \in C([0,1]^d,\mathbb{R})$, and $\varepsilon \geq 0$, there exists a* Step *network $f(\cdot;\mathbb{F}_{p,q}) : \mathbb{F}_{p,q}^d \to \mathbb{F}_{p,q}$ of 3 layers and at most $(6d+2)K^d$ parameters where $K = \min\{k \in \mathbb{N} : \delta^{-1} \leq k\}$, $\delta = \max\{\eta, \omega_{f^*}^{-1}(\varepsilon)\}$ such that*

$$|f_\theta(\mathbf{x};\mathbb{F}_{p,q}) - f^*(x)| \leq |f^*(\mathbf{x};\mathbb{F}_{p,q}) - \lceil f^*(x)\rfloor| + \varepsilon, \quad \forall \mathbf{x} \in [0,1]^d \cap \mathbb{F}_{p,q}^d.$$

Here, $\eta \triangleq 2^{-p+e_{min}}$ denotes the smallest positive floating-point number as we defined in Section 2.2.

Unlike Theorem 3 under $\mathbb{F}_p$, we allow $\varepsilon = 0$ in Theorem 9. This is because $[0,1]^d \cap \mathbb{F}_{p,q}^d$ is always finite, and hence, we can achieve $\varepsilon = 0$ using the memorization result (Theorem 8) with a finite number of parameters as in Corollary 4. We note that the number of parameters in Theorems 8 and 9 coincide with results under $\mathbb{F}_p$ up to a constant multiplicative factor.

4.2. ReLU network results

While Step network results for $\mathbb{F}_{p,q}$ (Theorems 8 and 9) can be shown as in the $\mathbb{F}_p$ case, ReLU network results do not naturally follow from $\mathbb{F}_p$ to $\mathbb{F}_{p,q}$ due to the overflow (and underflow) in $\mathbb{F}_{p,q}$. For example, recall Eq. (3)

$$\mathbb{1}[x \geq z] = \mathrm{ReLU}\left(\frac{1}{z-z^-}(x-z^-)\right) - \mathrm{ReLU}\left(\frac{1}{z-z^-}(x-z)\right),$$

and consider operations under $\mathbb{F}_{p,q}$. If z is small enough (e.g., $z = 2^{e_{min}}$) and x is large enough (e.g., $x = 2^{e_{max}}$). Then, the computation of $\frac{1}{z-z^-}(x-z^-)$ under $\mathbb{F}_{p,q}$ can incur an overflow. However, by carefully analyzing floating-point operations, we can implement the indicator function using a ReLU network of $O(1)$ parameters (see Lemma 27); using this we can also show that memorization and universal approximation are possible under $\mathbb{F}_{p,q}$. We present the proofs of Theorems 10 and 11 in Sections 6.5 and 6.6, respectively.

Theorem 10. *For any $d \in [2^p]$ and for $\kappa = (2-u) \times 2^{-2-p+e_{max}}$, there exists a ReLU network $f_\theta(\cdot;\mathbb{F}_{p,q}) : (\mathbb{F}_{p,q} \cap [-\kappa,\kappa])^d \to \mathbb{F}_{p,q}$ of 4 layers and $20dn + 5n$ parameters satisfying the following: for any $\mathcal{D} = \{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_n, y_n)\} \subset (\mathbb{F}_{p,q} \cap [-\kappa,\kappa])^d \times \mathbb{F}_{p,q}$ with $\mathbf{z}_i \neq \mathbf{z}_j$ for all $i \neq j$, there exists θ_D such that*

$$f_{\theta_D}(\mathbf{z}_i) = y_i, \quad \forall i \in [n].$$

Furthermore, computing $f_\theta(\mathbf{x};\mathbb{F}_{p,q})$ does not incur overflow for all $\mathbf{x} \in \mathbb{F}_{p,q}^d$ satisfying $\|\mathbf{x}\|_\infty \leq (2-u) \times 2^{-3+2^{q-2}}$.

Theorem 11. *For any $d \in [2^p]$, $f^* \in C([0,1]^d, \mathbb{R})$, and $\varepsilon \geq 0$, there exists a ReLU network $f(\cdot;\mathbb{F}_{p,q}) : \mathbb{F}_{p,q}^d \to \mathbb{F}_{p,q}$ of 4 layers and at most $(20d+2)K^d$ parameters where $K = \min\{k \in \mathbb{N} : \delta^{-1} \leq k\}$, $\delta = \max\{\eta, \omega_{f^*}^{-1}(\varepsilon)\}$ such that*

$$|f(\mathbf{x};\mathbb{F}_{p,q}) - f^*(\mathbf{x})| \leq |f^*(\mathbf{x};\mathbb{F}_{p,q}) - \lceil f^*(x) \rfloor| + \varepsilon, \quad \forall \mathbf{x} \in [0,1]^d \cap \mathbb{F}_{p,q}^d.$$

Here, $\eta \triangleq 2^{-p+e_{min}}$ denotes the smallest positive floating-point number as we defined in Section 2.2.

In Theorem 10, the supremum norm of an input to memorize is bounded by $\kappa = (2-u) \times 2^{-2-p+e_{max}}$; we use this condition to bypass the overflow during intermediate computations in our network construction.

Remark 12. To check whether the condition in Theorem 10 can be satisfied in practice, we evaluate the $B_x \triangleq (2-u) \times 2^{-3+2^{q-2}}$ and $\kappa \triangleq (2-u) \times 2^{-2-p+e_{max}}$ for popular floating-point formats including IEEE 754 formats and bfloat16. For the 16-bit half-precision format, we have $B_x = 63.969$ and $\kappa = 15.992$. For the 32-bit single-precision format, we have $B_x = 4.6117 \times 10^{18}$ and $\kappa = 1.01141 \times 10^{31}$. For the 64-bit double-precision format, we have $B_x = 3.3520 \times 10^{153}$ and $\kappa = 9.9980 \times 10^{291}$. For the 128-bit quadruple-precision format, we have $B_x = 1.3634 \times 10^{2465}$ and $\kappa = 2.8642 \times 10^{4897}$. For the bfloat16 format, we have $B_x = 4.5938 \times 10^{18}$ and $\kappa = 6.6202 \times 10^{35}$. This implies that the condition in Theorem 10 can be often satisfied in many floating-point number formats and inputs except for half-precision. In the 16-bit half-precision format, we need to carefully bound the input data to have the universal approximation property using Theorem 10. On the other hand, bfloat16 format has larger B_x and κ, i.e., bfloat16 can be better than the half-precision format in terms of the universal approximation.

In Theorem 11, we also allow $\varepsilon = 0$ as in Theorem 9. As in our previous results, Theorems 10 and 11 use $O(n)$ parameters and $O((\omega_{f^*}^{-1}(\varepsilon))^{-1})$ parameters, respectively.

5. Proofs of results under $\mathbb{F}_p$

5.1. Technical lemmas

We present several technical lemmas regarding the computation of operations in $\mathbb{F}_p$. Lemma 13 is the well-known Sterbenz's lemma (Sterbenz, 1973, Theorem 4.3.1) (also in Muller, Brunie, de Dinechin, Jeannerod, Joldes, Lefevre, Melquiond, Revol, and Torres (2018, Lemma 4.1)). In particular, Lemmas 14–17 have $\mathbb{F}_{p,q}$ versions analogous in Section 6.

For $x \in \mathbb{F}_p$, we represent x as

$$x = s_x \times a_x \times 2^{e_x}, \ s_x \in \{-1,1\}, a_x = 1.x_1 \cdots x_p.$$

For $x \in \mathbb{F}_p$, we define $\mu(x)$ as

$$\mu(x) \triangleq \inf\{m \in \mathbb{Z} : x \times 2^{-m} \in \mathbb{Z}\}.$$

Note that if $x \neq 0$, we can represent x as

$$x = (n_x \times 2^{-e_x+\mu(x)}) \times 2^{e_x}, \tag{5}$$

for $n_x = x \times 2^{-\mu(x)} \in \mathbb{N}$ with $2^{e_x-\mu(x)} \leq n_x < 2^{1+e_x-\mu(x)}$.

For $x \in \mathbb{R}$, we define the ceiling function $\lceil x \rceil_{\mathbb{Z}}$ as

$$\lceil x \rceil_{\mathbb{Z}} = \min\{n \in \mathbb{Z} : n \geq x\}.$$

We now formally introduce the statements of technical lemmas used for proving our results under $\mathbb{F}_p$.

Lemma 13 (*Sterbenz's Lemma*). *Let $x, y \in \mathbb{F}_p$. If $0 \leq y/2 \leq x \leq y$, then $x \ominus y$ and $y \ominus x$ are exact.*

Lemma 14. *Let $x = n \times 2^m$ for some $n \in \mathbb{N}$, $m \in \mathbb{Z}$. If $0 < n < 2^{1+p}$, then x is representable by $\mathbb{F}_p$.*

Proof. Since $0 \leq n < 2^{1+p}$, there exists $c_0 \in \{0\} \cup [p]$ such that $2^{p-c_0} \leq n < 2^{1+p-c_0}$. Note that x has the following representation in $\mathbb{F}_p$,

$$x = (n \times 2^{-p+c_0}) \times 2^{p-c_0+m}, \ 1 \leq n \times 2^{-p+c_0} < 2.$$

Then we express $n \times 2^{-p+c_0}$ as

$$n \times 2^{-p+c_0} = 1.\underbrace{w_1 \cdots w_{p-c_0}}_{p-c_0 \text{ times}},$$

for some $w_1, \ldots, w_{p-c_0} \in \{0,1\}$. Therefore x is representable by $\mathbb{F}_p$. □

Lemma 15. *Let $x, y \in \mathbb{F}_p$ and $s_x = s_y$ and $e_x \leq e_y$. If $\mu(x) \geq e_y - p$, then $x \ominus y$ and $y \ominus x$ are exact. In addition, if $|x+y| \leq 2^{1+e_y}$, then $x \oplus y$ is exact.*

Proof. Let $k \triangleq \mu(x) - e_y + p \geq 0$. As described in Eq. (5), we can represent x and y as

$$x = (n_x \times 2^{-e_x+\mu(x)}) \times 2^{e_x} = n_x \times 2^{\mu(x)}, \ 2^{e_x-\mu(x)} \leq n_x < 2^{1+e_x-\mu(x)},$$
$$y = (n_y \times 2^{-p+c_y}) \times 2^{e_y} = n_y \times 2^{-p+e_y}, \ 2^{p-c_y} \leq n_y < 2^{1+p},$$

for some $n_x, n_y \in \mathbb{N}$. Since $k \geq 0$, we have

$$x = (2^k n_x) \times 2^{-p+e_y}, \ 2^{p-(e_y-e_x+c_y)} \leq 2^k n_x < 2^{1+p-(e_y-e_x)}.$$

Therefore for $n' = n_y - 2^k n_x \in \mathbb{N}$, we have

$$y - x = n' \times 2^{-p+e_y}, \ 2^p - 2^{1+p-(e_y-e_x)} < n' < 2^{1+p} - 2^{p-(e_y-e_x)},$$

which leads to

$$-2^p < n' < 2^{p+1} - 1.$$

Since $|n'| = 0$ or $|n'| < 2^{1+p} - 1$, by Lemma 14, $y - x = n' \times 2^{-p+e_y}$ is representable by $\mathbb{F}_p$. This ensures that $y \ominus x$ is exact.

Now suppose $|x + y| \le 2^{1+e_y}$. Then for $n'' = n_x + n_y \in \mathbb{N}$, we have

$$x + y = n'' \times 2^{-p+e_y},\ 2^{p-(e_y-e_x)} + 2^p \le n'' < 2^{1+p-(e_y-e_x)} + 2^{1+p}.$$

Since $|x + y| \le 2^{1+e_y}$, we have $n'' \le 2^{1+p}$. If $|n''| = 0$ or $|n''| < 2^{1+p}$, by Lemma 14, $x + y = n'' \times 2^{-p+e_y}$ is representable by $\mathbb{F}_p$. If $|n''| = 2^{1+p}$, $x + y = 2^{1+e_y}$ is obviously representable by $\mathbb{F}_p$. Therefore, $x \oplus y$ is exact. □

Lemma 16. *In $\mathbb{F}_p$, if $x, y \in [2^{1+p}]$, $x \ominus y$ and $y \ominus x$ are exact. In addition if $x, y \in [2^p]$, then $x \oplus y$ is exact.*

Proof. Without loss of generality, suppose $x \le y$ for $x, y \in [2^{1+p}]$. Since $y - x < 2^{1+p}$, $y - x$ is representable by $\mathbb{F}_p$ by Lemma 14, ensuring that $x \ominus y$ and $y \ominus x$ are exact.

In addition, suppose $x, y \in [2^p]$ and $x \le y$. Since $\mu(x) \ge 0$ and $e_y \le p$, we have $\mu(x) \ge -p + e_y$. Since $|x + y| < 2^{1+p}$, $x \oplus y$ is exact by Lemma 15. □

Lemma 17. *Let $x, y \in \mathbb{F}_p$ and $e_x \le e_y$. If $\mu(x) \le e_y - p - 2$, then $y \oplus x = y \oplus x = y$.*

Proof. Since $|x| \le 2^{-2-p+e_y}$, we have $\lceil x + y \rfloor = \lceil -x + y \rfloor = y$. □

Lemma 18. *For any $d, p \in \mathbb{N}$, let $\alpha_1, \beta_1, \dots, \alpha_d, \beta_d \in \mathbb{F}_p$ such that $\alpha_i \le \beta_i$ for all $i \in [d]$. Then, for any sequence $(\mathbf{x}_i)_{i\in\mathbb{N}}$ in $([\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d]) \cap \mathbb{F}_p^d$, there exists a subsequence of $(\mathbf{x}_i)_{i\in\mathbb{N}}$ that converges to some $\mathbf{z}^* \in ([\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d]) \cap \mathbb{F}_p^d$.*

Proof. Let $\mathcal{S} = [\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d]$. Since $\mathcal{S}$ is compact, there exists a convergent subsequence $(\mathbf{z}_i)_{i\in\mathbb{N}}$ of $(\mathbf{x}_i)_{i\in\mathbb{N}}$ and $(\mathbf{z}_i)_{i\in\mathbb{N}}$ converges to some point, say $\mathbf{z}^* = (z_1^*, \dots, z_d^*)$, in $\mathcal{S}$. However, if $z_j^* \notin \mathbb{F}_p$ for some $j \in [d]$, then $\inf_{y\in\mathbb{F}_p} |z_j^* - y| > 0$ by the definition of $\mathbb{F}_p$, i.e., $\mathbf{z}_1, \mathbf{z}_2, \dots \in \mathbb{F}_p^d$ cannot converge to $\mathbf{x}^*$. Hence, $\mathbf{z}^* \in \mathcal{S} \cap \mathbb{F}_p^d$ and this completes the proof. □

Lemma 19. *For any $d, p \in \mathbb{N}$, let $\alpha_1, \beta_1, \dots, \alpha_d, \beta_d \in \mathbb{F}_p$ such that $\alpha_i \le \beta_i$ for all $i \in [d]$. Then, for any continuous function $f : [\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d] \to \mathbb{R}$, there exists $\mathbf{x}^* \in ([\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d]) \cap \mathbb{F}_p^d$ such that*

$$|f(\mathbf{x}^*) - \lceil f(\mathbf{x}^*) \rfloor| = \inf_{\mathbf{x}\in([\alpha_1,\beta_1]\times\cdots\times[\alpha_d,\beta_d])\cap\mathbb{F}_p^d} |f(\mathbf{x}) - \lceil f(\mathbf{x}) \rfloor|.$$

Proof. Let $\mathcal{S} = ([\alpha_1, \beta_1] \times \cdots \times [\alpha_d, \beta_d]) \cap \mathbb{F}_p^d$. Suppose for a contradiction that such $\mathbf{x}^*$ does not exist, i.e., there is no $\mathbf{x} \in \mathcal{S}$ such that $f(\mathbf{x}) \in \mathbb{F}_p$. Then, there exists a sequence $(\mathbf{x}_i)_{i\in\mathbb{N}}$ in $\mathcal{S}$ such that $(|f(\mathbf{x}_i) - \lceil f(\mathbf{x}_i) \rfloor|)_{i\in\mathbb{N}}$ is strictly decreasing and

$$|f(\mathbf{x}_i) - \lceil f(\mathbf{x}_i) \rfloor| \to \inf_{\mathbf{x}\in\mathcal{S}} |f(\mathbf{x}) - \lceil f(\mathbf{x}) \rfloor|, \tag{6}$$

as $i \to \infty$. By Lemma 18, there exists a subsequence $(\mathbf{z}_i)_{i\in\mathbb{N}}$ of $(\mathbf{x}_i)_{i\in\mathbb{N}}$ that converges to some point $\mathbf{z}^* \in \mathcal{S}$. Since $f(\mathbf{z}^*) \notin \mathbb{F}_p$, by the definition of $\mathbb{F}_p$, we have

$$f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)} < f(\mathbf{z}^*) < f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)}.$$

Let $\varepsilon = \min\{f(\mathbf{z}^*) - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}, f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)\}$. We consider two cases to show the contradiction: $\varepsilon = (f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)})/2$ and $\varepsilon \ne (f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)})/2$. Suppose $\varepsilon = (f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)})/2$, i.e., $f(\mathbf{z}^*) = (f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} + f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)})/2$. Let $(\mathbf{y}_i)_{i\in\mathbb{N}}$ be a subsequence of $(\mathbf{z}_i)_{i\in\mathbb{N}}$ such that $|f(\mathbf{z}^*) - f(\mathbf{y}_i)| \le \varepsilon$ for all $i \in \mathbb{N}$ and $(|f(\mathbf{z}^*) - f(\mathbf{y}_i)|)_{i\in\mathbb{N}}$ is strictly decreasing; such a subsequence always exist due to the continuity of f and our choice of $(\mathbf{z}_i)_{i\in\mathbb{N}}$. Then, one can observe that $(|f(\mathbf{y}_i) - \lceil f(\mathbf{y}_i) \rfloor|)_{i\in\mathbb{N}}$ is monotonically increasing as $\lceil f(\mathbf{y}_i) \rfloor \in \{f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}, f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)}\}$ and $f(\mathbf{y}_i) \to f(\mathbf{z}^*) = (f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)} + f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)})/2$. Namely, $(|f(\mathbf{y}_i) - \lceil f(\mathbf{y}_i) \rfloor|)_{i\in\mathbb{N}}$ is not strictly decreasing although $(\mathbf{y}_i)_{i\in\mathbb{N}}$ is a subsequence of $(\mathbf{x}_i)_{i\in\mathbb{N}}$. This contradicts the assumption that $(|f(\mathbf{x}_i) - \lceil f(\mathbf{x}_i) \rfloor|)_{i\in\mathbb{N}}$ is strictly decreasing.

We now consider the case that $\varepsilon \ne (f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)})/2$. Without loss of generality, suppose $f(\mathbf{z}^*) - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)} < f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)$. Let

$$\delta = \frac{f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} - f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}}{2} - \varepsilon = \frac{f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)} + f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}}{2} - f(\mathbf{z}^*) > 0,$$

and choose a subsequence $(\mathbf{y}_i')_{i\in\mathbb{N}}$ of $(\mathbf{z}_i)_{i\in\mathbb{N}}$ such that $|f(\mathbf{z}^*) - f(\mathbf{y}_i')| < \delta$ for all $i \in \mathbb{N}$ and $(|f(\mathbf{z}^*) - f(\mathbf{y}_i')|)_{i\in\mathbb{N}}$ is strictly decreasing. Then, since $f(\mathbf{z}^*)$ and $f(\mathbf{y}_i')$ are closer to $f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}$ than $f(\mathbf{z}^*)^{(\ge,\mathbb{F}_p)}$, we have $\lceil f(\mathbf{z}^*) \rfloor = \lceil f(\mathbf{y}_i') \rfloor = f(\mathbf{z}^*)^{(\le,\mathbb{F}_p)}$. Namely, we must have $f(\mathbf{y}_i') \ge f(\mathbf{z}^*)$ for all $i \in \mathbb{N}$ to have strictly decreasing $(|f(\mathbf{y}_i') - \lceil f(\mathbf{y}_i) \rfloor|)_{i\in\mathbb{N}}$. However, this implies that

$$|f(\mathbf{y}_i') - \lceil f(\mathbf{y}_i) \rfloor| \ge |f(\mathbf{z}^*) - \lceil f(\mathbf{z}^*) \rfloor|,$$

for all $i \in \mathbb{N}$, which implies

$$|f(\mathbf{z}^*) - \lceil f(\mathbf{z}^*) \rfloor| = \inf_{\mathbf{x}\in\mathcal{S}} |f(\mathbf{x}) - \lceil f(\mathbf{x}) \rfloor|,$$

from our choice of $(\mathbf{x}_i)_{i\in\mathbb{N}}$ and Eq. (6). This contradicts the assumption that $\mathbf{x}^*$ in the statement of the lemma does not exist and completes the proof. □

Lemma 20. *There exist* ReLU *networks $\psi_\theta(\cdot; \mathbb{F}_p) : \mathbb{F}_p \to \mathbb{F}_p$ of 3 layers and 5 parameters that satisfies the following: for any $z \in \mathbb{F}_p \setminus \{0\}$, there exist $\theta_{1,1,z}, \theta_{1,2,z}, \theta_{2,1,z}, \theta_{2,2,z} \in \mathbb{F}_p^5$ such that for any $x \in \mathbb{F}_p$,*

$$\psi_{\theta_{1,1,z}}(x; \mathbb{F}_p) \oplus \psi_{\theta_{1,2,z}}(x; \mathbb{F}_p) = \mathbb{1}[x \ge z],$$
$$\psi_{\theta_{2,1,z}}(x; \mathbb{F}_p) \oplus \psi_{\theta_{2,2,z}}(x; \mathbb{F}_p) = \mathbb{1}[x \le z].$$

In addition, $\psi_{\theta_{i,1,z}}(x; \mathbb{F}_p), -\psi_{\theta_{i,2,z}}(x; \mathbb{F}_p) \in \{0\} \cup [2^p]$ for all $i \in \{1, 2\}$.

Proof. We first consider $z > 0$. To construct $\mathbb{1}[x \ge z]$, we define a three-layer ReLU network $f_1(x; \mathbb{F}_p)$ as follows:

$$f_1(x; \mathbb{F}_p) = \psi_{\theta_{1,1,z}}(x; \mathbb{F}_p) \oplus \psi_{\theta_{1,2,z}}(x; \mathbb{F}_p),$$
$$\psi_{\theta_{1,1,z}}(x; \mathbb{F}_p) = \phi_{\theta_{1,1,z}}(x; \mathbb{F}_p),\ \psi_{\theta_{1,2,z}}(x; \mathbb{F}_p) = \phi_{\theta_{1,2,z}}(x; \mathbb{F}_p),$$
$$\phi_{\theta_{1,1,z}}(x; \mathbb{F}_p) = 2^{\tilde{c}} \otimes \mathrm{ReLU}\big((-2^{p-e_z} \otimes g(x)) \oplus (2 - a_z - \tilde{u}) \times 2^p\big),$$
$$\phi_{\theta_{1,2,z}}(x; \mathbb{F}_p) = -2^{\tilde{c}} \otimes \mathrm{ReLU}\big((-2^{p-e_z} \otimes g(x)) \oplus (2 - a_z - u) \times 2^p\big),$$

where

$$g(x) = \mathrm{ReLU}(-x \oplus (2 - u) \times 2^{e_z}),$$
$$\tilde{u} = \begin{cases} 0 & \text{if } a_z \ne 1, \\ 2^{-1-p} & \text{if } a_z = 1, \end{cases}$$
$$\tilde{c} = \begin{cases} 0 & \text{if } a_z \ne 1, \\ 1 & \text{if } a_z = 1. \end{cases}$$

Note that $z^- = (a_z - u + \tilde{u}) \times 2^{e_z}$.

If $(2 - u) \times 2^{-1+e_z} \le x \le (2 - 2u) \times 2^{e_z}$, then $-x \oplus (2 - u) \times 2^{e_z}$ is exact by Lemma 13. Therefore, we have

$$g(x) \begin{cases} \ge (2 - u) \times 2^{-1+e_z} & \text{if } x \le (2 - 2u) \times 2^{-1+e_z}, \\ = -x + (2 - u) \times 2^{e_z} & \text{if } (2 - u) \times 2^{-1+e_z} \le x \le (2 - 2u) \times 2^{e_z}, \\ = 0 & \text{if } x \ge (2 - u) \times 2^{e_z}. \end{cases}$$

If $x \le (2 - 2u) \times 2^{-1+e_z}$, since we have

$$(2 - a_z - \tilde{u}) \le (1 - 2^{-1-p}) = (2 - u) \times 2^{-1},$$

and

$$\begin{aligned}(-2^{p-e_z} \otimes g(x)) &\le (-2^{p-e_z} \times (2 - u) \times 2^{-1+e_z}) \\ &= -(2 - u) \times 2^{-1+p} \le -(2 - a_z - \tilde{u}) \times 2^p,\end{aligned}$$

it holds that $\phi_{\theta_{1,1,z}}(x; \mathbb{F}_p) = \phi_{\theta_{1,2,z}}(x; \mathbb{F}_p) = 0$.

If $(2 - u) \times 2^{-1+e_z} \le x \le (2 - 2u) \times 2^{e_z}$, we have $2^{p-e_z} \otimes g(x) = n_1$, where $n_1 = 1, 2, \dots, 2^p - 1$, leading $2^{p-e_z} \otimes g(x) \in [2^p]$. Since

$(2-a_z-\tilde{u})\times 2^p, (2-a_z-u)\times 2^p \in [2^p]$, by Lemma 16, all operations in $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_p)$ and $\phi_{\theta_{1,2,z}}(x;\mathbb{F}_p)$ are exact. Hence

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_p) = 2^{\tilde{c}} \times \mathrm{ReLU}(2^{p-e_z}x - (a_z - u + \tilde{u})\times 2^p),$$
$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_p) = -2^{\tilde{c}} \times \mathrm{ReLU}(2^{p-e_z}x - a_z \times 2^p),$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_p), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_p) \in \{0\}\cup[2^p]$. Therefore

$$f_1(x;\mathbb{F}_p) = \begin{cases} 0 & \text{if } (2-u)\times 2^{-1+e_z} \le x \le z^-, \\ 1 & \text{if } z \le x \le (2-2u)\times 2^{e_z}. \end{cases}$$

If $x \ge (2-u)\times 2^{e_z}$, we have

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_p) = 2^{\tilde{c}}\big((2-a_z-\tilde{u})\times 2^p\big),$$
$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_p) = -2^{\tilde{c}}\big((2-a_z-u)\times 2^p\big),$$
$$f_1(x;\mathbb{F}_p) = \phi_{\theta_{1,1,z}}(x;\mathbb{F}_p) \oplus \phi_{\theta_{1,2,z}}(x;\mathbb{F}_p) = 2^{\tilde{c}}(u-\tilde{u}) = 1,$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_p), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_p) \in \{0\}\cup[2^p]$. Therefore we conclude

$$f_1(x;\mathbb{F}_p) = \mathbb{1}[x \ge z],$$

for all $x \in \mathbb{F}_p$.

We also construct $\mathbb{1}[x \le z]$ in a similar way. To this end, we define $f_2(x;\mathbb{F}_p)$ as

$$f_2(x;\mathbb{F}_p) = \psi_{\theta_{2,1,z}}(x;\mathbb{F}_p) \oplus \psi_{\theta_{2,2,z}}(x;\mathbb{F}_p),$$
$$\psi_{\theta_{2,1,z}}(x;\mathbb{F}_p) = \phi_{\theta_{2,1,z}}(x;\mathbb{F}_p),\ \psi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = \phi_{\theta_{2,2,z}}(x;\mathbb{F}_p),$$
$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p) = 1 \otimes \mathrm{ReLU}\big((-2^{p-e_z} \otimes \mathrm{ReLU}(x \ominus 2^{e_z})) \oplus ((a_z - 1 + u)\times 2^p)\big),$$
$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = -1 \otimes \mathrm{ReLU}\big((-2^{p-e_z} \otimes \mathrm{ReLU}(x \ominus 2^{e_z})) \oplus (a_z - 1)\times 2^p\big).$$

If $2^{e_z} < x < 2^{1+e_z}$, $x \ominus 2^{e_z}$ is exact by Lemma 13. Therefore we have

$$\mathrm{ReLU}(x \ominus 2^{e_z}) \begin{cases} = 0 & \text{if } x < 2^{e_z}, \\ = x - 2^{e_z} & \text{if } 2^{e_z} \le x \le 2^{1+e_z}. \\ \ge 2^{e_z} & \text{if } x > 2^{1+e_z}. \end{cases}$$

If $x < 2^{e_z}$, we have

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p) = \mathrm{ReLU}((a_z - 1 + u)\times 2^p),$$
$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = -\mathrm{ReLU}((a_z - 1)\times 2^p),$$
$$f_2(x;\mathbb{F}_p) = \phi_{\theta_{2,1,z}}(x;\mathbb{F}_p) + \phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = 1,$$

with $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p), -\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) \in \{0\}\cup[2^p]$.

If $2^{e_z} \le x \le 2^{1+e_z}$, we have $\mathrm{ReLU}(x - 2^{e_z}) = n_2 \times 2^{-p+e_z}$ where $n_2 = 0, 1, \dots, 2^p$. Since $2^{p-e_z} \otimes \mathrm{ReLU}(x \ominus 2^{e_z}) \in [2^p]$ and $(a_z - 1 + u)\times 2^p, (a_z - 1)\times 2^p \in [2^p]$, by Lemma 16, all operations in $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p)$ and $\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p)$ are exact. Hence

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p) = \mathrm{ReLU}(-2^{p-e_z}x + (a_z + u)\times 2^p),$$
$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = -\mathrm{ReLU}(-2^{p-e_z}x + a_z \times 2^p),$$

with $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p), -\phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) \in \{0\}\cup[2^p]$. Therefore,

$$f_2(x;\mathbb{F}_p) = \begin{cases} 1 & \text{if } 2^{e_z} \le x \le z, \\ 0 & \text{if } z^+ \le x \le 2^{1+e_z}. \end{cases}$$

If $x > 2^{1+e_z}$, we have

$$1 \ge a_z - 1 + u,$$
$$-2^{p-e_z} \otimes \mathrm{ReLU}(x \ominus 2^{e_z}) \le -2^p \le -(a_z - 1 + u)\times 2^p,$$

which leads to $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_p) = \phi_{\theta_{2,2,z}}(x;\mathbb{F}_p) = 0$. Therefore we conclude

$$f_2(x;\mathbb{F}_p) = \mathbb{1}[x \le z].$$

Now, we consider $z < 0$, we define

$$f_1(x;\mathbb{F}_p) = \psi_{1,1,z}(x;\mathbb{F}_p) \oplus \psi_{1,2,z}(x;\mathbb{F}_p),$$
$$f_2(x;\mathbb{F}_p) = \psi_{2,1,z}(x;\mathbb{F}_p) \oplus \psi_{2,2,z}(x;\mathbb{F}_p),$$
$$\psi_{1,1,z}(x;\mathbb{F}_p) = \phi_{2,1,-z}(-x;\mathbb{F}_p),\ \psi_{1,2,z}(x;\mathbb{F}_p) = \phi_{2,2,-z}(-x;\mathbb{F}_p),$$
$$\psi_{2,1,z}(x;\mathbb{F}_p) = \phi_{1,1,-z}(-x;\mathbb{F}_p),\ \psi_{2,2,z}(x;\mathbb{F}_p) = \phi_{1,2,-z}(-x;\mathbb{F}_p).$$

Then we have

$$f_1(x;\mathbb{F}_p) = \phi_{\theta_{2,1,-z}}(-x;\mathbb{F}_p) \oplus \phi_{\theta_{2,2,-z}}(-x;\mathbb{F}_p) = \mathbb{1}[-x \le -z] = \mathbb{1}[x \ge z],$$
$$f_2(x;\mathbb{F}_p) = \phi_{\theta_{1,1,-z}}(-x;\mathbb{F}_p) \oplus \phi_{\theta_{1,2,-z}}(-x;\mathbb{F}_p) = \mathbb{1}[-x \ge -z] = \mathbb{1}[x \le z].$$

Finally, we have $\psi_{\theta_{i,1,z}}(x;\mathbb{F}_p), \psi_{\theta_{i,2,z}}(x;\mathbb{F}_p) \in \{0\}\cup[2^p]$ for all $i \in \{1,2\}$, and

$$f_1(x;\mathbb{F}_p) = \mathbb{1}[x \ge z],\ f_2(x;\mathbb{F}_p) = \mathbb{1}[x \le z],$$

for all $z \in \mathbb{F}_p \setminus \{0\}$.

Lastly, it is easy to observe that $\psi_{\theta_{i,j,z}}$ for all $i,j \in \{1,2\}$ and $z \in \mathbb{F}_p \setminus \{0\}$ share the same network architecture of 3 layers and 5 parameters. This completes the proof. □

5.2. Proof of Lemma 1

Let $\mathbf{x} = (x_1, \dots, x_d)$. For each $i \in [d]$, we define $g_{2j-1}(x_j;\mathbb{F}_p) = \mathbb{1}\big[-x_j \oplus \beta_j \ge 0\big]$ and $g_{2j}(x_j;\mathbb{F}_p) = \mathbb{1}\big[x_j \ominus \alpha_j \ge 0\big]$, and define $f_{\theta_{\alpha,\beta}}$ as follows:

$$f_{\theta_{\alpha,\beta}}(\mathbf{x};\mathbb{F}_p) = \mathbb{1}\left[\left(\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p)\right) \ominus 2d \ge 0\right].$$

From the definition of g_i, one can observe that

$$g_{2j-1}(x_j;\mathbb{F}_p) + g_{2j}(x_j;\mathbb{F}_p) = \begin{cases} 2 & \text{if } x_j \in [\alpha_j, \beta_j], \\ 1 & \text{if } x_j \notin [\alpha_j, \beta_j]. \end{cases}$$

Since $\bigoplus_{i=1}^{m} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p) \le m \le 2^{1+p} - 1$ for $1 \le m \le (2d-1)$, and $|g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p)| = 0, 1$, by Lemma 16, $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p)$ is exact. Therefore, we have $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p) = 2d$ if $\mathbf{x} \in \prod_{i=1}^{d}[\alpha_i, \beta_i]$ and $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_p) < 2d$ otherwise, i.e.,

$$f_{\theta_{\alpha,\beta}}(\mathbf{x},\mathbb{F}_p) = \mathbb{1}\left[\mathbf{x} \in \prod_{j=1}^{d}[\alpha_j, \beta_j]\right].$$

Here, $f_{\theta_{\alpha,\beta}}$ can be constructed via a three-layer Step network of $6d+2$ parameters ($4d$ parameters for the first layer, $2d+1$ parameters for the second layer, and 1 parameter for the last layer.) This completes the proof.

5.3. Proof of Theorem 2

We use Lemma 1 to prove Theorem 2. Consider

$$f_{\theta_{\mathbf{z}_i,\mathbf{z}_i}}(\mathbf{x};\mathbb{F}_p) = \mathbb{1}\left[\mathbf{x} \in \prod_{j=1}^{d}[\mathbf{z}_{i,j}, \mathbf{z}_{i,j}]\right] = \mathbb{1}\big[\mathbf{x} = \mathbf{z}_i\big],$$

and

$$f_{\theta_D}(\mathbf{x};\mathbb{F}_p) = \bigoplus_{i=1}^{n}\Big(y_i \otimes f_{\theta_{\mathbf{z}_i,\mathbf{z}_i}}(\mathbf{x},\mathbb{F}_p)\Big),$$

where $f_{\theta_{\mathbf{z}_i,\mathbf{z}_i}}(\mathbf{x},\mathbb{F}_p)$ is from Section 5.2 in Lemma 1. Then, $f_{\theta_D}(\mathbf{z}_i;\mathbb{F}_p) = y_i$ for all $i \in [n]$ and f can be implemented by a three-layer Step network of $6dn + 2n$ parameters ($4dn$ parameters for the first layer, $2dn + n$ parameters for the second layer, and n parameters for the last layer).

5.4. Proof of Eq. (4)

If $x < 1$, since $x \le (1-u)$, we have $x \times 2^p \le (1-u)\times 2^p < 2^p$. Therefore, $f(x;\mathbb{F}_p) = 0$. If $1 \le x \le 1.1 \times 2^1$, we have $(x\times 2^p) \in [2^{1+p}]$, $(1-u)\times 2^p, 2^p \in [2^p]$. By Lemma 16, $(x\times 2^p)\ominus((1-u)\times 2^p)$ and $(x\times 2^p)\ominus 2^p$ are exact. Therefore,

$$f(x;\mathbb{F}_p) = \big((x\times 2^p) - ((1-u)\times 2^p)\big) - \big((x\times 2^p) - 2^p\big) = 1.$$

If $1.1 \times 2^1 \le x < 1.01 \times 2^2$, let $n_x = (x - 1.1 \times 2^1) \times 2^{p-1} \in \mathbb{N}$. Then we have $0 \le n_x < 2^p$ and

$$(x \times 2^p) \ominus ((1-u) \times 2^p) = \lceil 2^{p+1} + 2n_x + 1 \rfloor,$$
$$(x \times 2^p) \ominus 2^p = \lceil 2^{p+1} + 2n_x \rfloor.$$

For $0 \le n < 2^{p+1}$, since

$$\lceil 2^{p+1} + n \rfloor = \begin{cases} 2^{p+1} + n & \text{if } n \equiv 0, 2 \pmod 4, \\ 2^{p+1} + n - 1 & \text{if } n \equiv 1 \pmod 4, \\ 2^{p+1} + n + 1 & \text{if } n \equiv 3 \pmod 4, \end{cases}$$

we have

$$f(x; \mathbb{F}_p) = \begin{cases} 0 & \text{if } n_x \equiv 0 \pmod 2, \\ 2 & \text{if } n_x \equiv 1 \pmod 2. \end{cases}$$

If $x \ge 1.01 \times 2^2$, let $x' = x - 1$. Then we have $x' \ge 2^2$ and

$$(x \times 2^p) \ominus ((1-u) \times 2^p) = \lceil (x - 1 + u) \times 2^p \rfloor = \lceil x' + u \rfloor \times 2^p,$$
$$(x \times 2^p) \ominus 2^p = \lceil (x-1) \times 2^p \rfloor = x' \times 2^p.$$

Since $\lceil x' + u \rfloor = x'$ by Lemma 17, we have $f(x; \mathbb{F}_p) = 0$.

5.5. Proof of Theorem 3

If $\omega_{f^*}^{-1}(\varepsilon) = \infty$, let $\delta = 1$, and we have $K = 1$. If $\omega_{f^*}^{-1}(\varepsilon) \ne \infty$, let $\delta = (\omega_{f^*}^{-1}(\varepsilon))^{-1}$. For each $i \in [K]$, we define

$$\alpha_i = \begin{cases} i\delta & \text{if } i \in \{0, 1, \dots, K-1\}, \\ 1 & \text{if } i = K, \end{cases}$$

$$\mathcal{I}_i = \begin{cases} [\alpha_{i-1}^{(\ge, \mathbb{F}_p)}, \alpha_i^{(<, \mathbb{F}_p)}] \cap \mathbb{F}_p & \text{if } i \in [K-1], \\ [\alpha_{K-1}^{(\ge, \mathbb{F}_p)}, \alpha_K^{(\le, \mathbb{F}_p)}] \cap \mathbb{F}_p & \text{if } i = K. \end{cases}$$

Without loss of generality, we assume that $\mathcal{I}_i \ne \emptyset$ for all $i \in [K]$; otherwise, we remove empty $\mathcal{I}_j$, decrease K, and re-index $\mathcal{I}_i$ so that $\mathcal{I}_i$ if non-empty for all $i \in [K]$. We note that since $0, 1 \in \mathbb{F}_p$, there is at least one non-empty $\mathcal{I}_i$ and $K \ge 1$. Then, by the above definitions, it holds that $\sup \mathcal{I}_i - \inf \mathcal{I}_i \le \delta$. Since $\alpha_i^{(<, \mathbb{F}_p)} < \alpha_i^{(\ge, \mathbb{F}_p)}$, $\mathcal{I}_{i_1}$ and $\mathcal{I}_{i_2}$ are disjoint if $i_1 \ne i_2$, and $\dot{\bigcup}_{i \in [K]} \mathcal{I}_i = [0,1] \cap \mathbb{F}_p$. For each $\boldsymbol{\iota} = (\iota_1, \dots, \iota_d) \in [K]^d$, we also define

$$\boldsymbol{\gamma}_{\boldsymbol{\iota}} = \operatorname*{arg\,min}_{\mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}} |f^*(\mathbf{x}) - \lceil f^*(\mathbf{x}) \rfloor|,$$

which is well-defined by Lemma 19.

We are now ready to introduce our Step network construction f_θ:

$$f_\theta(\mathbf{x}; \mathbb{F}_p) = \bigoplus_{\boldsymbol{\iota} \in [K]^d} \lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}}) \rfloor \otimes \mathbb{1}\left[\left(\bigoplus_{j=1}^{2d} h_{\boldsymbol{\iota},j}(x_{\lceil j/2 \rceil_{\mathbb{Z}}})\right) \ominus d \ge 0\right],$$

where for each $j \in [d]$ and $\boldsymbol{\iota} = (\iota_1, \dots, \iota_d) \in [K]^d$,

$$h_{\boldsymbol{\iota},2j-1}(x) = \begin{cases} \mathbb{1}\left[x - \alpha_{j-1}^{(\ge, \mathbb{F}_p)} \ge 0\right] & \text{if } \iota_j \in \{2, \dots, K\}, \\ \mathbb{1}\left[x + \alpha_0^{(\ge, \mathbb{F}_p)} \ge 0\right] & \text{if } \iota_j = 1, \end{cases}$$

$$h_{\boldsymbol{\iota},2j}(x) = \begin{cases} -\mathbb{1}\left[x - \alpha_j^{(\ge, \mathbb{F}_p)} \ge 0\right] & \text{if } \iota_j \in \{1, \dots, K-1\}, \\ -\mathbb{1}\left[x - \alpha_K^{(>, \mathbb{F}_p)} \le 0\right] & \text{if } \iota_j = K. \end{cases}$$

Since $h_{\boldsymbol{\iota},j}(x) \in \{-1, 0, 1\}$, $h_{\boldsymbol{\iota},2j-1}(x) + h_{\boldsymbol{\iota},2j}(x) = \{-1, 0, 1\}$, we have

$$\left|\bigoplus_{j=1}^{m} h_{\boldsymbol{\iota},j}(x_{\lceil j/2 \rceil_{\mathbb{Z}}})\right| \le d \le 2^p.$$

for any $1 \le m \le 2d$. Therefore, by Lemma 16, all operations in the computation of $\bigoplus_{j=1}^{2d} h_{\boldsymbol{\iota},j}(x_{\lceil j/2 \rceil_{\mathbb{Z}}})$ are exact. Furthermore, for each $\mathbf{x} \in \mathbb{F}_p^d \cap [0,1]^d$, we have

$$\bigoplus_{j=1}^{2d} h_{\boldsymbol{\iota},j}(x_{\lceil j/2 \rceil_{\mathbb{Z}}}) \begin{cases} = d & \text{if } \mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}, \\ < d & \text{if } \mathbf{x} \in ([0,1]^d \cap \mathbb{F}_p^d) \setminus (\mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}). \end{cases}$$

Since $\dot{\bigcup}_{\boldsymbol{\iota} \in [K]^d} \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d} = \mathbb{F}_p^d \cap [0,1]^d$, we have

$$f_\theta(\mathbf{x}; \mathbb{F}_p) = \lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}}) \rfloor, \ \forall \mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}.$$

Hence, for each $\boldsymbol{\iota} \in [K]^d$ and $\mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}$,

$$\begin{aligned} |f_\theta(\mathbf{x}; \mathbb{F}_p) - f^*(\mathbf{x})| &= |\lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}}) \rfloor - f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})| + |f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}}) - f^*(\mathbf{x})| \\ &\le |\lceil f^*(\mathbf{x}) \rfloor - f^*(\mathbf{x})| + \varepsilon, \end{aligned}$$

where we use $\|\mathbf{x} - \boldsymbol{\gamma}_{\boldsymbol{\iota}}\|_\infty \le \omega_{f^*}(\varepsilon)$ for the above inequality. Since each $h_{\boldsymbol{\iota},j}$ can be implemented by a Step network of 3 parameters, $\mathbb{1}\left[\left(\bigoplus_{j=1}^{2d} h_{\boldsymbol{\iota},j}(x_{\lceil j/2 \rceil_{\mathbb{Z}}})\right) \ominus d \ge 0\right]$ can be implemented using $6d+1$ parameters. This implies that our f_θ can be implemented by a Step network of 3 layers and $(6d+2)K^d$ parameters. This completes the proof.

5.6. Proof of Theorem 5

Define $\delta = \frac{1}{2} \min\{|z_{i,j}| : z_{i,j} \ne 0, i \in [n], j \in [d]\}$. For each $i \in [n]$, we also define $h_{i,1}, \dots, h_{i,4d}$ as follows: for each $j \in [d]$,

$$h_{i,4j-3} = \psi_{\theta_{1,1,t_{i,j,1}}}, \ h_{i,4j-2} = \psi_{\theta_{1,2,t_{i,j,1}}}, \ h_{i,4j-1} = -\psi_{\theta_{1,1,t_{i,j,2}}}, \ h_{i,4j} = -\psi_{\theta_{1,2,t_{i,j,2}}},$$

where $t_{i,j,1} = z_{i,j}$ and $t_{i,j,2} = z_{i,j}^+$ if $z_{i,j} \ne 0$, $t_{i,j,1} = -\delta$ and $t_{i,j,2} = \delta^+$ if $z_{i,j} = 0$, and $\psi_{\theta_{1,1,z}}, \psi_{\theta_{1,2,z}}$ are defined in Lemma 20.

By Lemma 20, we have

$$h_{i,4j-3}(x), -h_{i,4j-2}(x), -h_{i,4j-1}(x), h_{i,4j}(x) \in \{0\} \cup [2^p],$$
$$\bigoplus_{k=1}^{4} h_{i,4j-4+k}(x) = \begin{cases} \mathbb{1}[x = z_{i,j}] & \text{if } z_{i,j} \ne 0, \\ \mathbb{1}[x \in [-\delta, \delta]] & \text{if } z_{i,j} = 0, \end{cases}$$

for all $x \in \mathbb{F}_p$.

Let $0 \le m < d \le 2^p$. For each m, we have $h_{i,4m+1}(x) + h_{i,4m+2}(x) = l_{4m+1} \in \{0, 1\}$ and

$$-2^p < -m \le \bigoplus_{j=1}^{4m} h_{i,j}(x) \le m < 2^p, \ h_{i,4m+1}(x) \in \{0\} \cup [2^p],$$
$$-2^p \le -m + h_{i,4m+1}(x) \le \bigoplus_{j=1}^{4m+1} h_{i,j}(x) \le m + h_{i,4m+1}(x) < 2^{p+1},$$
$$-h_{i,4m+2}(x) \in \{0\} \cup [2^p],$$
$$-2^p < -m + l_{4m+1} \le \bigoplus_{j=1}^{4m+2} h_{i,j}(x) \le m + l_{4m+1} \le 2^p.$$

By Lemma 16, all above operations in $\bigoplus_{j=1}^{4m+2} h_{i,j}(x)$ are exact.

We have $h_{i,4m+3}(x) + h_{i,4m+4}(x) = l_{4m+3} \in \{0, -1\}, l_{4m+1} + l_{4m+3} \in \{0, -1\}$, and

$$-2^p \le -m + l_{4m-1} \le \bigoplus_{j=1}^{4m+2} h_{i,j}(x) \le 2^p,$$
$$-h_{i,4m+3}(x) \in \{0\} \cup [2^p],$$
$$-2^{1+p} \le -m + l_{4m-1} + h_{i,4m+3}(x) \le \bigoplus_{j=1}^{4m+3} h_{i,j}(x) \le 2^p + h_{i,4m+3}(x),$$
$$h_{i,4m+4}(x) \in \{0\} \cup [2^p],$$
$$-2^p \le -m + l_{4m-1} + l_{4m-3} \le \bigoplus_{j=1}^{4m+4} h_{i,j}(x) < 2^p + l_{4m-3} \le 2^p.$$

By Lemma 16, all above operations in $\bigoplus_{j=1}^{4m+2} h_{i,j}(x)$ are exact.

Therefore we conclude that all operations in $\bigoplus_{j=1}^{4d} h_{i,j}(x)$ are exact with $|\bigoplus_{j=1}^{4d} h_{i,j}(x)| \in \{0\} \cup [2^p]$.

We design the target network f_θ as follows:

$$f_\theta(\mathbf{x};\mathbb{F}_p)=\bigoplus_{i=1}^{n} y_i\otimes \mathrm{ReLU}\left(\left(\bigoplus_{j=1}^{4d} h_{i,j}(x_{\lceil j/4\rceil_{\mathbb{Z}}})\right)\ominus(d-1)\right).$$

Since for each $k\in[n]$

$$\bigoplus_{j=1}^{4d} h_{i,j}(z_{k,\lceil j/4\rceil_{\mathbb{Z}}})\begin{cases}=d & \text{if } \mathbf{z}_k=\mathbf{z}_i,\\ <d & \text{if } \mathbf{z}_k\neq\mathbf{z}_i,\end{cases}$$

f_θ memorizes the target dataset. Since there are 5 parameters for each $h_{i,j}$, f_θ has $20dn+2n$ parameters. This completes the proof.

5.7. Proof of Theorem 6

The proof of Theorem 6 is almost identical to that of Theorem 3; we define $\mathcal{I}_i$, α_i, and $\boldsymbol{\gamma}_\iota$ as in Section 5.5. For each $\iota\in[K]^d$, $j\in[d]$, we also define $h_{\iota,1},\dots,h_{\iota,4d}$ as follows:

$$h_{\iota,4j-3}=\psi_{\theta_{1,1,t_{\iota,j,1}}},\ h_{\iota,4j-2}=\psi_{\theta_{1,2,t_{\iota,j,1}}},\ h_{\iota,4j-1}=-\psi_{\theta_{1,1,t_{\iota,j,2}}},\ h_{\iota,4j}=-\psi_{\theta_{1,2,t_{\iota,j,2}}},$$

where

$$t_{\iota,j,1}=\begin{cases}\alpha_{j-1}^{(\geq,\mathbb{F}_p)} & \text{if } \iota_j\in\{2,\dots,K\},\\ -\alpha_0^{(\geq,\mathbb{F}_p)} & \text{if } \iota_j=1,\end{cases}$$

$$t_{\iota,j,2}=\begin{cases}\alpha_{j}^{(\geq,\mathbb{F}_p)} & \text{if } \iota_j\in\{1,\dots,K-1\},\\ \alpha_K^{(>,\mathbb{F}_p)} & \text{if } \iota_j=K,\end{cases}$$

and $\psi_{\theta_{1,1,z}},\psi_{\theta_{1,2,z}}$ are defined in Lemma 20. Namely, we have

$$h_{\iota,4j-3}(x)\oplus h_{\iota,4j-2}(x)=\begin{cases}\mathbb{1}\left[x\geq\alpha_{j-1}^{(\geq,\mathbb{F}_p)}\right] & \text{if } \iota_j\in\{2,\dots,K\},\\ \mathbb{1}\left[x\geq-\alpha_0^{(\geq,\mathbb{F}_p)}\right] & \text{if } \iota_j=1,\end{cases}$$

$$h_{\iota,4j-1}(x)\oplus h_{\iota,4j}(x)=\begin{cases}-\mathbb{1}\left[x\geq\alpha_{j}^{(\geq,\mathbb{F}_p)}\right] & \text{if } \iota_j\in\{1,\dots,K-1\},\\ -\mathbb{1}\left[x\geq-\alpha_K^{(>,\mathbb{F}_p)}\right] & \text{if } \iota_j=K,\end{cases}$$

for all $x\in\mathbb{F}_p$ by Lemmas 16 and 20. We design the target network f_θ as follows:

$$f_\theta(\mathbf{x};\mathbb{F}_p)=\bigoplus_{\iota\in[K]^d}\lceil f^*(\boldsymbol{\gamma}_\iota)\rfloor\otimes\mathrm{ReLU}\left(\left(\bigoplus_{j=1}^{4d}h_{\iota,j}(x_{\lceil j/4\rceil_{\mathbb{Z}}})\right)\ominus(d-1)\geq 0\right).$$

By similar argument presented in the proof of Theorem 5, all operations in $\bigoplus_{j=1}^{4d}h_{\iota,j}(x_{\lceil j/4\rceil_{\mathbb{Z}}})$ are exact by Lemma 16, i.e., for each $k\in[n]$

$$\bigoplus_{j=1}^{4d}h_{\iota,j}(x_{\lceil j/4\rceil_{\mathbb{Z}}})\begin{cases}=d & \text{if } \mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d},\\ \leq d-1 & \text{if } ([0,1]^d\cap\mathbb{F}_p^d)\setminus(\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}),\end{cases}$$

This implies that for each $\iota\in[K]^d$ and $\mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}$,

$$\begin{aligned}|f_\theta(\mathbf{x};\mathbb{F}_p)-f^*(\mathbf{x})|&=|\lceil f^*(\boldsymbol{\gamma}_\iota)\rfloor-f^*(\boldsymbol{\gamma}_\iota)|+|f^*(\boldsymbol{\gamma}_\iota)-f^*(\mathbf{x})|\\&\leq|\lceil f^*(\mathbf{x})\rfloor-f^*(\mathbf{x})|+\varepsilon,\end{aligned}$$

where we use $\|\mathbf{x}-\boldsymbol{\gamma}_\iota\|_\infty\leq\omega_{f^*}(\varepsilon)$ for the above inequality. Since each $h_{\iota,j}$ can be implemented by a ReLU network of 3 layers and 5 parameters by Lemma 20, $\mathbb{1}\left[\left(\bigoplus_{j=1}^{4d}h_{\iota,j}(x_{\lceil j/4\rceil_{\mathbb{Z}}})\right)\ominus(d-1)\geq 0\right]$ can be implemented using $20d+1$ parameters. This implies that our f can be implemented by a ReLU network of 4 layers and $(20d+2)K^d$ parameters. This completes the proof.

6. Proofs of results under $\mathbb{F}_{p,q}$

6.1. Addition notations for $\mathbb{F}_{p,q}$

In this section, we introduce notations frequently used for proving our results under $\mathbb{F}_{p,q}$. For $x\in\mathbb{F}_{p,q}$, if x is normal, we represent x as

$$x=s_x\times a_x\times 2^{e_x},\ s_x\in\{-1,1\},a_x=1.x_1\cdots x_p.$$

If x is subnormal, it is standard to represent x as

$$x=s_x\times a_x\times 2^{e_{min}},\ s_x\in\{-1,1\},a_x=0.x_1\cdots x_p,$$
$$x_1=\cdots=x_{c_x-1}=0,x_{c_x}=1.$$

for some $1\leq c_x<p$. However, instead of using the above representation, we opt for a different one for the sake of convenience.

$$x=\mathfrak{s}_x\times\mathfrak{a}_x\times 2^{\mathfrak{e}_x},\ \mathfrak{s}_x\in\{-1,1\},\mathfrak{a}_x=1.x_1\cdots x_{p-c_x},\mathfrak{e}_x=e_{min}-c_x.$$

It is convenient because, regardless of whether x is normal or subnormal, we have the following representation for x:

$$x=\mathfrak{s}_x\times\mathfrak{a}_x\times 2^{\mathfrak{e}_x},\ \mathfrak{s}_x\in\{-1,1\},\mathfrak{a}_x=1.\mathfrak{x}_1\cdots\mathfrak{x}_p,\ 2^{\mathfrak{e}_x}\leq x<2^{1+\mathfrak{e}_x}. \tag{7}$$

Note that if x is normal, we have $\mathfrak{s}_x=s_x,\mathfrak{a}_x=a_x,\mathfrak{e}_x=e_x$. We define $\mu(x)$ as

$$\mu(x)\triangleq\inf\{m\in\mathbb{Z}:x\times 2^{-m}\in\mathbb{N}\}.$$

Note that if $x\neq 0$, we can represent x as

$$x=(n_x\times 2^{-\mathfrak{e}_x+\mu(x)})\times 2^{\mathfrak{e}_x}, \tag{8}$$

for $n_x=x\times 2^{-\mu(x)}\in\mathbb{N}$ with $2^{\mathfrak{e}_x-\mu(x)}\leq n_x<2^{1+\mathfrak{e}_x-\mu(x)}$. We define e_0 as

$$e_0\triangleq e^{q-2}-1.$$

Definition 21. For a number $x\in\mathbb{R}$, we say x is "representable" by $\mathbb{F}_{p,q}$ if $x\in\mathbb{F}_{p,q}$.

Remark 22. For nonzero $x\in\mathbb{R}$, suppose x has a following representation

$$x=\mathfrak{s}_x\times\mathfrak{a}_x\times 2^{\mathfrak{e}_x},\ \mathfrak{s}_x\in\{-1,1\},1\leq\mathfrak{a}_x<2,\ \mathfrak{e}_x\in\mathbb{Z},2^{\mathfrak{e}_x}\leq x<2^{1+\mathfrak{e}_x}.$$

Define $c_x\triangleq\max\{0,e_{min}-\mathfrak{e}_x\}$. Then x is representable by $\mathbb{F}_{p,q}$ if

$$-p+e_{min}\leq\mathfrak{e}_x\leq e_{max},\ \mathfrak{a}_x\times 2^{p-c_x}\in\mathbb{N},$$

which leads to

$$-p+e_{min}\leq\mathfrak{e}_x\leq e_{max},\ 0\leq\mathfrak{e}_x-\mu(x)\leq p-c_x. \tag{9}$$

We say representability test on x is to check whether x satisfies Eq. (9).

6.2. Technical lemmas

We introduce technical lemmas used for proving our results under $\mathbb{F}_{p,q}$.

Lemma 23. *Let $x=n\times 2^m$ for some $n\in\mathbb{N}$, $m\in\mathbb{Z}$. Let $c_x=\max\{0,e_{min}-m\}$. If*

$$0<n<2^{1+p-c_x},\ -p+e_{min}\leq m\leq -p+e_{max},$$

then x is representable by $\mathbb{F}_{p,q}$.

Proof. Since $0\leq n<2^{1+p-c_x}$, there exists $c_0\in\{0\}\cup[p-c_x]$ such that $2^{p-c_x-c_0}\leq n<2^{1+p-c_x-c_0}$. Note that x has the following representation in $\mathbb{F}_{p,q}$,

$$x=(n\times 2^{-p+c_x+c_0})\times 2^{p-c_x-c_0+m},$$
$$1\leq n\times 2^{-p+c_x+c_0}<2,\ -p+e_{min}\leq p-c_x-c_0+m\leq e_{max}.$$

Then we express $n \times 2^{-p+c_x+c_0}$ as

$$n \times 2^{-p+c_x+c_0} = 1.\underbrace{w_1 \cdots w_{p-c_x-c_0}}_{p-c_x-c_0 \text{ times}},$$

for some $w_1, \ldots, w_{p-c_x-c_0} \in \{0,1\}$. Therefore x is representable by $\mathbb{F}_{p,q}$. □

Lemma 24. *Let $x, y \in \mathbb{F}_{p,q}$ and $\mathfrak{s}_x = \mathfrak{s}_y$ and $\mathfrak{e}_x \le \mathfrak{e}_y$. If $\mu(x) \ge \mathfrak{e}_y - p$, then $x \ominus y$ and $y \ominus x$ are exact. In addition, if $|x+y| \le 2^{1+\mathfrak{e}_y}$, then $x \oplus y$ is exact.*

Proof. Let $c_y = \max\{0, e_{min} - \mathfrak{e}_y\}$. Note that if $\mathfrak{e}_y \ge e_{min}$ (i.e. y is normal), we have $c_y = 0$. If $\mathfrak{e}_y < e_{min}$ (i.e. y is subnormal), we have $c_y = e_{min} - \mathfrak{e}_y > 0$ and $\mu(x) \ge -p + e_{min} = -p + \mathfrak{e}_y + c_y$. Let $k \triangleq \mu(x) - \mathfrak{e}_y + p - c_y \ge 0$. As described in Eq. (8), we can represent x and y as

$$x = (n_x \times 2^{-\mathfrak{e}_x+\mu(x)}) \times 2^{\mathfrak{e}_x} = n_x \times 2^{\mu(x)},\ 2^{\mathfrak{e}_x-\mu(x)} \le n_x < 2^{1+\mathfrak{e}_x-\mu(x)},$$
$$y = (n_y \times 2^{-p+c_y}) \times 2^{\mathfrak{e}_y} = n_y \times 2^{-p+c_y+\mathfrak{e}_y},\ 2^{p-c_y} \le n_y < 2^{1+p-c_y},$$

for some $n_x, n_y \in \mathbb{N}$. Since $k \ge 0$, we have

$$x = (2^k n_x) \times 2^{-p+\mathfrak{e}_y+c_y},\ 2^{p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)} \le 2^k n_x < 2^{1+p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)}.$$

Therefore for $n' = n_y - 2^k n_x \in \mathbb{N}$, we have

$$y - x = n' \times 2^{-p+\mathfrak{e}_y+c_y},\ 2^{p-c_y} - 2^{1+p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)} < n' < 2^{1+p-c_y} - 2^{p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)},$$

which leads to

$$-2^{p-c_y} < n' < 2^{p+1-c_y} - 1.$$

Since $|n'| = 0$ or $|n'| < 2^{1+p-c_y} - 1$ and $-p + e_{min} \le -p + \mathfrak{e}_y + c_y \le e_{max}$, by Lemma 23, $y - x = n' \times 2^{-p+\mathfrak{e}_y+c_y}$ is representable by $\mathbb{F}_{p,q}$. This ensures that $y \ominus x$ is exact.

Now suppose $|x+y| \le 2^{1+\mathfrak{e}_y}$. Then for $n'' = n_x + n_y \in \mathbb{N}$, we have

$$x + y = n'' \times 2^{-p+\mathfrak{e}_y+c_y},\ 2^{p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)} + 2^{p-c_y} \le n'' < 2^{1+p-(\mathfrak{e}_y-\mathfrak{e}_x+c_y)} + 2^{1+p-c_y}.$$

Since $|x+y| \le 2^{1+\mathfrak{e}_y}$, we have $n'' \le 2^{1+p-c_y}$. If $|n''| = 0$ or $|n''| < 2^{1+p-c_y}$, since $-p + e_{min} \le -p + \mathfrak{e}_y + c_y \le e_{max}$, by Lemma 23, $x + y = n'' \times 2^{-p+\mathfrak{e}_y+c_y}$ is representable by $\mathbb{F}_{p,q}$. If $|n''| = 2^{1+p-c_y}$, $x + y = 2^{1+\mathfrak{e}_y}$ is obviously representable $\mathbb{F}_{p,q}$. Therefore, $x \oplus y$ is exact. □

Lemma 25. *In $\mathbb{F}_{p,q}$, if $x, y \in [2^{1+p}]$, $x \ominus y$ and $y \ominus x$ are exact. In addition if $x, y \in [2^p]$, then $x \oplus y$ is exact.*

Proof. Without loss of generality, suppose $x \le y$ for $x, y \in [2^{1+p}]$. Since $y - x < 2^{1+p}$, $y - x$ is representable by $\mathbb{F}_{p,q}$ by Lemma 23, ensuring that $x \ominus y$ and $y \ominus x$ are exact.

In addition, suppose $x, y \in [2^p]$ and $x \le y$. Since $\mu(x) \ge 0$ and $\mathfrak{e}_y \le p$, we have $\mu(x) \ge -p + \mathfrak{e}_y$. Since $|x+y| < 2^{1+p}$, $x \oplus y$ is exact by Lemma 24. □

Lemma 26. *Let $x, y \in \mathbb{F}_{p,q}$ and $\mathfrak{e}_x \le \mathfrak{e}_y$. If $\mathfrak{e}_x \le -2 - p + \mathfrak{e}_y$, then $y \oplus x = y \ominus x = y$.*

Proof. Since $|x| \le 2^{-2-p+\mathfrak{e}_y}$, we have $\lceil x + y \rfloor = \lceil -x + y \rfloor = y$. □

Lemma 27. *There exists* ReLU *networks $\psi_\theta(\cdot; \mathbb{F}_{p,q}) : \mathbb{F}_{p,q} \to \mathbb{F}_{p,q}$ of 3 layers and 5 parameters that satisfies the following: for any $z \in \mathbb{F}_{p,q}$ satisfying $0 < |z| \le (2-u) \times 2^{-2-p+e_{max}}$, there exist $\theta_{1,1,z}, \theta_{1,2,z}, \theta_{2,1,z}, \theta_{2,2,z} \in \mathbb{F}_{p,q}^5$ such that*

$$\psi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) \oplus \psi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}) = \mathbb{1}[x \ge z],$$
$$\psi_{\theta_{2,1,z}}(x; \mathbb{F}_{p,q}) \oplus \psi_{\theta_{2,2,z}}(x; \mathbb{F}_{p,q}) = \mathbb{1}[x \le z],$$

for $|x| \le (2-u) \times 2^{-2+e_0}$. In addition, we have $\psi_{\theta_{i,1,z}}(x; \mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$, $-\psi_{\theta_{i,2,z}}(x; \mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$ for all $i \in \{1,2\}$.

Proof. In this proof, to ensure that indicator functions are realizable by ReLU networks in $\mathbb{F}_{p,q}$, we should check (1) whether every parameter in the network is representable by $\mathbb{F}_{p,q}$ and (2) whether every number occurring during the intermediate calculations is also representable by $\mathbb{F}_{p,q}$. Hence, we conduct the representability tests (Remark 22) for each condition.

We first consider $z > 0$. We define

$$u_0 = \begin{cases} 2^{-p}\ (= u) & \text{if } \mathfrak{e}_z \ge e_{min}, \\ 2^{-p+c_z} & \text{if } \mathfrak{e}_z < e_{min}, \end{cases}$$
$$k = \begin{cases} -p + e_0 & \text{if } 0 < \mathfrak{e}_z \le -2 - p + e_{max}, \\ 3 - p - e_0 - c_z & \text{if } \mathfrak{e}_z \le 0, \end{cases}$$
$$\tilde{u}_0 = \begin{cases} 0 & \text{if } \mathfrak{a}_z \ne 1, \text{ or } \mathfrak{e}_z = -p + e_{min}, \\ u_0 \times 2^{-1} & \text{if } \mathfrak{a}_z = 1, \end{cases}$$
$$\tilde{c} = \begin{cases} 0 & \text{if } \mathfrak{a}_z \ne 1, \text{ or } \mathfrak{e}_z = -p + e_{min}, \\ 1 & \text{if } \mathfrak{a}_z = 1, \end{cases}$$

where $c_z = \max\{e_{min} - \mathfrak{e}_z, 0\} \ge 0$, and recall that

$$e_{min} = -2^{q-1} + 2, e_{max} = 2^{q-1} - 1, e_0 = 2^{q-2} - 1, 4 \le p \le 2^{q-2} + 2.$$

To construct $\mathbb{1}[x \ge z]$, we define a three-layer ReLU network $f_1(x; \mathbb{F}_{p,q})$ as follows:

$$f_1(x; \mathbb{F}_{p,q}) = \psi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) \oplus \psi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}), \tag{10}$$

where

$$\psi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) = \phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}),\ \psi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}) = \phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}),$$
$$\phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \otimes \mathrm{ReLU}\big((-2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \oplus (2 - \mathfrak{a}_z - \tilde{u}_0) \times 2^{p+k}\big),$$
$$\phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \otimes \mathrm{ReLU}\big((-2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \oplus (2 - \mathfrak{a}_z - u_0) \times 2^{p+k}\big).$$

Here, $g_1(x)$ is defined as

$$g_1(x) := \mathrm{ReLU}(-x \oplus ((2 - u_0) \times 2^{\mathfrak{e}_z})).$$

Let

$$\zeta_{1,1}(x) := (-2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \oplus (2 - \mathfrak{a}_z - \tilde{u}_0) \times 2^{p+k},$$
$$\zeta_{1,2}(x) := (-2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \oplus (2 - \mathfrak{a}_z - u_0) \times 2^{p+k},$$
$$\phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \otimes \mathrm{ReLU}(\zeta_{1,1}(x)),$$
$$\phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \otimes \mathrm{ReLU}(\zeta_{1,2}(x)).$$

Note that (1) the following are parameters in $\phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q})$ and $\phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q})$:

$$2^{\tilde{c}-c_z-k}, 2^{p-\mathfrak{e}_z+k}, (2 - u_0) \times 2^{\mathfrak{e}_z}, (2 - \mathfrak{a}_z - \tilde{u}_0) \times 2^{p+k}, (2 - \mathfrak{a}_z - u_0) \times 2^{p+k}. \tag{11}$$

The following are (2) the numbers occurring during the intermediate calculations in $\phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q})$ and $\phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q})$:

$$g_1(x), -2^{p-\mathfrak{e}_z+k} \otimes g_1(x), \zeta_{1,1}(x), \zeta_{1,2}(x), \phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}), \phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}). \tag{12}$$

Now, we consider the following cases.

Case 1-1: $\mathfrak{e}_z = -p + e_{min}$. In this case, we have

$$z = 2^{-p+e_{min}},\ c_z = p,\ k = 3 - 2p - e_0, \tag{13}$$
$$2^{\tilde{c}-c_z-k} = 2^{-3+p+e_0}, 2^{p-\mathfrak{e}_z+k} = 2^{3-e_0-e_{min}}, (2 - u_0) \times 2^{\mathfrak{e}_z} = 2^{-p+e_{min}}, \tag{14}$$
$$(2 - \mathfrak{a}_z - \tilde{u}_0) \times 2^{p+k} = 2^{3-p-e_0},\ (2 - \mathfrak{a}_z - u_0) \times 2^{p+k} = 0, \tag{15}$$
$$\phi_{\theta_{1,1,z}}(x; \mathbb{F}_{p,q}) = 2^{-3+p+e_0} \otimes \mathrm{ReLU}\big((-2^{3-e_0-e_{min}} \otimes g_1(x)) \oplus 2^{3-p-e_0}\big),$$
$$\phi_{\theta_{1,2,z}}(x; \mathbb{F}_{p,q}) = -2^{-3+p+e_0} \otimes \mathrm{ReLU}\big(-2^{3-e_0-e_{min}} \otimes g_1(x)\big),$$

where $g_1(x) = \mathrm{ReLU}(-x \oplus 2^{-p+e_{min}})$.

If $-(2-u) \times 2^{-2+e_0} \le x \le 0$, we have

$$2^{-p+e_{min}} \le \mathrm{ReLU}(-x \oplus 2^{-p+e_{min}})$$
$$\le \mathrm{ReLU}((2-u) \times 2^{-2+e_0} \oplus 2^{-p+e_{min}}) = (2-u) \times 2^{-2+e_0},$$

where the last equation is due to Lemma 26. Therefore,

$$2^{-p+e_{min}} \le g_1(x) \le (2-u)\times 2^{-2+e_0}, \tag{16}$$

which leads to

$$-2^{3-e_0-e_{min}} \otimes g_1(x) \le -2^{3-e_0-e_{min}} \otimes 2^{-p+e_{min}} = -2^{3-p-e_0}, \tag{17}$$

and

$$-2^{3-e_0-e_{min}} \otimes g_1(x) \ge -(2-u)\times 2^{1-e_{min}} = -(2-u)\times 2^{e_{max}}. \tag{18}$$

Hence, the following inequalities hold:

$$-(2-u)\times 2^{e_{max}} \le \zeta_{1,1}(x) \le 0, \tag{19}$$

$$-(2-u)\times 2^{e_{max}} \le \zeta_{1,2}(x) \le 0. \tag{20}$$

Therefore,

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 0, \tag{21}$$

which leads to $f_1(x;\mathbb{F}_{p,q}) = 0$.

If $2^{-p+e_{min}} \le x \le \Omega$, we have

$$\mathrm{ReLU}(2^{-p+e_{min}} \ominus x) = 0,\ \zeta_{1,1}(x) = 2^{3-p-e_0},\ \zeta_{1,2}(x) = 0, \tag{22}$$

leading to $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 1, \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 0, f_1(x;\mathbb{F}_{p,q}) = 1$.

Representability test for Case 1-1. (1) We check that the parameters in Eqs. (14) and (15) are representable by $\mathbb{F}_{p,q}$. Since

$$-3+p+e_0 \le -3+(2^{q-2}+2)+(2^{q-2}-1) = 2^{q-1}-2 < e_{max},$$

2^{-3+p+e_0} is representable by $\mathbb{F}_{p,q}$. Apparently, $2^{3-e_0-e_{min}}$, $2^{-p+e_{min}}$, 2^{3-p-e_0}, 2^{3-p-e_0}, and 0 are representable by $\mathbb{F}_{p,q}$.

(2) We check that the intermediate numbers in Eq. (12) are representable by $\mathbb{F}_{p,q}$, which is apparent by Eqs. (16)–(22)

Therefore, we conclude

$$f_1(x;\mathbb{F}_{p,q}) \begin{cases} = 0 & \text{if } -(2-u)\times 2^{-2+e_0} \le x \le 0, \\ = 1 & \text{if } 2^{-p+e_{min}} \le x \le \Omega. \end{cases} \tag{23}$$

Case 1-2: $1-p+e_{min} \le \mathfrak{e}_z \le 0$.

In this case, we have $k = 3-p-e_0-c_z$. Let $\Omega_1 = \min\{\Omega, (2-u)\times 2^{-3+e_{max}+e_0+\mathfrak{e}_z+c_z}\}$. Since

$$-3+e_{max}+e_0+\mathfrak{e}_z+c_z \ge -3+e_0+(e_{max}+e_{min}) = -2+e_0,$$

we have $\Omega_1 \ge (2-u)\times 2^{-2+e_0}$.

Suppose $-\Omega_1 \le x \le (2-2u_0)\times 2^{-1+\mathfrak{e}_z}$. If $-\Omega_1 \le x \le -2^{2+p+\mathfrak{e}_z}$, we have $(2-u_0)\times 2^{\mathfrak{e}_z} \ominus x = -x$ by Lemma 26. If $-2^{2+p+\mathfrak{e}_z} < x \le (2-2u_0)\times 2^{-1+\mathfrak{e}_z}$, we have $(2-u_0)\times 2^{-1+\mathfrak{e}_z} \le (2-u_0)\times 2^{\mathfrak{e}_z} \ominus x < 2^{2+p+\mathfrak{e}_z}$. Therefore, we have

$$2^{\mathfrak{e}_z} \le g_1(x) \le \Omega_1, \tag{24}$$

Since

$$2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 \le 2^{p-\mathfrak{e}_z+k} \otimes (2-u)\times 2^{-3+e_{max}+e_0+\mathfrak{e}_z+c_z} = (2-u)\times 2^{e_{max}} = \Omega,$$

we have $2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 \le \Omega$. If $\Omega_1 = (2-u)\times 2^{-3+e_{max}+e_0+\mathfrak{e}_z+c_z}$, we have $2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 = \Omega$. Hence, $2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 \ge \Omega \ge (2-u)\times 2^{4+e_0}$. If $\Omega_1 = \Omega$, since

$$\begin{aligned} &p-\mathfrak{e}_z+k+e_{max} = p-\mathfrak{e}_z+(3-p-e_0-c_z)+e_{max} \\ &= 3-e_0-(\mathfrak{e}_z+c_z)+e_{max} \ge 3-e_0+e_{max} = 4+e_0, \end{aligned}$$

we have $2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 \ge (2-u)\times 2^{4+e_0}$. Therefore, we conclude that

$$(2-u)\times 2^{4+e_0} \le 2^{p-\mathfrak{e}_z+k} \otimes \Omega_1 \le \Omega. \tag{25}$$

By Lemma 24, $-2^{p+k} \oplus (2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}$ and $-2^{p+k} \oplus (2-\mathfrak{a}_z-u_0)\times 2^{p+k}$ are exact. Together with Eq. (24) and Eq. (25), we have

$$2^{\mathfrak{e}_z} \le g_1(x) \le \Omega_1,$$

$$-(2-u)\times 2^{4+e_0} \le -2^{p-\mathfrak{e}_z+k} \otimes g_1(x) \le -2^{p+k}, \tag{26}$$

$$-(2-u)\times 2^{4+e_0} \le \zeta_{1,1}(x) \le -(\mathfrak{a}_z-1+\tilde{u}_0)\times 2^{p+k} < 0, \tag{27}$$

$$-(2-u)\times 2^{4+e_0} \le \zeta_{1,2}(x) \le -(\mathfrak{a}_z-1+u_0)\times 2^{+p+k} < 0, \tag{28}$$

which leads to $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 0$.

If $2^{\mathfrak{e}_z} \le x \le (2-2u_0)\times 2^{\mathfrak{e}_z}$, $-x \oplus ((2-u_0)\times 2^{\mathfrak{e}_z})$ is exact by Lemma 13. Therefore, we have $2^{p-\mathfrak{e}_z+k} \otimes g_1(x) = (n_1 \times 2^{-p+c_z})\times 2^{p+k}$ where $n_1 = 1, 2, \ldots, 2^{p-c_z}-1$ which leads to

$$2^{p-\mathfrak{e}_z+k} \otimes g_1(x) \le (2-2u_0)\times 2^{-1+p+k},\ \mu(2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \ge k. \tag{29}$$

We consider the following cases.

Case 1-2-1: $\mathfrak{a}_z \ne 1$.

In this case, we have

$$2^k \le (2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}, (2-\mathfrak{a}_z-u_0)\times 2^{p+k} < 2^{p+k}, \tag{30}$$
$$\mu((2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}), \mu((2-\mathfrak{a}_z-u_0)\times 2^{p+k}) \ge k.$$

Case 1-2-2: $\mathfrak{a}_z = 1$.

In this case, we have

$$(2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k} = (2-u_0)\times 2^{-1+p+k}, \tag{31}$$

$$(2-\mathfrak{a}_z-u_0)\times 2^{p+k} = (2-2u_0)\times 2^{-1+p+k}. \tag{32}$$

In both cases, since $\mu(2^{p-\mathfrak{e}_z+k} \otimes g_1(x)) \ge k$, by Lemma 24, all operations in $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q})$ are exact. Hence

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \times \mathrm{ReLU}\big(2^{p-\mathfrak{e}_z+k} \times x - ((\mathfrak{a}_z-u_0+\tilde{u}_0)\times 2^{p+k})\big), \tag{33}$$

$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \times \mathrm{ReLU}\big(2^{p-\mathfrak{e}_z+k} \times x - (\mathfrak{a}_z \times 2^{p+k})\big), \tag{34}$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$. Therefore, we have

$$f_1(x;\mathbb{F}_{p,q}) = \begin{cases} 0 & \text{if } (2-u_0)\times 2^{-1+\mathfrak{e}_z} \le x \le z^-, \\ 1 & \text{if } z \le x \le (2-2u_0)\times 2^{\mathfrak{e}_z}. \end{cases}$$

If $(2-u_0)\times 2^{\mathfrak{e}_z} \le x \le \Omega$, we have $g_1(x) = 0$. Hence

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \otimes ((2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}), \tag{35}$$

$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \otimes ((2-\mathfrak{a}_z-u_0)\times 2^{p+k}), \tag{36}$$

$$f_1(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 2^{p+\tilde{c}-c_z} \times (u_0-\tilde{u}_0) = 1. \tag{37}$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$.

Representability test for Case 1-2.

(1) We check that the parameters in Eq. (11) are representable by $\mathbb{F}_{p,q}$. Since

$$\begin{aligned} &e_{min} < \tilde{c}-c_z-k \le 1-(3-p-e_0) = -2+p+e_0 \\ &= -2+(2^{q-2}+2)+(2^{q-2}-1) \le 2^{q-1}-1 = e_{max}, \\ &e_{min} < 3-e_0 \le p-\mathfrak{e}_z+k = p-\mathfrak{e}_z+(3-p-e_0-c_z) \\ &= (3-e_0)+(-\mathfrak{e}_z-c_z) \le 3-e_0-e_{min} \\ &= 3-(2^{q-2}-1)+(2^{q-1}-2) = 2^{q-2}+2 < e_{max}, \end{aligned}$$

$2^{\tilde{c}-c_z-k}$ and $2^{p-\mathfrak{e}_z+k}$ are representable by $\mathbb{F}_{p,q}$. Since

$$\begin{aligned} &e_{min} = -2^{q-1}+2 \le 4-p-2^{q-2} \le 3-p-e_0 \le p+k = 3-e_0-c_z < e_{max}, \\ &e_{min} < 3-p-e_0 = (-p+c_z)+(p+k) \le \mu(t) \le \mathfrak{e}_t \le p+k < e_{max}, \\ &\mathfrak{e}_t-\mu(t) \le (p+k)-(-(p-c_z)+(p+k)) = p-c_z, \end{aligned}$$

where $\mathfrak{a}_z \ne 2-u_0$ for $t = (2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}$, $(2-\mathfrak{a}_z-u_0)\times 2^{p+k}$, they are representable by $\mathbb{F}_{p,q}$. If $\mathfrak{a}_z = 2-u_0$, it is obvious. Apparently, $(2-u_0)\times 2^{\mathfrak{e}_z}$ is representable by $\mathbb{F}_{p,q}$.

(2) We check that the intermediate numbers in Eq. (12) are representable by $\mathbb{F}_{p,q}$, which is apparent by Eqs. (24) and (26)–(37). Therefore, we conclude

$$f_1(x;z) \begin{cases} = 0 & \text{if } -\Omega_1 \le x \le z^-, \\ = 1 & \text{if } z \le x \le \Omega. \end{cases}$$

Case 1-3: $0 < \mathfrak{e}_z \le -2-p+e_{max}$. In this case, $c_z = 0$, $u_0 = u$, and $k = -p+e_0$.

Let $\Omega_2 = \min\{\Omega, (2-u)\times 2^{e_{max}-e_0+\mathfrak{e}_z}\}$. Since

$$e_{max} - e_0 + \mathfrak{e}_z \geq e_{max} - e_0 + 1 = 2 + e_0,$$

we have $\Omega_2 \geq (2-u)\times 2^{2+e_0}$.

Suppose $-\Omega_2 \leq x \leq (2-u)\times 2^{-1+\mathfrak{e}_z}$. If $-\Omega_2 \leq x < -\min\{\Omega_2, 2^{2+p+\mathfrak{e}_z}\}$, we have $(2-u)\times 2^{\mathfrak{e}_z} \ominus x = -x$ by Lemma 26. If $-\min\{\Omega_2, 2^{2+p+\mathfrak{e}_z}\} < x \leq (2-u)\times 2^{-1+\mathfrak{e}_z}$, we have $(2-u)\times 2^{-1+\mathfrak{e}_z} \leq (2-u)\times 2^{\mathfrak{e}_z} \ominus x < 2^{2+p+\mathfrak{e}_z}$. Therefore, we have

$$(2-u)\times 2^{-1+\mathfrak{e}_z} \leq g_1(x) \leq \Omega_2, \tag{38}$$

If $\Omega_2 = (2-u)\times 2^{e_{max}-e_0+\mathfrak{e}_z}$, we have $2^{p-\mathfrak{e}_z+k} \otimes \Omega_2 = \Omega$. If $\Omega_2 = \Omega$, we have $(2-u)\times 2^{e_0} \leq (2-u)\times 2^{-\mathfrak{e}_z+e_0+e_{max}} = 2^{p-\mathfrak{e}_z+k} \otimes \Omega_2$. Therefore, we have

$$(2-u)\times 2^{e_0} \leq 2^{p-\mathfrak{e}_z+k} \otimes \Omega_2. \tag{39}$$

Hence, Eq. (38) and Eq. (39) lead to

$$-(2-u)\times 2^{e_0} \leq -2^{p-\mathfrak{e}_z+k} \otimes g_1(x) \leq -(2-u)\times 2^{-1+p+k}. \tag{40}$$

By Lemma 24, $-(2-u)\times 2^{-1+p+k} \oplus (2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}$ and $-(2-u)\times 2^{-1+p+k} \oplus (2-\mathfrak{a}_z-u)\times 2^{p+k}$ are exact. Together with Eq. (38) and Eq. (39), we have

$$-(2-u)\times 2^{e_0} \leq \zeta_{1,1}(x) \leq -(2(\mathfrak{a}_z-1)-u+2\tilde{u})\times 2^{-1+p+k} \leq 0, \tag{41}$$

$$-(2-u)\times 2^{e_0} \leq \zeta_{1,2}(x) \leq -(2(\mathfrak{a}_z-1)+u)\times 2^{-1+p+k} < 0, \tag{42}$$

which leads to $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 0$.

If $2^{\mathfrak{e}_z} \leq x \leq (2-2u)\times 2^{\mathfrak{e}_z}$, by similar arguments in **Case 1-2** and $e_{min} < -1+p+k, p+k < e_{max}$, we have the same results as those in Eqs. (29)–(32) from **Case 1-2**. Since $\mu(2^{p-\mathfrak{e}_z+k}) \otimes g_1(x) \geq k$, by Lemma 24 all operations in $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q})$ are exact. Hence

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k}\times \mathrm{ReLU}\big(2^{p-\mathfrak{e}_z+k}\times x - ((\mathfrak{a}_z-u+\tilde{u})\times 2^{p+k})\big), \tag{43}$$

$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k}\times \mathrm{ReLU}\big(2^{p-\mathfrak{e}_z+k}\times x - (\mathfrak{a}_z\times 2^{p+k})\big), \tag{44}$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) \in [2^p]$. Therefore,

$$f_1(x;\mathbb{F}_{p,q}) = \begin{cases} 0 & \text{if } (2-u)\times 2^{-1+\mathfrak{e}_z} \leq x \leq z^-, \\ 1 & \text{if } z \leq x \leq (2-2u)\times 2^{\mathfrak{e}_z}. \end{cases}$$

If $(2-u)\times 2^{\mathfrak{e}_z} \leq x \leq \Omega$, we have $g_1(x) = 0$. Hence

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \otimes ((2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}), \tag{45}$$

$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \otimes ((2-\mathfrak{a}_z-u_0)\times 2^{p+k}), \tag{46}$$

$$f_1(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 2^{p+\tilde{c}-c_z}\times(u_0-\tilde{u}_0) = 1, \tag{47}$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\}\cup[2^p]$.

Representability test for Case 1-3. (1) We check that the parameters in Eq. (11) are representable by $\mathbb{F}_{p,q}$. Since

$$e_{min} \leq p - e_0 \leq \tilde{c} - c_z - k \leq 1 - (-p+e_0) = 1 + p - e_0 < e_{max},$$
$$-p + e_{min} \leq 1 - p - e_{max} = p - (-2-p+e_{max}) + (-p+e_0)$$
$$\leq p - \mathfrak{e}_z + k = p - \mathfrak{e}_z + (-p+e_0) = e_0 - \mathfrak{e}_z < e_{max},$$
$$e_{min} < p + k = p + (-p+e_0) = e_0 < e_{max},$$

$2^{\tilde{c}-c_z-k}$, $2^{p-\mathfrak{e}_z+k}$, $(2-\mathfrak{a}_z-\tilde{u}_0)\times 2^{p+k}$, and $(2-\mathfrak{a}_z-u_0)\times 2^{p+k}$ are representable by $\mathbb{F}_{p,q}$. Apparently, $(2-u_0)\times 2^{\mathfrak{e}_z}$ is representable by $\mathbb{F}_{p,q}$.

(2) We check that the intermediate numbers in Eq. (12) are representable by $\mathbb{F}_{p,q}$, which is apparent by Eqs. (38) and (40)–(47). Hence, we conclude

$$f_1(x;z) \begin{cases} = 0 & \text{if } -\Omega_2 \leq x \leq z^-, \\ = 1 & \text{if } z \leq x \leq \Omega. \end{cases} \tag{48}$$

Finally, combining **Case 1-1**, **Case 1-2**, and **Case 1-3**, since $\Omega_1 \geq (2-u)\times 2^{-2+e_0}$ and $\Omega_2 \geq (2-u)\times 2^{2+e_0}$, we conclude that

$$f_1(x;z) \begin{cases} = 0 & \text{if } -(2-u)\times 2^{-2+e_0} \leq x \leq z^-, \\ = 1 & \text{if } z \leq x \leq \Omega, \end{cases} \tag{49}$$

for all z satisfying $0 < z \leq (2-u)\times 2^{-2-p+e_{max}}$.

To construct $\mathbf{1}[x \leq z]$, we define a three-layer ReLU network $f_2(x;\mathbb{F}_{p,q})$ as follows:

$$f_2(x;\mathbb{F}_{p,q}) = \psi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) \oplus \psi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}), \tag{50}$$

where

$$\psi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}),\ \psi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}),$$
$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}\big((-2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x))) \oplus (\mathfrak{a}_z-1+u_0)\times 2^{p+k}\big),$$
$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}\big((-2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x))) \oplus (\mathfrak{a}_z-1)\times 2^{p+k}\big).$$

Here, $g_2(x)$ is defined as

$$g_2(x) = x \ominus 2^{\mathfrak{e}_z}.$$

Let

$$\zeta_{2,1}(x) := (-2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x))) \oplus (\mathfrak{a}_z-1+u_0)\times 2^{p+k},$$
$$\zeta_{2,2}(x) := (-2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x))) \oplus (\mathfrak{a}_z-1)\times 2^{p+k},$$
$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}(\zeta_{2,1}(x)),$$
$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}(\zeta_{2,2}(x)).$$

Note that (1) the following are parameters in $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q})$:

$$2^{-c_z-k}, 2^{p-\mathfrak{e}_z+k}, 2^{\mathfrak{e}_z}, (\mathfrak{a}_z-1+u_0)\times 2^{p+k}, (\mathfrak{a}_z-1)\times 2^{p+k}. \tag{51}$$

The following are (2) the numbers occurring during the intermediate calculations in $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q})$:

$$g_2(x), -2^{p-\mathfrak{e}_z+k} \otimes g_2(x), \zeta_{2,1}(x), \zeta_{2,2}(x), \phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}), \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}). \tag{52}$$

We consider the following cases.

Case 2-1: $-p + e_{min} \leq \mathfrak{e}_z \leq 0$.

In this case, we have $k = 3 - p - e_0 - c_z$.

If $-\Omega \leq x \leq 2^{\mathfrak{e}_z}$, we have $g_2(x) = 0$. Hence we have

$$\zeta_{2,1}(x) = (\mathfrak{a}_z-1+u_0)\times 2^{p+k},\ \zeta_{2,2}(x) = (\mathfrak{a}_z-1)\times 2^{p+k}, \tag{53}$$

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}((\mathfrak{a}_z-1+u_0)\times 2^{p+k}) = (\mathfrak{a}_z-1+u_0)\times 2^{p-c_z}, \tag{54}$$

$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}((\mathfrak{a}_z-1)\times 2^{p+k}) = -(\mathfrak{a}_z-1)\times 2^{p-c_z}, \tag{55}$$

$$f_2(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k}\times u_0 = 1, \tag{56}$$

with $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\}\cup[2^p]$.

If $(1+u_0)\times 2^{\mathfrak{e}_z} \leq x \leq (2-u_0)\times 2^{\mathfrak{e}_z}$, $x \ominus 2^{\mathfrak{e}_z}$ is exact by Lemma 13. Then $g_2(x) = x - 2^{\mathfrak{e}_z} = n_2\times 2^{-p+\mathfrak{e}_z+c_z}$ where $n_2 \in [2^{p-c_z}-1]$, which leads to

$$0 < 2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x)) \leq (2-u_0)\times 2^{-1+p+k}, \tag{57}$$

$$\mu(2^{p-\mathfrak{e}_z+k} \otimes \mathrm{ReLU}(g_2(x))) \geq k. \tag{58}$$

Since

$$0 \leq (\mathfrak{a}_z-1)\times 2^{p+k}, (\mathfrak{a}_z-1+u_0)\times 2^{p+k} < 2^{1+p+k}, \tag{59}$$

by Lemma 24, all operations in $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q})$ are exact. Hence,

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}\big(-2^{p-\mathfrak{e}_z+k}\times x + (\mathfrak{a}_z+u_0)\times 2^{p+k}\big), \tag{60}$$

$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}\big(-2^{p-\mathfrak{e}_z+k}\times x + \mathfrak{a}_z\times 2^{p+k}\big), \tag{61}$$

with $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\}\cup[2^p]$. Therefore, we have

$$f_2(x;\mathbb{F}_{p,q}) = \begin{cases} 1 & \text{if } (1+u_0)\times 2^{\mathfrak{e}_z} \leq x \leq z, \\ 0 & \text{if } z^+ \leq x \leq (2-u_0)\times 2^{\mathfrak{e}_z}. \end{cases}$$

Suppose $2^{1+\mathfrak{e}_z} \le x \le \Omega_1$. If $2^{1+\mathfrak{e}_z} \le x < 2^{2+p+\mathfrak{e}_z}$, we have $2^{\mathfrak{e}_z} \le x \ominus 2^{\mathfrak{e}_z} < 2^{2+p+\mathfrak{e}_z}$. If $2^{2+p+\mathfrak{e}_z} \le x \le \Omega_1$, we have $x \ominus 2^{\mathfrak{e}_z} = x$ by Lemma 26. Therefore, we have

$$2^{\mathfrak{e}_z} \le g_2(x) \le \Omega_1, \tag{62}$$

By Lemma 24, $-2^{p+k} \oplus (\mathfrak{a}_z - 1 + u_0) \times 2^{p+k}$ and $-2^{p+k} \oplus (\mathfrak{a}_z - 1) \times 2^{p+k}$ are exact. Together with Eq. (25) and Eq. (63), we have

$$2^{\mathfrak{e}_z} \le g_2(x) \le \Omega_1,$$

$$-(2-u) \times 2^{4+e_0} \le -2^{p-\mathfrak{e}_z+k} \otimes g_2(x) \le -2^{p+k}, \tag{63}$$

$$-(2-u) \times 2^{4+e_0} \le \zeta_{2,1}(x) \le -(2-\mathfrak{a}_z-u_0) \times 2^{p+k} \le 0, \tag{64}$$

$$-(2-u) \times 2^{4+e_0} \le \zeta_{2,2}(x) \le -(2-\mathfrak{a}_z) \times 2^{p+k} < 0, \tag{65}$$

which leads to $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = 0$.

Representability test for Case 2-1.

(1) We check that the parameters in Eq. (51) are representable by $\mathbb{F}_{p,q}$. Since $2^{-c_z-k}, 2^{p-\mathfrak{e}_z}$ are shown to be representable by $\mathbb{F}_{p,q}$ in **Case 1-1** and **Case 1-2** and $2^{\mathfrak{e}_z}$ is obvious, we only need to check $(\mathfrak{a}_z - 1 + u_0) \times 2^{p+k}$ and $(\mathfrak{a}_z - 1) \times 2^{p+k}$. Since

$$e_{min} = -2^{q-1} + 2 \le 4 - p - 2^{q-2} \le 3 - p - e_0 \le p + k = 3 - e_0 - c_z < e_{max},$$
$$e_{min} < 3 - p - e_0 = (-p + c_z) + (p + k) \le \mu(t) \le \mathfrak{e}_t \le p + k < e_{max},$$
$$\mathfrak{e}_t - \mu(t) \le (p + k) - (-(p - c_z) + (p + k)) = p - c_z,$$

where $\mathfrak{a}_x \ne 1$ for $t = (\mathfrak{a}_z - 1 + u_0) \times 2^{p+k}, (\mathfrak{a}_z - 1) \times 2^{p+k}$, they are representable by $\mathbb{F}_{p,q}$. If $\mathfrak{a}_x = 1$, it is obvious.

(2) We check that the intermediate numbers in Eq. (52) are representable by $\mathbb{F}_{p,q}$, which is apparent by Eqs. (53)–(65). Therefore, we conclude

$$f_2(x;\mathbb{F}_{p,q}) \begin{cases} = 1 & \text{if } -\Omega \le x \le z, \\ = 0 & \text{if } z^+ \le x \le \Omega_1. \end{cases} \tag{66}$$

Case 2-2: $0 < \mathfrak{e}_z \le -2 - p + e_{max}$.

In this case, $c_z = 0, u_0 = u$, and $k = -p + e_0$.

If $-\Omega \le x \le 2^{\mathfrak{e}_z}$, we have $g_2(x) = 0$. Similar to **Case 2-1**, we have

$$\zeta_{2,1}(x) = (\mathfrak{a}_z - 1 + u) \times 2^{p+k},\ \zeta_{2,2}(x) = (\mathfrak{a}_z - 1) \times 2^{p+k}, \tag{67}$$

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}((\mathfrak{a}_z - 1 + u) \times 2^{p+k}) = (\mathfrak{a}_z - 1 + u) \times 2^{p-c_z}, \tag{68}$$

$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}((\mathfrak{a}_z - 1) \times 2^{p+k}) = -(\mathfrak{a}_z - 1) \times 2^{p-c_z}, \tag{69}$$

$$f_2(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \times u = 1, \tag{70}$$

with $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$.

If $(1+u) \times 2^{\mathfrak{e}_z} \le x \le (2-u) \times 2^{\mathfrak{e}_z}$, by similar arguments in **Case 2-1** with $e_{min} \le p+k \le e_{max}$, we have the same results as those in Eqs. (57)–(59) from **Case 2-1**. By Lemma 24, all operations in $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q})$ and $\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q})$ are exact. Hence,

$$\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = 2^{-c_z-k} \otimes \mathrm{ReLU}\big(-2^{p-\mathfrak{e}_z+k} \times x + (\mathfrak{a}_z + u) \times 2^{p+k}\big), \tag{71}$$

$$\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = -2^{-c_z-k} \otimes \mathrm{ReLU}\big(-2^{p-\mathfrak{e}_z+k} \times x + \mathfrak{a}_z \times 2^{p+k}\big), \tag{72}$$

with $|\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q})|, |\phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q})| \in \{0\} \cup [2^p]$. Therefore, we have

$$f_2(x;\mathbb{F}_{p,q}) = \begin{cases} 1 & \text{if } (1+u) \times 2^{\mathfrak{e}_z} \le x \le z, \\ 0 & \text{if } z^+ \le x \le (2-u) \times 2^{\mathfrak{e}_z}. \end{cases}$$

$$\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) = 2^{\tilde{c}-c_z-k} \otimes ((2 - \mathfrak{a}_z - \tilde{u}_0) \times 2^{p+k}), \tag{73}$$

$$\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = -2^{\tilde{c}-c_z-k} \otimes ((2 - \mathfrak{a}_z - u) \times 2^{p+k}), \tag{74}$$

$$f_1(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) = 2^{p+\tilde{c}-c_z} \times (u - \tilde{u}_0) = 1, \tag{75}$$

with $\phi_{\theta_{1,1,z}}(x;\mathbb{F}_{p,q}), -\phi_{\theta_{1,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$.

Suppose $2^{1+\mathfrak{e}_z} \le x \le \Omega_2$. If $2^{1+\mathfrak{e}_z} \le x < \min\{\Omega_2, 2^{2+p+\mathfrak{e}_z}\}$, we have $2^{\mathfrak{e}_z} \le x \ominus 2^{\mathfrak{e}_z} < \min\{\Omega_2, 2^{2+p+\mathfrak{e}_z}\}$. If $\min\{\Omega_2, 2^{2+p+\mathfrak{e}_z}\} \le x \le \Omega_2$, we have $x \ominus 2^{\mathfrak{e}_z} = x$ by Lemma 26. Therefore, we have

$$2^{\mathfrak{e}_z} \le g_2(x) \le \Omega_2, \tag{76}$$

By Lemma 24, $-2^{p+k} \oplus (\mathfrak{a}_z - 1 + u) \times 2^{p+k}$ and $-2^{p+k} \oplus (\mathfrak{a}_z - 1) \times 2^{p+k}$ are exact. Together with Eq. (39) and Eq. (76), we have

$$2^{\mathfrak{e}_z} \le g_2(x) \le \Omega_2,$$

$$-(2-u) \times 2^{e_0} \le -2^{p-\mathfrak{e}_z+k} \otimes g_2(x) \le -2^{p+k}, \tag{77}$$

$$-(2-u) \times 2^{e_0} \le \zeta_{2,1}(x) \le -(2-\mathfrak{a}_z-u) \times 2^{p+k} \le 0, \tag{78}$$

$$-(2-u) \times 2^{e_0} \le \zeta_{2,2}(x) \le -(2-\mathfrak{a}_z) \times 2^{p+k} < 0, \tag{79}$$

which leads to $\phi_{\theta_{2,1,z}}(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,2,z}}(x;\mathbb{F}_{p,q}) = 0$.

Representability test for Case 2-2.

(1) We check that the parameters in Eq. (51) are representable by $\mathbb{F}_{p,q}$. Since $2^{-c_z-k}, 2^{p-\mathfrak{e}_z}$ are shown to be representable by $\mathbb{F}_{p,q}$ in **Case 1-3** and $2^{\mathfrak{e}_z}$ is obvious, we only need to check $(\mathfrak{a}_z - 1 + u) \times 2^{p+k}$ and $(\mathfrak{a}_z - 1) \times 2^{p+k}$. Since

$$e_{min} < p + k = e_0 < e_{max},$$
$$e_{min} < -p + e_0 = (-p + c_z) + (p + k) \le \mu(t) \le \mathfrak{e}_t \le p + k < e_{max},$$
$$\mathfrak{e}_t - \mu(t) \le (p + k) - (-(p - c_z) + (p + k)) = p - c_z,$$

where $\mathfrak{a}_x \ne 1$ for $t = (\mathfrak{a}_z - 1 + u) \times 2^{p+k}, (\mathfrak{a}_z - 1) \times 2^{p+k}$, they are representable by $\mathbb{F}_{p,q}$. If $\mathfrak{a}_x = 1$, it is obvious.

(2) We check that the intermediate numbers in Eq. (52) are representable by $\mathbb{F}_{p,q}$, which is apparent by Eqs. (67)–(79). Therefore, we conclude

$$f_2(x;\mathbb{F}_{p,q}) \begin{cases} = 1 & \text{if } -\Omega \le x \le z, \\ = 0 & \text{if } z^+ \le x \le \Omega_2. \end{cases} \tag{80}$$

Finally, combining **Case 2-1** and **Case 2-2**, since $\Omega_1 \ge (2-u) \times 2^{-2+e_0}$ and $\Omega_2 \ge (2-u) \times 2^{2+e_0}$, we conclude that

$$f_2(x;\mathbb{F}_{p,q}) \begin{cases} = 1 & \text{if } -\Omega \le x \le z, \\ = 0 & \text{if } z^+ \le x \le (2-u) \times 2^{-2+e_0}, \end{cases} \tag{81}$$

for all z satisfying $0 < z \le (2-u) \times 2^{-2-p+e_{max}}$.

Now, we consider $z < 0$, we define

$$f_1(x;\mathbb{F}_{p,q}) = \psi_{1,1,z}(x;\mathbb{F}_{p,q}) \oplus \psi_{1,2,z}(x;\mathbb{F}_{p,q}),$$
$$f_2(x;\mathbb{F}_{p,q}) = \psi_{2,1,z}(x;\mathbb{F}_{p,q}) \oplus \psi_{2,2,z}(x;\mathbb{F}_{p,q}),$$
$$\psi_{1,1,z}(x;\mathbb{F}_{p,q}) = \phi_{2,1,-z}(-x;\mathbb{F}_{p,q}),\ \psi_{1,2,z}(x;\mathbb{F}_{p,q}) = \phi_{2,2,-z}(-x;\mathbb{F}_{p,q}),$$
$$\psi_{2,1,z}(x;\mathbb{F}_{p,q}) = \phi_{1,1,-z}(-x;\mathbb{F}_{p,q}),\ \psi_{2,2,z}(x;\mathbb{F}_{p,q}) = \phi_{1,2,-z}(-x;\mathbb{F}_{p,q}).$$

Then we have

$$f_1(x;\mathbb{F}_{p,q}) = \phi_{\theta_{2,1,-z}}(-x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{2,2,-z}}(-x;\mathbb{F}_{p,q})$$
$$= \begin{cases} 0 & \text{if } -(2-u) \times 2^{-2+e_0} \le x \le z^-, \\ 1 & \text{if } z \le x \le \Omega, \end{cases}$$

$$f_2(x;\mathbb{F}_{p,q}) = \phi_{\theta_{1,1,-z}}(-x;\mathbb{F}_{p,q}) \oplus \phi_{\theta_{1,2,-z}}(-x;\mathbb{F}_{p,q})$$
$$= \begin{cases} 1 & \text{if } -\Omega \le x \le z, \\ 0 & \text{if } z^+ \le x \le (2-u) \times 2^{-2+e_0}, \end{cases}$$

for all z satisfying $-(2-u) \times 2^{-2-p+e_{max}} \le z < 0$.

Finally, we have $\psi_{\theta_{i,1,z}}(x;\mathbb{F}_{p,q}). - \psi_{\theta_{i,2,z}}(x;\mathbb{F}_{p,q}) \in \{0\} \cup [2^p]$ for all $i \in \{1,2\}$, and

$$f_1(x;\mathbb{F}_{p,q}) = \begin{cases} 0 & \text{if } -(2-u) \times 2^{-2+e_0} \le x \le z^-, \\ 1 & \text{if } z \le x \le \Omega, \end{cases}$$

$$f_2(x;\mathbb{F}_{p,q}) = \begin{cases} 1 & \text{if } -\Omega \le x \le z, \\ 0 & \text{if } z^+ \le x \le (2-u) \times 2^{-2+e_0}, \end{cases}$$

for all z satisfying $0 < |z| \le (2-u) \times 2^{-2-p+e_{max}}$.

Lastly, it is easy to observe that $\psi_{\theta_{i,j,z}}$ for all $i, j \in \{1,2\}$ and $z \in \mathbb{F}_{p,q} \setminus \{0\}$ share the same network architecture of 3 layers and 5 parameters. This completes the proof. □

6.3. Proof of Theorem 8

The proof is almost identical to Theorem 2.

We define f_{θ_D} as follows.

$$f_{\theta_D}(\mathbf{x};\mathbb{F}_{p,q}) = \bigoplus_{i=1}^{n}\left(y_i \otimes f_{\theta_{\mathbf{z}_i,\mathbf{z}_i}}(\mathbf{x},\mathbb{F}_{p,q})\right),$$

$$f_{\theta_{\alpha,\beta}}(\mathbf{x},\mathbb{F}_{p,q}) = \mathbb{1}\left[\bigoplus_{i=1}^{2d}\left(g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q})\right)\ominus 2d\right],\ \forall \alpha,\beta\in\mathbb{F}_{p,q}^d,$$

$$g_{2j-1}(x_j;\mathbb{F}_{p,q}) = \mathbb{1}\left[-x_j+\beta_j\geq 0\right],\ g_{2j}(x_j,\mathbb{F}_{p,q}) = \mathbb{1}\left[x_j\ominus\alpha_j\geq 0\right],\ \forall j\in[d].$$

From the definition of g_i, for $i\in[2d]$, one can observe that

$$g_{2j-1}(x_j;\mathbb{F}_{p,q}) + g_{2j}(x_j;\mathbb{F}_{p,q}) = \begin{cases} 2 & \text{if } x_j\in[\alpha_j,\beta_j],\\ 1 & \text{if } x_j\notin[\alpha_j,\beta_j].\end{cases}$$

Since $\bigoplus_{i=1}^{m} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q}) \leq m \leq 2^{1+p}-1$ for $1\leq m\leq(2d-1)$, and $|g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q})| = 0,1$, by Lemma 25, $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q})$ is exact. Therefore, we have $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q}) = 2d$ if $\mathbf{x}\in\prod_{i=1}^{d}[\alpha_i,\beta_i]$ and $\bigoplus_{i=1}^{2d} g_i(x_{\lceil i/2\rceil_{\mathbb{Z}}};\mathbb{F}_{p,q}) < 2d$ otherwise, i.e.,

$$f_{\theta_{\alpha,\beta}}(\mathbf{x},\mathbb{F}_{p,q}) = \mathbb{1}\left[\mathbf{x}\in\prod_{j=1}^{d}[\alpha_j,\beta_j]\right],$$

$$f_{\theta_{\mathbf{z}_i,\mathbf{z}_i}}(\mathbf{x},\mathbb{F}_{p,q}) = \mathbb{1}\left[\mathbf{x}\in\prod_{j=1}^{d}[\mathbf{z}_{i,j},\mathbf{z}_{i,j}]\right] = \mathbb{1}\left[\mathbf{x}=\mathbf{z}_i\right].$$

Then, $f_{\theta_D}(\mathbf{z}_i;\mathbb{F}_{p,q}) = y_i$ for all $i\in[n]$ and f can be implemented by a three-layer Step network of $6dn+2n$ parameters ($4dn$ parameters for the first layer, $2dn+n$ parameters for the second layer, and n parameters for the last layer).

6.4. Proof of Theorem 9

The proof is almost identical to Theorem 3.

If $\omega_{f^*}^{-1}(\varepsilon)\geq\eta$, for $\mathbf{x},\mathbf{x}'\in\mathbb{F}_{p,q}$ with $\|\mathbf{x}-\mathbf{x}'\|_\infty\leq\delta=\omega_{f^*}^{-1}(\varepsilon)$, we have $|f^*(\mathbf{x})-f^*(\mathbf{x}')|\leq\varepsilon$. If $\eta>\omega_{f^*}^{-1}(\varepsilon)$, for $\mathbf{x},\mathbf{x}'\in\mathbb{F}_{p,q}$ with $\|\mathbf{x}-\mathbf{x}'\|_\infty\leq\delta<\eta$, we have $|f^*(\mathbf{x})-f^*(\mathbf{x}')| = 0$ since $\mathbf{x}=\mathbf{x}'$. Hence for $\mathbf{x},\mathbf{x}'\in\mathbb{F}_{p,q}$ with $\|\mathbf{x}-\mathbf{x}'\|_\infty\leq\delta$, we have

$$|f^*(\mathbf{x})-f^*(\mathbf{x}')|\leq\varepsilon. \tag{82}$$

Now, for each $i\in[K]$, we define

$$\alpha_i = \begin{cases} i\delta & \text{if } i\in\{0,1,\dots,K-1\},\\ 1 & \text{if } i=K,\end{cases}$$

$$\mathcal{I}_i = \begin{cases} [\alpha_{i-1}^{(\geq,)},\alpha_i^{(<,\mathbb{F}_{p,q})}]\cap\mathbb{F}_{p,q} & \text{if } i\in[K-1],\\ [\alpha_{K-1}^{(\geq,\mathbb{F}_{p,q})},\alpha_K^{(\leq,\mathbb{F}_{p,q})}]\cap\mathbb{F}_{p,q} & \text{if } i=K.\end{cases}$$

Without loss of generality, we assume that $\mathcal{I}_i\neq\emptyset$ for all $i\in[K]$; otherwise, we remove empty $\mathcal{I}_j$, decrease K, and re-index $\mathcal{I}_i$ so that $\mathcal{I}_i$ if non-empty for all $i\in[K]$. We note that since $0,1\in\mathbb{F}_{p,q}$, there is at least one non-empty $\mathcal{I}_i$ and $K\geq 1$. Additionally, unlike in $\mathbb{F}_p$, in $\mathbb{F}_{p,q}$ $\mathcal{I}_i$ is finite due to the existence of the smallest positive number. Then, by the above definitions, it holds that $\sup\mathcal{I}_i-\inf\mathcal{I}_i\leq\delta$, $\mathcal{I}_{i_1}$ and $\mathcal{I}_{i_2}$ are disjoint if $i_1\neq i_2$, and $\bigcup_{i\in[K]}\mathcal{I}_i=[0,1]\cap\mathbb{F}_{p,q}$. For each $\boldsymbol{\iota}=(\iota_1,\dots,\iota_d)\in[K]^d$, we also define

$$\boldsymbol{\gamma}_{\boldsymbol{\iota}} = \underset{\mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}}{\arg\min}\ |f^*(\mathbf{x})-\lceil f^*(\mathbf{x})\rfloor|,$$

which is well-defined since $\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}$ is non-empty and finite.

We are now ready to introduce our Step network construction f_θ:

$$f_\theta(\mathbf{x};\mathbb{F}_{p,q}) = \bigoplus_{\boldsymbol{\iota}\in[K]^d}\lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})\rfloor\otimes\mathbb{1}\left[\left(\bigoplus_{j=1}^{2d}h_{\boldsymbol{\iota},j}(x_{\lceil j/2\rceil_{\mathbb{Z}}})\right)\ominus d\geq 0\right],$$

where for each $j\in[d]$ and $\boldsymbol{\iota}=(\iota_1,\dots,\iota_d)\in[K]^d$,

$$h_{\boldsymbol{\iota},2j-1}(x) = \begin{cases}\mathbb{1}\left[x-\alpha_{j-1}^{(\geq,\mathbb{F}_{p,q})}\geq 0\right] & \text{if } \iota_j\in\{2,\dots,K\},\\ \mathbb{1}\left[x+\alpha_0^{(\geq,\mathbb{F}_{p,q})}\geq 0\right] & \text{if } \iota_j=1,\end{cases}$$

$$h_{\boldsymbol{\iota},2j}(x) = \begin{cases}-\mathbb{1}\left[x-\alpha_j^{(\geq,\mathbb{F}_{p,q})}\geq 0\right] & \text{if } \iota_j\in\{1,\dots,K-1\},\\ -\mathbb{1}\left[x-\alpha_K^{(>,\mathbb{F}_{p,q})}\leq 0\right] & \text{if } \iota_j=K.\end{cases}$$

Since $h_{\boldsymbol{\iota},j}(x)\in\{-1,0,1\}$, $h_{\boldsymbol{\iota},2j-1}(x)+h_{\boldsymbol{\iota},2j}(x)=\{-1,0,1\}$, we have

$$\left|\bigoplus_{j=1}^{m}h_{\boldsymbol{\iota},j}(x_{\lceil j/2\rceil_{\mathbb{Z}}})\right|\leq d\leq 2^p.$$

for any $1\leq m\leq 2d$. Therefore, by Lemma 25, all operations in the computation of $\bigoplus_{j=1}^{2d}h_{\boldsymbol{\iota},j}(x_{\lceil j/2\rceil_{\mathbb{Z}}})$ are exact. Furthermore, for each $\mathbf{x}\in\mathbb{F}_{p,q}^d\cap[0,1]^d$, we have

$$\bigoplus_{j=1}^{2d}h_{\boldsymbol{\iota},j}(x_{\lceil j/2\rceil_{\mathbb{Z}}})\begin{cases}=d & \text{if } \mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d},\\ <d & \text{if } \mathbf{x}\in([0,1]^d\cap\mathbb{F}_{p,q}^d)\setminus(\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}).\end{cases}$$

Since $\bigcup_{\boldsymbol{\iota}\in[K]^d}\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}=\mathbb{F}_{p,q}^d\cap[0,1]^d$, we have

$$f_\theta(\mathbf{x};\mathbb{F}_{p,q}) = \lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})\rfloor,\ \forall\mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}.$$

Hence, for each $\boldsymbol{\iota}\in[K]^d$ and $\mathbf{x}\in\mathcal{I}_{\iota_1}\times\cdots\times\mathcal{I}_{\iota_d}$,

$$\begin{aligned}|f_\theta(\mathbf{x};\mathbb{F}_{p,q})-f^*(\mathbf{x})| &= |\lceil f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})\rfloor-f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})|+|f^*(\boldsymbol{\gamma}_{\boldsymbol{\iota}})-f^*(\mathbf{x})|\\ &\leq|\lceil f^*(\mathbf{x})\rfloor-f^*(\mathbf{x})|+\varepsilon,\end{aligned}$$

where we use Eq. (82) for the above inequality. Since each $h_{\boldsymbol{\iota},j}$ can be implemented by a Step network of 3 parameters, $\mathbb{1}\left[\left(\bigoplus_{j=1}^{2d}h_{\boldsymbol{\iota},j}(x_{\lceil j/2\rceil_{\mathbb{Z}}})\right)\ominus d\geq 0\right]$ can be implemented using $6d+1$ parameters. This implies that our f_θ can be implemented by a Step network of 3 layers and $(6d+2)K^d$ parameters. This completes the proof.

6.5. Proof of Theorem 10

Recall that $\eta=2^{-p+e_{max}}$ is the smallest positive number in $\mathbb{F}_{p,q}$. For each $i\in[n]$, we also define $h_{i,1},\dots,h_{i,4d}$ as follows: for each $j\in[d]$, if $z_{i,j}\neq 0$,

$$h_{i,4j-3}=\psi_{\theta_{1,1,t_{i,j,1}}},\ h_{i,4j-2}=\psi_{\theta_{1,2,t_{i,j,1}}},$$

$$h_{i,4j-1}=-\psi_{\theta_{1,1,t_{i,j,2}}},\ h_{i,4j}=-\psi_{\theta_{1,2,t_{i,j,2}}},$$

where $t_{i,j,1}=z_{i,j}$ and $t_{i,j,2}=z_{i,j}^+$. If $z_{i,j}=0$,

$$h_{i,4j-3}=-\psi_{\theta_{2,1,t_{i,j,1}}},\ h_{i,4j-2}=-\psi_{\theta_{2,2,t_{i,j,1}}},$$

$$h_{i,4j-1}=-\psi_{\theta_{1,1,t_{i,j,2}}},\ h_{i,4j}=-\psi_{\theta_{1,2,t_{i,j,2}}},$$

where $t_{i,j,1}=-\eta$ and $t_{i,j,2}=\eta$, and $\psi_{\theta_{1,1,z}},\psi_{\theta_{1,2,z}}$ are defined in Lemma 27.

From Lemma 27, the signs of $h_{i,4j-3}(x),h_{i,4j-2}(x)$ are different. The signs of $h_{i,4j-1}(x),h_{i,4j}(x)$ are also different. Let $\mathfrak{s}_{4j-3}=\mathfrak{s}_{h_{i,4j-3}(x)},\mathfrak{s}_{4j-1}=\mathfrak{s}_{h_{i,4j-1}(x)}$. Since $|z_{i,j}|<(2-u)\times 2^{-2-p+e_{max}}$ for all $i\in[n],j\in[d]$, by Lemma 27, we have

$$|h_{i,4j-3}(x)|,|h_{i,4j-2}(x)|,|h_{i,4j-1}(x)|,|h_{i,4j}(x)|\in\{0\}\cup[2^p],$$

$$\bigoplus_{k=1}^{4}h_{i,4j-4+k}(x)=\begin{cases}\mathbb{1}\left[x=z_{i,j}\right] & \text{if } z_{i,j}\neq 0,\\ -\mathbb{1}[x\leq-\eta]-\mathbb{1}[x\geq\eta]=\mathbb{1}[x=0]-1 & \text{if } z_{i,j}=0,\end{cases}$$

for all $x\in\mathbb{F}_{p,q}$ with $|x|\leq(2-u)\times 2^{-3+2^{q-2}}$.

Let $0\leq m<d\leq 2^p$. For m, suppose $\mathfrak{s}_{4m+1}=1$. Then we have $h_{i,4m+1}(x)+h_{i,4m+2}(x)=l_{4m+1}\in\{0,1\}$ and

$$-2^p<-m\leq\bigoplus_{j=1}^{4m}h_{i,j}(x)\leq m<2^p,\ h_{i,4m+1}(x)\in\{0\}\cup[2^p],$$

$$-2^p \le -m + h_{i,4m+1}(x) \le \bigoplus_{j=1}^{4m+1} h_{i,j}(x) \le m + h_{i,4m+1}(x) < 2^{p+1},$$
$$-h_{i,4m+2}(x) \in \{0\} \cup [2^p],$$
$$-2^p < -m + l_{4m+1} \le \bigoplus_{j=1}^{4m+2} h_{i,j}(x) \le m + l_{4m+1} \le 2^p.$$

By Lemma 25, all above operations in $\bigoplus_{j=1}^{4m+2} h_{i,j}(x)$ are exact.

Suppose $\mathfrak{s}_{4m+1} = -1$. Then we have $h_{i,4m+1}(x) + h_{i,4m+2}(x) = l_{4m+1} \in \{0, -1\}$, and

$$-2^p < -m \le \bigoplus_{j=1}^{4m} h_{i,j}(x) \le m < 2^p, \ -h_{i,4m+1}(x) \in \{0\} \cup [2^p],$$
$$-2^{1+p} < -m + h_{i,4m+1}(x) \le \bigoplus_{j=1}^{4m+1} h_{i,j}(x) \le m + h_{i,4m+1}(x) \le 2^p,$$
$$h_{i,4m+2}(x) \in \{0\} \cup [2^p],$$
$$-2^p \le -m + l_{4m+1} \le \bigoplus_{j=1}^{4m+2} h_{i,j}(x) \le m + l_{4m+1} < 2^p.$$

By Lemma 25, all above operations in $\bigoplus_{j=1}^{4m+2} h_{i,j}(x)$ are exact.

Since $\mathfrak{s}_{4m+3} = -1$, we have $h_{i,4m+3}(x) + h_{i,4m+4}(x) = l_{4m+3} \in \{0, -1\}, l_{4m+1} + l_{4m+3} \in \{0, -1\}$, and

$$-2^p \le -m + l_{4m-1} \le \bigoplus_{j=1}^{4m+2} h_{i,j}(x) \le 2^p,$$
$$-h_{i,4m+3}(x) \in \{0\} \cup [2^p],$$
$$-2^{1+p} \le -m + l_{4m-1} + h_{i,4m+3}(x) \le \bigoplus_{j=1}^{4m+3} h_{i,j}(x) \le 2^p + h_{i,4m+3}(x),$$
$$h_{i,4m+4}(x) \in \{0\} \cup [2^p],$$
$$-2^p \le -m + l_{4m-1} + l_{4m-3} \le \bigoplus_{j=1}^{4m+4} h_{i,j}(x) < 2^p + l_{4m-3} \le 2^p.$$

By Lemma 25, all above operations in $\bigoplus_{j=1}^{4m+2} h_{i,j}(x)$ are exact.

Therefore we conclude that all operations in $\bigoplus_{j=1}^{4d} h_{i,j}(x)$ are exact with $|\bigoplus_{j=1}^{4d} h_{i,j}(x)| \in \{0\} \cup [2^p]$.

We design the target network f_θ as follows:

$$f_\theta(\mathbf{x}; \mathbb{F}_{p,q}) = \bigoplus_{i=1}^{n} y_i \otimes \mathrm{ReLU}\left(\left(\bigoplus_{j=1}^{4d} h_{i,j}(x_{\lceil j/4 \rceil_{\mathbb{Z}}})\right) \oplus b_i\right),$$

where $b_i = |j \in [d] : z_{i,j} = 0| - (d-1)$. Since for each $k \in [n]$

$$\bigoplus_{j=1}^{4d} h_{i,j}(z_{k,\lceil j/4 \rceil_{\mathbb{Z}}}) \begin{cases} = b_i + 1 & \text{if } \mathbf{z}_k = \mathbf{z}_i, \\ \le b_i & \text{if } \mathbf{z}_k \ne \mathbf{z}_i, \end{cases}$$

f_θ memorizes the target dataset. Since there are 5 parameters for each $h_{i,j}$, f_θ has $20dn + 2n$ parameters. This completes the proof.

6.6. Proof of Theorem 11

The proof of Theorem 11 is almost identical to that of Theorem 9; we define $\mathcal{I}_i$, α_i, and $\boldsymbol{\gamma}_\iota$ as in Section 6.4. For each $\iota \in [K]^d$, $j \in [d]$, we also define $h_{\iota,1}, \dots, h_{\iota,4d}$ as follows:

$$h_{\iota,4j-3} = \begin{cases} \psi_{\theta_{1,1,t_{\iota,j,1}}} & \text{if } \iota_j \in \{2, \dots, K\}, \\ -\psi_{\theta_{2,1,t_{\iota,j,1}}} & \text{if } \iota_j = 1, \end{cases}$$
$$h_{\iota,4j-2} = \begin{cases} \psi_{\theta_{1,2,t_{\iota,j,1}}} & \text{if } \iota_j \in \{2, \dots, K\}, \\ -\psi_{\theta_{2,2,t_{\iota,j,1}}} & \text{if } \iota_j = 1, \end{cases}$$
$$h_{\iota,4j-1} = -\psi_{\theta_{1,1,t_{\iota,j,2}}},$$
$$h_{\iota,4j} = -\psi_{\theta_{1,2,t_{\iota,j,2}}},$$
$$t_{\iota,j,1} = \begin{cases} -\eta & \text{if } \iota_j = 1, \\ \alpha_{j-1}^{(\ge,\mathbb{F}_{p,q})} & \text{if } \iota_j \in \{2, \dots, K\}, \end{cases}$$
$$t_{\iota,j,2} = \begin{cases} \alpha_j^{(\ge,\mathbb{F}_{p,q})} & \text{if } \iota_j \in \{1, \dots, K-1\}, \\ \alpha_K^{(>,\mathbb{F}_{p,q})} & \text{if } \iota_j = K. \end{cases}$$

and $\psi_{\theta_{1,1,z}}, \psi_{\theta_{1,2,z}}$ are defined in Lemma 27.

Namely, we have

$$h_{\iota,4j-3}(x) \oplus h_{\iota,4j-2}(x) = \begin{cases} -\mathbb{1}[x \le -\eta] & \text{if } \iota_j = 1, \\ \mathbb{1}\left[x \ge \alpha_{j-1}^{(\ge,\mathbb{F}_{p,q})}\right] & \text{if } \iota_j \in \{2, \dots, K\}, \end{cases}$$
$$h_{\iota,4j-1}(x) \oplus h_{\iota,4j}(x) = \begin{cases} -\mathbb{1}\left[x \ge \alpha_j^{(\ge,\mathbb{F}_{p,q})}\right] & \text{if } \iota_j \in \{1, \dots, K-1\}, \\ -\mathbb{1}\left[x \ge \alpha_K^{(>,\mathbb{F}_{p,q})}\right] & \text{if } \iota_j = K, \end{cases}$$
$$\bigoplus_{k=1}^{4} h_{\iota,4j-4+k}(x) = \begin{cases} \mathbb{1}\left[0 \le x < \alpha_1^{(\ge,\mathbb{F}_{p,q})}\right] - 1 & \text{if } \iota_j = 1, \\ \mathbb{1}\left[\alpha_{j-1}^{(\ge,\mathbb{F}_{p,q})} \le x < \alpha_j^{(\ge,\mathbb{F}_{p,q})}\right] & \text{if } \iota_j \in \{2, \dots, K-1\}, \end{cases}$$

for all $x \in \mathbb{F}_{p,q}$ with $|x| \le (2-u) \times 2^{-3+2q-2}$, by Lemmas 25 and 27. We design the target network f as follows:

$$f_\theta(\mathbf{x}; \mathbb{F}_{p,q}) = \bigoplus_{\iota \in [K]^d} \lceil f^*(\boldsymbol{\gamma}_\iota) \rfloor \otimes \mathbb{1}\left[\left(\bigoplus_{j=1}^{4d} h_{\iota,j}(x_{\lceil j/4 \rceil_{\mathbb{Z}}})\right) \oplus b_i \ge 0\right].$$

where $b_i = |j \in [d] : z_{i,j} = 0| - (d-1)$.

By similar argument presented in the proof of Theorem 10, all operations in $\bigoplus_{j=1}^{4d} h_{\iota,j}(x_{\lceil j/4 \rceil_{\mathbb{Z}}})$ are exact by Lemma 25, i.e., for each $k \in [n]$

$$\bigoplus_{j=1}^{4d} h_{\iota,j}(x_{\lceil j/4 \rceil_{\mathbb{Z}}}) \begin{cases} = -b_i + 1 & \text{if } \mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}, \\ \le -b_i & \text{if } ([0,1]^d \cap \mathbb{F}_{p,q}^d) \setminus (\mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}), \end{cases}$$

This implies that for each $\iota \in [K]^d$ and $\mathbf{x} \in \mathcal{I}_{\iota_1} \times \cdots \times \mathcal{I}_{\iota_d}$,

$$|f_\theta(\mathbf{x}; \mathbb{F}_{p,q}) - f^*(\mathbf{x})| = |\lceil f^*(\boldsymbol{\gamma}_\iota) \rfloor - f^*(\boldsymbol{\gamma}_\iota)| + |f^*(\boldsymbol{\gamma}_\iota) - f^*(\mathbf{x})|$$
$$\le |\lceil f^*(\mathbf{x}) \rfloor - f^*(\mathbf{x})| + \varepsilon,$$

where we use Eq. (82) for the above inequality. Since each $h_{\iota,j}$ can be implemented by a ReLU network of 3 layers and 5 parameters by Lemma 27, $\mathbb{1}\left[\left(\bigoplus_{j=1}^{4d} h_{\iota,j}(x_{\lceil j/4 \rceil_{\mathbb{Z}}})\right) \oplus b_i \ge 0\right]$ can be implemented using $20d + 1$ parameters. This implies that our f can be implemented by a ReLU network of 4 layers and $(20d+2)K^d$ parameters. This completes the proof.

7. Discussions

In this section, we discuss possible extensions of our research. One possible direction is to explore general activation functions, which may require more cautious inspections of floating-point operations. For example, in our ReLU network constructions, we approximate a Step function via a ReLU network of size $O(1)$ using the *flat region* of ReLU, i.e., $\mathrm{ReLU}(x) = 0$ for all $x \le 0$. However, even most piecewise linear activation functions, such as Leaky-ReLU, do not have such a flat region, i.e., our ReLU network constructions do not easily generalize.

We expect that networks using general smooth functions require more caution. This is because there are various different ways to implement each function in floating-point numbers. Consider the exponential function e^x, for instance. Mathematically, it can be computed via the Taylor series expansion as $e^x = 1 \oplus x \oplus (x \otimes x \otimes 2^{-1}) \oplus \cdots$. However, the result of floating-point operations can vary depending on their operation order, e.g., $((\frac{1}{6} \otimes x) \otimes x) \otimes x \ne \frac{1}{6} \otimes (x \otimes (x \otimes x))$. Additionally, a standard approach to implement the exponential function is to divide the input range into several regions and calculate the ouput differently in each region (Muller, 2016). Such a complicated

method is used to guarantee a small rounding error in the output; for instance, the GNU C library (Loosemore, Stallman, McGrath, Oram, & Drepper, 2024) claims that its implementation of various mathematical functions (e.g., $\exp, \log, \sin$) has a maximum error of 10 ulps[3] for most inputs. Therefore, when discussing the universal approximation property of a specific activation function in the floating-point setting, we should specify not just the activation itself but also its computation protocol and its guaranteed error bounds. It would be an interesting research topic to analyze the approximation capacity of each activation computing protocol.

We can also explore another arithmetic approach, integer arithmetic, which is utilized in quantized neural networks. Additionally, it will be an interesting research topic to investigate the expressive power of more general network architectures such as residual networks, and convolutional neural networks.

8. Conclusion

In this work, we investigate the expressive power of neural networks under the fully floating-point setting; all parameters of neural networks, inputs, and intermediate values are floating-point, and neural networks are represented as compositions of floating-point operations. Under unbounded exponent floating-point arithmetic $\mathbb{F}_p$, we first demonstrate that Step network has memorization universal approximation properties (Theorems 2 and 3). Following this, we establish that ReLU network also exhibits memorization universal approximation properties by constructing the indicator function using ReLU network (Theorems 5 and 6). Additionally, under bounded exponent floating-point arithmetic $\mathbb{F}_{p,q}$, we also demonstrate that Step and ReLU network possess memorization and universal approximation properties (Theorems 8–11). Because underflow and overflow in $\mathbb{F}_{p,q}$ presents challenges in constructing the network, we develop several technical lemmas to address this (Lemmas 23–27).

On the other hand, most prior works regarding universal approximation assume real-valued inputs and parameters and/or exact mathematical operations, which cannot be simulated by modern computers that can only represent a tiny subset of the real numbers and apply inexact operations. Considering that almost all neural networks are indeed implemented under floating-point machinery, it is important to theoretically analyze the properties of floating-point neural networks. To the best of our knowledge, this is the first work to tackle the universal approximation property under the floating-point setting. We believe that our results and analyses under floating-point operations would contribute to a better understanding of the performance of modern deep and narrow networks that are executed on actual computers.

CRediT authorship contribution statement

Yeachan Park: Writing – original draft, Methodology, Investigation, Formal analysis, Writing – review & editing. **Geonho Hwang:** Investigation, Methodology, Writing – original draft, Writing – review & editing. **Wonyeol Lee:** Conceptualization, Methodology, Writing – review & editing. **Sejun Park:** Conceptualization, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

[3] The ulp of $x \in \mathbb{R}$ is commonly defined as $\mathrm{ulp}(x) = 2^{\max\{e_{min}, \lfloor \log_2 |x| \rfloor\} - p}$ (Muller et al., 2018, Definition 2.6), and the ulp error of $y \in \mathbb{R}$ from x is defined as $\mathrm{err}_{\mathrm{ulp}}(x, y) = |x - y|/\mathrm{ulp}(x)$ (Goldberg, 1991).

Data availability

No data was used for the research described in the article.

Acknowledgments

YP was supported by a KIAS Individual Grant [AP090301] via the Center for AI and Natural Sciences at Korea Institute for Advanced Study. GH was supported by a KIAS Individual Grant [AP092801] via the Center for AI and Natural Sciences at Korea Institute for Advanced Study. SP was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00079, Artificial Intelligence Graduate School Program, Korea University) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korea government (2022R1F1A1076180).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

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

Baum, E. B. (1988). On the capabilities of multilayer perceptrons. *Journal of Complexity*.

Boldo, S. (2015). Stupid is as stupid does: Taking the square root of the square of a floating-point number. *Electronic Notes in Theoretical Computer Science*, *317*, 27–32.

Boldo, S., Jeannerod, C., Melquiond, G., & Muller, J. (2023). Floating-point arithmetic. *Acta Numerica*, *32*, 203–290.

Boldo, S., & Melquiond, G. (2011). Flocq: A unified library for proving floating-point algorithms in Coq. In *IEEE symposium on computer arithmetic* (pp. 243–252).

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, *2*(4), 303–314.

Ding, Y., Liu, J., Xiong, J., & Shi, Y. (2019). On the universal approximability and complexity bounds of quantized ReLU neural networks. In *International conference on learning representations*.

Goldberg, D. (1991). What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys (CSUR)*, *23*(1), 5–48.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Huang, G.-B., & Babri, H. A. (1998). Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. *IEEE Transactions on Neural Networks*.

(2019). *IEEE standard for floating-point arithmetic*: *Standard*, IEEE Computer Society.

Jeannerod, C. (2015). Exploiting structure in floating-point arithmetic. In *International conference on mathematical aspects of computer and information sciences* (pp. 25–34).

Jeannerod, C., Louvet, N., Muller, J., & Plet, A. (2016). Sharp error bounds for complex floating-point inversion. *Numerical Algorithms*, *73*(3), 735–760.

Jeannerod, C., & Rump, S. M. (2018). On relative errors of floating-point operations: Optimal bounds and applications. *Mathematics of Computation*, *87*(310), 803–819.

Loosemore, S., Stallman, R. M., McGrath, R., Oram, A., & Drepper, U. (2024). The GNU C library reference manual, for version 2.39. *Free Software Foundation*, URL https://sourceware.org/glibc/manual/2.39/.

Lu, Z., Pu, H., Wang, F., Hu, Z., & Wang, L. (2017). The expressive power of neural networks: A view from the width. In *Annual conference on neural information processing systems*.

Muller, J.-M. (2016). *Elementary functions: Algorithms and implementation* (3rd ed). Springer.

Muller, J.-M., Brunie, N., de Dinechin, F., Jeannerod, C.-P., Joldes, M., Lefevre, V., et al. (2018). *Handbook of floating-point arithmetic*. Springer.

Park, S., Lee, J., Yun, C., & Shin, J. (2021). Provable memorization via deep neural networks using sub-linear parameters. In *Conference on learning theory*.

Park, S., Yun, C., Lee, J., & Shin, J. (2021). Minimum width for universal approximation. In *International conference on learning representations*.

Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. *Acta Numerica*, *8*, 143–195.

Puheim, M., Nyulászi, L., Madarász, L., & Gašpar, V. (2014). On practical constraints of approximation using neural networks on current digital computers. In *IEEE 18th international conference on intelligent engineering systems*.

Sterbenz, P. H. (1973). *Floating-point computation*. Prentice Hall.

Vardi, G., Yehudai, G., & Shamir, O. (2022). On the optimal memorization power of RELU neural networks. In *Conference on learning theory*.

Vershynin, R. (2020). Memory capacity of neural networks with threshold and rectified linear unit activations. *SIAM Journal on Mathematics of Data Science*.

Wray, J., & Green, G. G. (1995). Neural networks, approximation theory, and finite precision computation. *Neural networks*.

Yarotsky, D. (2018). Optimal approximation of continuous functions by very deep ReLU networks. In *Conference on learning theory*.

Yun, C., Sra, S., & Jadbabaie, A. (2019). Small ReLU networks are powerful memorizers: A tight analysis of memorization capacity. In *Annual conference on neural information processing systems (neurIPS)*.