Mathematical Theory of Adversarial Deep Learning

Xiao-Shan Gao [1 2] **Lijia Yu** [1 2] **Shuang Liu** [1 2]

Abstract

In this *Show-and-Tell Demos* paper, progresses on mathematical theories for adversarial deep learning will be reported. Firstly, achieving robust memorization for certain neural networks is shown to be an NP-hard problem. Furthermore, neural networks with $O(Nn)$ parameters are constructed for optimal robust memorization of any dataset with dimension n and size N in polynomial time. Secondly, adversarial training is formulated as a Stackelberg game and is shown to result in a network with optimal adversarial accuracy when the Carlini-Wagner's margin loss is used. Finally, the bias classifier is introduced and is shown to be information-theoretically secure against the original-model gradient-based attack.

1. Introduction

In this paper, partial answers to three basic problems about adversarial deep learning are summarized.

Problem 1. What is the computational complexity for robust memorization with neural networks?

First, to compute robust neural networks with two layers and width 2 is shown to be NP-hard. Second, neural networks are explicitly constructed with $O(Nn)$ parameters for optimal robust memorization of any dataset with dimension n and size N in polynomial time. A lower bound for the width of networks to achieve optimal robust memorization is also given. Finally, neural networks are explicitly constructed with $O(Nn\log n)$ parameters for optimal robust memorization of any binary classification dataset by controlling the Lipschitz constant of the network.

To summarize, finding certain "small" robust networks is NP-hard and optimal robust networks with $O(Nn)$ parameters can be computed in polynomial time. But, $O(Nn)$ is too big to be practical and this leads to the following problem.

Problem 2. For a given hypothesis space of networks, how to obtain an optimal robust network against adversarial attacks?

Game theory has been used to answer Problem 2. In (Pinot et al., 2020; Meunier et al., 2021), adversarial deep learning was treated as a simultaneous game, assuming that the strategy spaces are certain probability distributions to ensure the existence of Nash equilibrium. However, this assumption is not always applicable in practical scenarios. In (Gao et al., 2022), an answer to this problem was given by formulating adversarial deep learning as a Stackelberg game. It was shown that the Stackelberg equilibrium for the game exists and the equilibrium DNN exhibits the highest adversarial accuracy among all DNNs with the same structure, when using Carlini-Wagner's margin loss (Carlini & Wagner, 2017).

To summarize, employing adversarial training with Carlini-Wagner loss yields optimal robust DNNs. However, the trade-off between accuracy and robustness presents a challenge as the robust accuracy achieved by the optimal DNN is still not sufficiently high. This motivates the following problem:

Problem 3. Can provably safe classifiers be built against adversarial attacks?

In (Yu & Gao, 2021), the bias classifier was introduced, that is, the bias part of a DNN with Relu as the activation function is used as a classifier. The existence of the bias classifier is proved and an effective training method for the bias classifier is given based on adversarial training. The bias classifier is shown to have comparable accuracies and robustness with DNNs of similar sizes against major attacks in many cases. Furthermore, the bias classifier can be made provably safe against the original-model gradient attack in the sense that the attack will generate a totally random search direction to generate adversarial examples for any input sample.

2. Related works

Computational complexity. The first NP-hardness result was given in (Blum & Rivest, 1992), which showed that it

[1]Academy of Mathematics and Systems Science, Chinese Academy of Sciences [2]University of Chinese Academy of Sciences. Correspondence to: Xiao-Shan Gao <xgao@mmrc.iss.ac.cn>.

2^{nd} *AdvML Frontiers workshop at* 40^{th} *International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023.

is NP-complete to train certain networks with three nodes. It was proved that even training a single ReLU node is NP-hard (Manurangsi & Reichman, 2018; Dey et al., 2020; Goel et al., 2020) .

Robust networks. Existence of robust networks was proved based on the uninversal approximation theory (Yang et al., 2020; Bastounis et al., 2021). In (Li et al., 2022), a robust interpolation network with for robust budget $\lambda_{\mathcal{D}}/4$ and $O(Nn\log(\frac{n}{\lambda_{\mathcal{D}}})+N\text{polylog}(\frac{N}{\lambda_{\mathcal{D}}}))$ parameters was constructed and it was shown that exponential number of parameters were needed for robust interpolation of infinite sets. To obtain robust networks by controlling the Lipschitz constant was studied in (Bubeck & Sellke, 2021; Zhang et al., 2022).

Certified robust radius. There exist lots of works to give lower bounds for the robust radius or certain security boundaries (Hein & Andriushchenko, 2017; Raghunathan et al., 2018; Shafahi et al., 2019). In (Cohen et al., 2019), random smoothing was proposed and security boundaries of adversaries were given. However, these safety bounds are usually very small when the depth of the DNN is large.

3. Robust memorization with neural networks

3.1. Notations

For $L\in\mathbb{N}_+$, denote $[L]=\{1,\dots,L\}$. For a matrix W and a vector b, denote $W^{j,k}$ to be the element of W at the j-th row and k-th column and $b^{(j)}$ the j-th element of b. For $\mu\in\mathbb{R}_+$ and $x\in\mathbb{R}^n$, denote $\mathbb{B}_\infty(x,\mu)=\{\tilde{x}\in\mathbb{R}^n : ||\tilde{x}-x||_\infty\le\mu\}$.

Consider feedforward neural networks $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}$ with D hidden layers and with $\sigma=\text{Relu}$ as the activation function. The l-th hidden layer of $\mathcal{F}$ can be written as

$$X_l=\sigma(W_lX_{l-1}+b_l)\in\mathbb{R}^{n_l}, l\in[D],$$

and the output is $X_{D+1}=W_{D+1}X_D+b_{D+1}\in\mathbb{R}^{n_{D+1}}$, where $n_0=n$, $X_0\in\mathbb{R}^n$ is the input, $W_l\in\mathbb{R}^{n_l\times n_{l-1}}$, $b_l\in\mathbb{R}^{n_l}$, $n_{D+1}=1$, and $X_{D+1}\in\mathbb{R}$ is the output. $\mathcal{F}$ is said to have depth $\text{depth}(\mathcal{F})=D+1$ and width $\text{width}(\mathcal{F})=\max_{i=1}^{D+1}n_i$. Denote $\mathcal{F}_l(X_0)=X_l$ to be the output of the l-th hidden layer of $\mathcal{F}(X_0)$ and $\mathcal{F}_l^j(X_0)$ the j-th element of $\mathcal{F}_l(X_0)$. The classification result of the network is

$$\widehat{\mathcal{F}}(x)=\text{argmin}_{l\in[L]}|\mathcal{F}(x)-l|.$$

We will explicitly construct networks from certain hypothesis space of networks defined below.

Definition 3.1. Denote the set of networks with depth d, width w, and p parameters by $\mathbf{H}_{n,d,w,p}=\{\mathcal{F}:\mathbb{R}^n\to\mathbb{R}:\text{depth}(\mathcal{F})=d,\text{width}(\mathcal{F})=w,\text{para}(\mathcal{F})=p\}$, where $\text{para}(\mathcal{F})=p$ means that there exists a fixed set $\mathcal{I}\subset\mathbb{N}^4$ with p elements such that $W_l^{i,j}\neq0$ and $b_l^{(s)}\neq0$ for $(l,i,j,s)\in\mathcal{I}$, and all other parameters are zero. We use $*$ to denote an arbitrary number in $\mathbb{N}$. For instance, $\mathbf{H}_{n,d,*,*}=\{\mathcal{F}:\mathbb{R}^n\to\mathbb{R}:\text{depth}(\mathcal{F})=d\}$ is the set of networks with depth d.

Definition 3.2. Let $N,n,L\in\mathbb{N}_+$, and $\mathcal{D}$ be a dataset in $\mathbb{R}^n$ with size N and label set $[L]$; that is $\mathcal{D}=\{(x_i,y_i)\}_{i=1}^N\subset\mathbb{R}^n\times[L]$. **Denote $\mathcal{D}_{n,N,L}$ to be the set of all such dataset.** The *separation bound* for a dataset $\mathcal{D}$ is defined to be

$$\lambda_{\mathcal{D}}=\min\{||x_i-x_j||_\infty:(x_i,y_i),(x_j,y_j)\in\mathcal{D},y_i\neq y_j\}.$$

The *robust accuracy* of a network $\mathcal{F}$ on $\mathcal{D}$ with respect to a given *robust budget* $\mu\in\mathbb{R}_+$ is

$$\text{RA}_{\mathcal{D}}(\mathcal{F},\mu)=\mathbb{P}_{(x,y)\in\mathcal{D}}(\forall\tilde{x}\in\mathbb{B}_\infty(x,\mu),\widehat{\mathcal{F}}(\tilde{x})=y).$$

The problem of *memorization for a dataset* $\mathcal{D}\in\mathcal{D}_{n,N,L}$ is to construct a neural network $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}$, such that $\mathcal{F}(x)=y,\forall(x,y)\in\mathcal{D}$.

Definition 3.3. The problem of **robust memorization for a given dataset $\mathcal{D}\in\mathcal{D}_{n,N,L}$ with budget** μ is to construct a network $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}$ satisfying $\text{RA}_{\mathcal{D}}(\mathcal{F},\mu)=1$. A network hypothesis space $\mathbf{H}$ is said to be an **optimal general robust memorization** for $\mathcal{D}$, if for any $\mu<0.5\lambda_{\mathcal{D}}$, there exists an $\mathcal{F}\in\mathbf{H}$ such that $\mathcal{F}$ is a robust memorization of $\mathcal{D}$ with budget μ.

3.2. Robust memorization is NP-hard

We prove a NP-hard results for computation of robust memorization networks with certain structures.

For $\alpha\in\mathbb{R}_+$ and a binary classification dataset $\mathcal{D}\subset\mathcal{D}_{n,N,2}$, denote $\text{RobM}(\mathcal{D},\alpha)$ to be the decision problem for the existence of an $\mathcal{F}\in\mathbf{H}_{n,2,2,*}$, which is a robust memorization of $\mathcal{D}$ with budget α. We have

Theorem 3.4. $\text{RobM}(\mathcal{D},\alpha)$ *is NP-hard. As a consequence, it is NP-hard to compute an $\mathcal{F}\in\mathbf{H}_{n,2,2,*}$, which is a robust memorization of $\mathcal{D}$ with budget α.*

The proof is given in Appendix A.1.

Remark 3.5. Note that, NP-hardness of computing robust memorization cannot be deduced from NP-hardness of computing memorization. This is because, for a given dataset $\mathcal{D}$ and a network hypothesis space $\mathbf{H}$, it may happen that there exists a memorization network $\mathcal{F}\in\mathbf{H}$ for $\mathcal{D}$, but there exists no robust memorization network $\mathcal{F}\in\mathbf{H}$ for $\mathcal{D}$.

3.3. Construction of optimal robust networks

In this section, we explicitly construct a robust memorization network for a given dataset. We first give a necessary condition for robust memorization.

Proposition 3.6. *If $\mathbf{H}=\{\mathcal{F}:\mathbb{R}^n\to\mathbb{R},\text{width}(\mathcal{F})=w\}$ is an optimal robust memorization of any dataset $\mathcal{D}\in\mathcal{D}_{n,N,L}$ with $N>n$, then* $\text{width}(\mathcal{F})=w\ge n$.

The proof is given in Appendix A.2.

Remark 3.7. In (Vardi et al., 2021), it was shown that width 12 networks are enough for memorization. Proposition 3.6 indicates that robust memorization needs essentially larger width.

The following theorem gives an optimal robust memorization for any given dataset.

Theorem 3.8. *For any dataset $\mathcal{D} \in \mathcal{D}_{n,N,L}$, the hypothesis space $\mathbf{H}_{n,2N+1,3n+1,O(Nn)}$ is an optimal robust memorization for $\mathcal{D}$. Furthermore, the optimal robust network can be explicitly constructed in polynomial-time.*

The proof is given in Appendix A.3.

3.4. Optimal robust memorization via Lipschitz

Controlling the Lipschitz constant is widely used to achieve robustness (Bubeck & Sellke, 2021; Zhang et al., 2022). In this section, we give a robust network based on Lipschitz constraint.

We consider a binary classification dataset $\mathcal{D} \in \mathcal{D}_{n,N,2}$. There exist $(x_i, 1), (x_j, 2) \in \mathcal{D}$ such that $||x_i - x_j||_\infty = \lambda_{\mathcal{D}}$. Thus, if $\mathcal{F}$ memorizes $\mathcal{D}$, then $\text{Lip}_\infty(\mathcal{F}) \geq |\mathcal{F}(x_i) - \mathcal{F}(x_j)|/||x_i - x_j||_\infty = 1/\lambda_{\mathcal{D}}$, which motivates the following definition.

Definition 3.9. A network $\mathcal{F}$ is called a *robust memorization of a dataset $\mathcal{D}$ via Lipschitz with budget $\mu < 0.5\lambda_{\mathcal{D}}$*, if $\mathcal{F}$ is a memorization of $\mathcal{D}$ and $\text{Lip}_\infty(\mathcal{F}) \leq 0.5/\mu$. $\mathcal{F}$ is called an *optimal robust memorization of $\mathcal{D}$ via Lipschitz*, if $\mathcal{F}$ is a memorization of $\mathcal{D}$ and $\text{Lip}_\infty(\mathcal{F}) = 1/\lambda_{\mathcal{D}}$.

We now construct an optimal robust network for any binary classification dataset via Lipschitz.

Theorem 3.10. *For any dataset $\mathcal{D} \in \mathcal{D}_{n,N,2}$, the hypothesis space $\mathbf{H}_{n,O(N\log(n)),O(n),O(Nn\log(n))}$ contains a network $\mathcal{F}$ which is an optimal robust memorization of $\mathcal{D}$ via Lipschitz. Furthermore, the network can be explicitly constructed in polynomial-time.*

Proof is in appendix A.4.

4. Achieve optimal robustness with Stackelberg game

In this section, we consider Problem 2, that is, for a given hypothesis space of networks, how to obtain an optimal robust network against adversarial attacks? Proofs for theorems in this section can be found in (Gao et al., 2022).

4.1. Adversarial training and robustness of DNN

Let $\mathcal{C} : \mathbb{I}^n \to \mathbb{R}^m$ be a classification DNN with m labels in $\mathcal{Y} = [m] = \{1, \ldots, m\}$ and $\mathbb{I} = [0, 1]$. For $x \in \mathbb{I}^n$, the classification result of $\mathcal{C}$ is $\widehat{\mathcal{C}}(x) = \text{argmax}_{l\in\mathcal{Y}}\, \mathcal{C}_l(x)$.

Let the parameter set of $\mathcal{C}$ be $\Theta \in \mathbb{R}^K$. Then $\mathcal{C}$ can be written as $\mathcal{C}_\Theta$. We assume the data to be classified satisfy a distribution $\mathcal{D}$ over $\mathbb{I}^n \times \mathcal{Y}$. Given a loss function Loss : $\mathbb{R}^m \times \mathcal{Y} \to \mathbb{R}$, the expected loss for the dataset is

$$\varphi_0(\Theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \text{Loss}(\mathcal{C}_\Theta(x), y). \tag{1}$$

Training $\mathcal{C}_\Theta$ is to solve the following optimization problem:

$$\text{argmin}_{\Theta\in\mathbb{R}^K}\, \varphi_0(\Theta). \tag{2}$$

In order to increase the robustness of a trained DNN, the *adversarial training* (Madry et al., 2017) was introduced by solving the following robust optimization problem:

$$\text{argmin}_{\Theta\in\mathbb{R}^K}\, \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \max_{\tilde{x}\in\mathbb{B}(x,\varepsilon)} \text{Loss}(\mathcal{C}_\Theta(\tilde{x}), y). \tag{3}$$

Given a DNN $\mathcal{C}$ and an attack radius ε, we define the *adversarial robustness measure* of $\mathcal{C}$ with respect to ε as

$$\text{AR}_{\mathcal{D}}(\mathcal{C}, \varepsilon) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \max_{\tilde{x}\in\mathbb{B}(x,\varepsilon)} \text{Loss}(\mathcal{C}(\tilde{x}), y), \tag{4}$$

which is the expected loss of $\mathcal{C}$ at the most-adversarial samples.

4.2. Adversarial training as a Stackelberg game

We formulate adversarial deep learning as a two-player zero-sum Stackelberg game $\mathcal{G}_s$, called *adversarial learning game*.

The leader of the game $\mathcal{G}_s$ is the Classifier, whose goal is to train a robust DNN $\mathcal{C}_\Theta : \mathbb{I}^n \to \mathbb{R}^m$, where the parameters Θ are in

$$\mathcal{S}_c = [-E, E]^K \text{ for } E \in \mathbb{R}_+. \tag{5}$$

In other words, the strategy space for the Classifier is $\mathcal{S}_c$.

The follower of the game $\mathcal{G}_s$ is the Adversary, whose goal is to create the best adversary within a given attack radius $\varepsilon \in \mathbb{R}_+$. The strategy space for the Adversary is

$$\mathcal{S}_a = \{A : \mathbb{I}^n \to \mathbb{B}_\varepsilon\}, \tag{6}$$

where $\mathbb{B}_\varepsilon = \{\delta \in \mathbb{R}^n : ||\delta|| \leq \varepsilon\}$.

The payoff function. Given $\Theta \in \mathcal{S}_c$ and $A \in \mathcal{S}_a$, the payoff function is the expected adversarial loss

$$\varphi_s(\Theta, A) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \text{Loss}(\mathcal{C}_\Theta(x + A(x)), y). \tag{7}$$

For game $\mathcal{G}_s$, the *best response set* of the adversary is

$$\gamma_s(\Theta) = \{\text{argmax}_{A\in\mathcal{S}_a} \varphi_s(\Theta, A)\} \text{ for } \Theta \in \mathcal{S}_c \tag{8}$$

And (Θ_s^*, A_s^*) is called a *Stackelberg equilibrium* of $\mathcal{G}_s$ if

$$\begin{aligned} &\Theta_s^* \in \text{argmin}_{\Theta\in\mathcal{S}_c, A(\Theta)\in\gamma_s(\Theta)}\, \varphi_s(\Theta, A(\Theta)) \\ &A_s^* \in \text{argmax}_{A\in\mathcal{S}_a}\, \varphi_s(\Theta_s^*, A). \end{aligned} \tag{9}$$

We now have:

Theorem 4.1. *Game $\mathcal{G}_s$ has a Stackelberg equilibrium (Θ_s^*, A_s^*), meanwhile the equilibrium DNN with parameters Θ_s^* minimizes the adversarial robustness measure in (4). Furthermore, Θ_s^* is the solution to the adversarial training in (3).*

4.3. Achieve maximal robust accuracy

Comparing with the robustness measurement $\mathrm{AR}_{\mathcal{D}}$ in (4), the robust accuracy

$$\mathrm{RA}_{\mathcal{D}}(\mathcal{C}, \epsilon) := \mathbb{P}_{(x,y)\sim\mathcal{D}}\left(\forall \tilde{x} \in \mathbb{B}(x, \varepsilon)\left(\widehat{\mathcal{C}}(\tilde{x}) = y\right)\right)$$

does not depend on the choice of loss functions, indicating its intrinsic nature.

Denote the game $\mathcal{G}_s$ as $\mathcal{G}_{cw}$, when the payoff function is

$$\varphi_{cw}(\Theta, A) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \mathrm{Loss}_{\mathrm{cw}}(\mathcal{C}_\Theta(x + A(x)), y) \quad (10)$$

and $\mathrm{Loss}_{\mathrm{cw}}(z, y) = \max_{l\in[m], l\neq y} z_l - z_y$ is the Carlini-Wagner loss. Then we have

Theorem 4.2. *Let $(\Theta_{\mathrm{cw}}^*, A_{\mathrm{cw}}^*)$ be a Stackelberg equilibrium of game $\mathcal{G}_{cw}$. Then $\mathcal{C}_{\Theta_{\mathrm{cw}}^*}$ has the largest adversarial accuracy for all DNNs whose parameters are in $\mathcal{S}_c$; that is, $\mathrm{RA}_{\mathcal{D}}(\mathcal{C}_{\Theta_{\mathrm{cw}}^*}, \varepsilon) \geq \mathrm{RA}_{\mathcal{D}}(\mathcal{C}_\Theta, \varepsilon)$ for any $\Theta \in \mathcal{S}_c$.*

4.4. Trade-off between robustness and accuracy

Theorem 4.2 allows us to formulate the trade-off problem as the following bi-level optimization problem, that is, to increase the accuracy of the DNN and still keep the maximal robust accuracy.

$$\begin{aligned} \Theta_o^* \quad = \quad & \mathrm{argmin}_{\Theta_s^*} \varphi_0(\Theta_s^*) \\ & \text{subject to} \\ & \Theta_s^* = \mathrm{argmin}_{\Theta\in\mathcal{S}_c} \max_{A\in\mathcal{S}_a} \varphi_{cw}(\Theta, A), \end{aligned} \quad (11)$$

where φ_0 and φ_{cw} are defined in (1) and (10), respectively.

The bi-level optimization problem (11) is, in general, difficult to solve. A natural way to train a robust and more accurate DNN is to do adversarial training with the following objective function:

$$\varphi_t(\Theta, A) = \varphi_s(\Theta, A) + \lambda\varphi_0(\Theta). \quad (12)$$

Then we have the following trade-off result:

Proposition 4.3. *Let (Θ_s^*, A_s^*) and (Θ_t^*, A_t^*) be the Stackelberg equilibria of the adversarial learning games with φ_s and φ_t as the payoff functions respectively. Then the network $\mathcal{C}_{\Theta_s^*}$ is more robust but less accurate than $\mathcal{C}_{\Theta_t^*}$ measured by φ_0.*

5. Information-theoretically safe bias classifier against adversarial attacks

In this section, we introduce the bias classifier and show that it can be made information-theoretically safe against the original-model gradient-based attack. Proofs for theorems in this Section can be found in (Yu & Gao, 2021).

5.1. Existence and training of bias classifier

Let $\mathcal{F} : \mathbb{I}^n \to \mathbb{R}^m$ be a classification DNN. For $x \in \mathbb{I}^n$, let $\mathcal{J}_{\mathcal{F}}(x) = W_x = \frac{\nabla\mathcal{F}(t)}{\nabla t}|_x$ be the Jacobian of $\mathcal{F}$ at x. Then

$$\mathcal{F}(x) = W_{\mathcal{F}}(x) + \mathcal{B}_{\mathcal{F}}(x) = \mathcal{J}_{\mathcal{F}}(x)x + B_x. \quad (13)$$

The bias part $\mathcal{B}_{\mathcal{F}} : \mathbb{I}^n \to \mathbb{R}^m$ will be used as a classifier and is called the *bias classifier*, which can be computed from $\mathcal{F}$ as follows:

$$\mathcal{B}_{\mathcal{F}}(x) = \mathcal{F}(x) - W_{\mathcal{F}}(x) = \mathcal{F}(x) - \mathcal{J}_{\mathcal{F}}(x) \cdot x. \quad (14)$$

Due to the property of the Relu function, $\mathcal{F}$ is a piecewise linear function and $\mathcal{B}_{\mathcal{F}}(x)$ is a piecewise constant function.

Existence of bias classifier We will prove that for any decision function $\mathcal{G} : \mathbb{I}^n \to [m]$, there exists a bias classifier $\mathcal{B}(x) : \mathbb{I}^n \to \mathbb{R}^m$, such that $\widehat{\mathcal{B}}(x)$ can be arbitrarily close to $\mathcal{G}$, where $\widehat{\mathcal{B}}(x)$ represents the classification results of $\mathcal{B}(x)$. The decision function is defined as below:

Definition 5.1. $\mathcal{G} : \mathbb{I}^n \to [m]$ is a decision function, if $P_1, \ldots, P_m$ are a partition of $\mathbb{I}^n$ into m measurable disjoint subsets, and $\mathcal{G}(x) = i$ if and only if $x \in P_i$.

We have the existence theorem.

Theorem 5.2. *Let $\mathcal{G} : \mathbb{I}^n \to [m]$ be a decision function. Then for $\epsilon \in \mathbb{R}_+$, there exist a bias classifier $\mathcal{B} : \mathbb{I}^n \to \mathbb{R}^m$ and an open set $D \subset \mathbb{I}^n$ with volume $V(D) < \epsilon$, such that $\widehat{\mathcal{B}}(x) = G(x)$ for $x \in \mathbb{I} \setminus D$.*

Training the bias classifier In order to increase the power of the bias part $\mathcal{B}_{\mathcal{F}}$ and to keep the training procedure efficient to update the parameters, we train the bias classifier by solving the following optimization problem

$$\min_{\Theta} \sum_{(x,y)\sim\mathcal{D}} [\max_{||\zeta||<\varepsilon} L_{\mathrm{ce}}(\mathcal{B}_{\mathcal{F}}(x+\zeta), y) + \gamma \max_{||\zeta||<\varepsilon} L_{\mathrm{ce}}(\mathcal{F}(x+\zeta), y)] \quad (15)$$

where $\gamma \in \mathbb{R}_+$ is a hyperparameter. Notice that the training (15) is a combination of adversarial training for $\mathcal{B}$ and $\mathcal{F}$.

A simple numerical experiment is used to show that the adversarial training and the training in (15) can increase the classification power of $\mathcal{B}_{\mathcal{F}}$.

Table 1. Accuracies of network Lenet-5 for MNIST on the test set. Standard training (2) does not give classification power to $\mathcal{B}_{\mathcal{F}}$. Adversarial training (3) increases the power of the bias part. Adversarial training for bias classifier (15) further decreases the power of the first-degree part to avoid attacks based on it.

Training	$W_{\mathcal{F}}$	$\mathcal{B}_{\mathcal{F}}$	$\mathcal{F}$
(2)	98.80%	15.62%	99.09%
(3)	90.61%	98.77%	99.19%
(15)	0.28%	99.09 %	99.43%

5.2. Information-theoretically safe (ITS) bias classifier

Original-model gradient based attack The most popular methods to generate adversaries, such as FGSM (Goodfellow et al., 2014) or PGD (Madry et al., 2017), use $\frac{\nabla \mathcal{F}(x)}{\nabla x}$ to make the loss function bigger. More precisely, adversaries are generated as follows:

$$x \to x + \varepsilon \,\mathrm{Sgn}\Big(\frac{\nabla L_{\mathrm{ce}}(\mathcal{F}(x), y)}{\nabla x}\Big) \tag{16}$$

for a small parameter $\varepsilon \in \mathbb{R}_+$. It is easy to see that, $\frac{\nabla L_{\mathrm{ce}}(\mathcal{F}(x),y)}{\nabla x}$ can be obtained from $\frac{\nabla \mathcal{F}(x)}{\nabla x}$. In the above attack, only the values of $\mathcal{F}(x)$ and $\frac{\nabla \mathcal{F}(x)}{\nabla x}$ are needed and the detailed structure of $\mathcal{F}$ is not needed. Motivated by this fact, we introduce the concept of gradient-based attack.

Definition 5.3. A *gradient-based attack* for a DNN $\mathcal{F} : \mathbb{I}^n \to \mathbb{R}^m$ is a map $\mathcal{A}_{\mathcal{F},\rho} : \mathbb{R}^n \to \mathbb{R}^n$ defined as

$$\mathcal{A}_{\mathcal{F},\rho}(x) = x + \rho\, D_{\mathcal{F}}(x) \tag{17}$$

where $D_{\mathcal{F}}(x) \in \{-1, 1\}^n$ is the *attack direction* for x and $D_{\mathcal{F}}(x)$ only depends on $\mathcal{F}(x)$ and $\frac{\nabla \mathcal{F}(x)}{\nabla x}$.

Since the gradient of $\mathcal{B}_{\mathcal{F}}$ is zero, we cannot use the gradient based attack for $\mathcal{B}_{\mathcal{F}}$. An obvious attack to the bias classifier is to create adversaries of $\mathcal{B}_{\mathcal{F}}$ using the gradients of $\mathcal{F}$, which is called *original-model gradient based attack*.

Information-theoretically safety First train a DNN $\mathcal{F} : \mathbb{I}^n \to \mathbb{R}^m$ with the method in Section 5.1. Let $W_R \in \mathbb{R}^{m\times n}$ satisfy a given distribution $\mathcal{M}$ of random matrices in $\mathbb{R}^{m\times n}$ and let

$$\begin{aligned} \widetilde{\mathcal{F}}(x) &= \mathcal{F}(x) + W_R x = (W_x + W_R)x + B_x \\ \mathcal{B}_{\widetilde{\mathcal{F}}}(x) &= \widetilde{\mathcal{F}}(x) - \tfrac{\nabla \widetilde{\mathcal{F}}(x)}{\nabla x} \cdot x = \widetilde{\mathcal{F}}(x) - \mathcal{J}_{\widetilde{\mathcal{F}}} \cdot x. \end{aligned} \tag{18}$$

It is easy to see that $\mathcal{B}_{\widetilde{\mathcal{F}}} = \mathcal{B}_{\mathcal{F}}$, that is, the bias classifiers for $\mathcal{F}$ and $\widetilde{\mathcal{F}}$ are the same. On the other hand, $\mathcal{J}_{\widetilde{\mathcal{F}}} = \frac{\nabla \widetilde{\mathcal{F}}(x)}{\nabla x} = \frac{\nabla \mathcal{F}(x)}{\nabla x} + W_R$ is random as shown below.

The safety of $\mathcal{B}_{\widetilde{\mathcal{F}}}$ against the attack $\mathcal{A}_{\widetilde{\mathcal{F}},\rho}(x)$ can be measured by the following *adversary creation rate*:

$$\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}, \mathcal{M}, \rho) = \mathbb{E}_{W_R\sim\mathcal{M}}[\mathbb{E}_{x\sim\mathcal{D}}[\mathbf{I}(\widehat{\mathcal{B}}_{\widetilde{\mathcal{F}}}(\mathcal{A}_{\widetilde{\mathcal{F}},\rho}(x)) \neq \widehat{\mathcal{B}}_{\widetilde{\mathcal{F}}}(x))]]. \tag{19}$$

Definition 5.4. The bias classifier $\mathcal{B}_{\widetilde{\mathcal{F}}}$ in (18) is called *information-theoretically safe* (ITS) against the gradient-based attack $\mathcal{A}_{\widetilde{\mathcal{F}},\rho}$ defined in (17), if the attack direction $D_{\widetilde{\mathcal{F}}}(x) = (\mathcal{A}_{\widetilde{\mathcal{F}},\rho}(x) - x)/\rho$ is a random vector in $\{-1, 1\}^n$ for any input $x \in \mathbb{I}^n$.

If $\mathcal{B}_{\widetilde{\mathcal{F}}}$ is ITS against $\mathcal{A}_{\widetilde{\mathcal{F}},\rho}$, then $\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}, \mathcal{M}, \rho)$ equals

$$\mathcal{C}(\mathcal{F}, \rho) = \tfrac{1}{2^n}\mathcal{E}_{x\sim\mathcal{D}} \textstyle\sum_{V\in\{-1,1\}^n}[\mathbf{I}(\widehat{\mathcal{B}}_{\mathcal{F}}(x + \rho V) \neq \widehat{\mathcal{B}}_{\mathcal{F}}(x))] \tag{20}$$

which depends only on $\mathcal{F}$ and ρ and will be used as a robustness measurement of the bias classifier.

Remark 5.5. The notion of *information-theoretically safe*, also called perfectly safe, is borrowed from cryptography (Goldreich, 2004)[p.476], which means that the ciphertext yields no information regarding the plaintext for cyphers which are perfectly random.

5.3. ITS against signed margin attack

We first show that $\mathcal{B}_{\widetilde{\mathcal{F}}}$ defined in (18) is safe against the following *signed margin attack* (Carlini & Wagner, 2017)

$$\mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}(x) = x + \rho\,\mathrm{Sgn}\Big(\frac{\nabla \widetilde{\mathcal{F}}_{n_x}(x)}{\nabla x} - \frac{\nabla \widetilde{\mathcal{F}}_y(x)}{\nabla x}\Big) \tag{21}$$

where $\rho \in \mathbb{R}_+$, y is the label of x, and $n_x = \arg\max_{i\neq y}\{\mathcal{F}_i(x)\}$.

We consider two types of random matrices for W_R in (18).

Definition 5.6. Let $\mathcal{U}(a, b)$ be the uniform distribution in $[a, b] \subset \mathbb{R}$. For $\lambda \in \mathbb{R}_+$, denote $\mathcal{U}_{m,n}(\lambda)$ to be the random matrices whose entries are in $\mathcal{U}(-\lambda, \lambda)$ and denote $\mathcal{M}_{m,n}(\lambda)$ to be the random matrices such that the entries of their i-row are in $(\mathcal{U}(-2i\lambda, -(2i-1)\lambda) \cup \mathcal{U}((2i-1)\lambda, 2i\lambda))^{m\times n}$.

Theorem 5.7. *Let $||J_{\mathcal{F}}||_\infty < \lambda/2$ and $W_R \in \mathcal{M}_{m,n}(\lambda)$ for $\lambda \in \mathbb{R}_+$. Then $\mathcal{B}_{\widetilde{\mathcal{F}}}$ is ITS against the attack $\mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}$ given in (21); that is, attacking $\mathcal{B}_{\widetilde{\mathcal{F}}}$ using $\mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}$ will generate a random attack direction $(\mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}(x) - x)/\rho$ for any $x \in \mathbb{I}^n$.*

Theorem 5.8. *If $||\mathcal{J}_{\mathcal{F}}||_\infty < \mu/2$ and $W_R \sim \mathcal{U}_{m,n}(\lambda)$, then $\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}, \mathcal{U}_{m,n}(\lambda), \rho) \le \mathcal{C}(\mathcal{F}, \rho) + \mu n/\lambda$. Furthermore, if $\lambda > \mu n/(\epsilon\mathcal{C}(\mathcal{F}, \rho))$, then $\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}, \mathcal{U}_{m,n}(\lambda), \rho) \le (1 + \epsilon)\mathcal{C}(\mathcal{F}, \rho)$.*

For the simpler distribution $\mathcal{U}_{m,n}$, Theorem 5.8 implies that $\mathcal{B}_{\widetilde{\mathcal{F}}}$ is close to ITS under attack $\mathcal{A}_{\widetilde{\mathcal{F}},\rho,1}$ if λ is sufficiently large.

5.4. ITS against FGSM attack

In this section, we show that $\mathcal{B}_{\widetilde{\mathcal{F}}}$ in (18) is safe against the FGSM attack (Goodfellow et al., 2014) for binary classification problems. Here is the FGSM attack:

$$\mathcal{A}_{\widetilde{\mathcal{F}},\rho,2}(x) = x + \rho\,\mathrm{Sgn}\Big(\frac{\nabla L(\widetilde{\mathcal{F}}(x), y)}{\nabla x}\Big). \tag{22}$$

Theorem 5.9. *For $\lambda \in \mathbb{R}_+$, if $||J_{\mathcal{F}}||_\infty < \lambda/2$, $W_R \sim \mathcal{M}_{m,n}(\lambda)$, and $m = 2$, then $\mathcal{B}_{\widetilde{\mathcal{F}}}$ is ITS against the attack $\mathcal{A}_{\widetilde{\mathcal{F}},\rho,2}$.*

Theorem 5.10. *If $||\mathcal{J}_{\mathcal{F}}||_\infty < \mu/2$, $W_R \sim \mathcal{U}_{m,n}(\lambda)$, and $m = 2$, then $\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}_{\widetilde{\mathcal{F}},\rho,2}, \mathcal{U}_{m,n}(\lambda), \rho) \le e^{n\mu/\lambda}\mathcal{C}(\mathcal{F}, \rho)$. Furthermore, if $\lambda > n\mu/\ln(1 + \epsilon)$, then $\mathcal{C}(\mathcal{B}_{\widetilde{\mathcal{F}}}, \mathcal{A}_{\widetilde{\mathcal{F}},\rho,2}, \mathcal{U}_{m,n}(\lambda), \rho) \le (1 + \epsilon)\mathcal{C}(\mathcal{F}, \rho)$.*

Theorem 5.10 shows that $\mathcal{B}_{\widetilde{\mathcal{F}}}$ is close to ITS against FGSM under distribution $\mathcal{U}_{m,n}(\lambda)$ for a sufficiently large λ.

6. Conclusion and problems for further studies

In this paper, we provide a summary of partial solutions to three fundamental challenges in adversarial deep learning: computationally complexity for robust memorization, the search for the optimal robust network, and building provably safe classifiers. While these methods are currently in their early stages and may not yet attain state-of-the-art performance for practical applications, further advancements in addressing these problems will pave the way for more reliable techniques in adversarial deep learning. Consequently, we propose the following research questions for future investigation.

On the computationally complexity for robust memorization, can we construct robust networks that achieve the lower bound of parameters $O(\sqrt{Nn})$ or have generalization power? On the training of optimal robust networks, can we train a robust network with optimal generalization power? On the provably safe classifiers, can we build a provably safe classifier with SOTA performance?

References

Bastounis, A., Hansen, A. C., and Vlačić, V. The mathematics of adversarial attacks in ai–why deep learning is unstable despite the existence of stable neural networks. *arXiv preprint arXiv:2109.06098*, 2021.

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

Bubeck, S. and Sellke, M. A universal law of robustness via isoperimetry. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 28811–28822. Curran Associates, Inc., 2021.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, 2017. doi: 10.1109/SP.2017.49.

Cohen, J., Rosenfeld, E., and Kolter, Z. Certified adversarial robustness via randomized smoothing. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 09–15 Jun 2019.

Dey, S. S., Wang, G., and Xie, Y. Approximation algorithms for training one-node relu neural networks. *IEEE Transactions on Signal Processing*, 68:6696–6706, 2020.

Gao, X.-S., Liu, S., and Yu, L. Achieving optimal adversarial accuracy for adversarial deep learning using stackelberg games. *arXiv preprint arXiv:2207.08137*, 2022.

Goel, S., Klivans, A., Manurangsi, P., and Reichman, D. Tight hardness results for training depth-2 relu networks. *arXiv preprint arXiv:2011.13550*, 2020.

Goldreich, O. *Foundations of Cryptography, Volume 2*. Cambridge university press Cambridge, 2004.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Hein, M. and Andriushchenko, M. Formal guarantees on the robustness of a classifier against adversarial manipulation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Li, B., Jin, J., Zhong, H., Hopcroft, J. E., and Wang, L. Why robust generalization in deep learning is difficult: Perspective of expressive power. *arXiv preprint arXiv:2205.13863*, 2022.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2017.

Manurangsi, P. and Reichman, D. The computational complexity of training relu (s). *arXiv preprint arXiv:1810.04207*, 2018.

Megiddo, N. On the complexity of polyhedral separability. *Discrete & Computational Geometry*, 3:325–337, 1988.

Meunier, L., Scetbon, M., Pinot, R., Atif, J., and Chevaleyre, Y. Mixed nash equilibria in the adversarial examples game. In *ICML*, 2021.

Pinot, R., Ettedgui, R., Rizk, G., Chevaleyre, Y., and Atif, J. Randomization matters. how to defend against strong adversarial attacks. In *ICML*, 2020.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. In *International Conference on Learning Representations*, 2018.

Shafahi, A., Huang, W. R., Studer, C., Feizi, S., and Goldstein, T. Are adversarial examples inevitable? In *International Conference on Learning Representations*, 2019.

Vardi, G., Yehudai, G., and Shamir, O. On the optimal memorization power of relu neural networks. *arXiv preprint arXiv:2110.03187*, 2021.

Yang, Y.-Y., Rashtchian, C., Zhang, H., Salakhutdinov, R. R., and Chaudhuri, K. A closer look at accuracy vs. robustness. In Larochelle, H., Ranzato, M., Hadsell, R.,

Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 8588–8601. Curran Associates, Inc., 2020.

Yu, L. and Gao, X.-S. Robust and information-theoretically safe bias classifier against adversarial attacks. *arXiv preprint arXiv:2111.04404*, 2021.

Zhang, H., Wu, Y., and Huang, H. How many data are needed for robust learning? *arXiv preprint arXiv:2202.11592*, 2022.

A. Appendix to Paper: Mathematical Theory of Adversarial Deep Learning

This appendix contains proofs for theorems in Section 3.

A.1. Proof of Theorem 3.4

We will show that RobM$(\mathcal{D}, \alpha)$ is computationally equivalent to the following NPC problem.

Definition A.1 (Reversible 6-SAT)**.** Let φ be a Boolean formula and let $\overline{\varphi}$ denote the formula obtained from φ by negating each variable. The Boolean formula φ is called *reversible* if either both φ and $\overline{\varphi}$ are satisfiable or both are not satisfiable. The *reversible satisfiability problem* is to recognize the satisfiability of reversible formulae in conjunctive normal form (CNF). By the *reversible 6-SAT*, we mean the reversible satisfiability problem for CNF formulae with six variables per clause. In (Megiddo, 1988), it was shown that the reversible 6-SAT is NPC.

Denote $\mathcal{H}_{n,2} = \mathbf{H}_{n,2,2,*}$. Let $\mathcal{F}(x) \in \mathcal{H}_{n,2}$. Then $\mathcal{F}(x)$ can be written as

$$\mathcal{F} = s_1\sigma(U_1x + b_1) + s_2\sigma(U_2x + b_2) + c \tag{23}$$

where $U_i \in \mathbb{R}^{1\times n}$, $b_i \in \mathbb{R}$, $s_i \in \{-1, 1\}$, $c \in \mathbb{R}$. To simplify the proof, we assume that $\widetilde{\mathcal{F}}(x) = \psi(\mathcal{F}(x))$, where ψ is the sign function. Also, let $\mathbf{1}_i \in \mathbb{R}^k$, whose i-th element is 1 and all other entries are 0.

We first prove a lemma.

Lemma A.2. *Let $\mathcal{F} \in \mathcal{H}_{n,2}$ and $z_i = i\mathbf{1}_1 \in \mathbb{R}^n$. If $\widetilde{\mathcal{F}}(z_1) = \widetilde{\mathcal{F}}(z_{-1}) = -1$ and $\widetilde{\mathcal{F}}(z_2) = \widetilde{\mathcal{F}}(z_{-2}) = 1$, then $s_1s_2 < 0$ when $c \geq 0$ and $s_1s_2 > 0$ when $c < 0$.*

Proof. Let $\mathcal{F}$ be of form (23). We just need to prove the lemma for $n = 1$. Consider two cases.

(c1): Assume $c \geq 0$. Since $-1 = \widetilde{\mathcal{F}}(z_1) = \psi(s_1\sigma(U_1z_1+b_1)+s_2\sigma(U_2z_1+b_2)+c) \geq \psi(s_1\sigma(U_1z_1+b_1)+s_2\sigma(U_2z_1+b_2))$, at least one of $s_1 = -1$ and $s_2 = -1$ holds. Assume that $s_2 = -1$. We will show that $s_1 = 1$. If this is not true, then $s_1 = s_2 = -1$. Because $-\sigma(a)/4 - 3/4\sigma(b) \leq -\sigma(a/4 + 3b/4)$, we have that

$$\begin{aligned}
& 0 \\
< \;& (-\sigma(U_1z_2 + b_1) - \sigma(U_2z_2 + b_2) + c)/4 + 3(-\sigma(U_1z_{-2} + b_1) - \sigma(U_2z_{-2} + b_2) + c)/4 \\
= \;& (-\sigma(U_1z_2 + b_1) - 3\sigma(U_1z_{-2} + b_1))/4 + (-\sigma(U_2z_2 + b_2) - 3\sigma(U_2z_{-2} + b_2))/4 + c \\
\leq \;& -\sigma(U_1z_{-1} + b_1) - \sigma(U_2z_{-1} + b_2) + c \\
< \;& 0
\end{aligned}$$

which is a contradiction.

(c2): Assume $c < 0$. Since $1 = \widetilde{\mathcal{F}}(z_2) = \psi(s_1\sigma(U_1z_2+b_1)+s_2\sigma(U_2z_2+b_2)+c) \leq \psi(s_1\sigma(U_1z_2+b_1)+s_2\sigma(U_2z_2+b_2))$, at least one of $s_1 = 1$ and $s_2 = 1$ holds. Assume $s_1 = 1$. We will show that $s_2 = 1$. If this is not true, then $s_2 = -1$. Then

$$1 = \mathcal{F}(z_2) = \psi(\sigma(U_1z_2 + b_1) - \sigma(U_2z_2 + b_2) + c) \leq \psi(\sigma(U_1z_2 + b_1) + c),$$

so $\sigma(U_1z_2 + b_1) + c > 0$. Similarly, we have $\sigma(U_1z_{-2} + b_1) + c > 0$. It is easy to know that $U_1z_1 + b_1 \geq \min\{U_1z_2 + b_1, U_1z_{-2} + b_1\}$, so $\sigma(U_1z_1 + b_1) + c > 0$. Similarly, we have $\sigma(U_1z_{-1} + b_1) + c > 0$. From $\mathcal{F}(z_1) = -1$ and $\sigma(U_1z_1 + b_1) + c > 0$, we have

$$0 > \sigma(U_1z_1 + b_1) - \sigma(U_2z_1 + b_2) + c > -\sigma(U_2z_1 + b_2).$$

So $0 < \sigma(U_2z_1 + b_2)$, which means $U_2z_1 + b_2 > 0$. Similarly, we have $U_2z_{-1} + b_2 > 0$. Now consider the linear function $L(x) = (U_1z_2 + b_1) - (U_2z_2 + b_2) + c$.

Since $c < 0$, $\sigma(U_1z_1 + b_1) + c > 0$, $U_2z_1 + b_2 > 0$, and $\mathcal{F}(z_1) = -1$, we have $L(z_1) = (U_1z_1 + b_1) - (U_2z_1 + b_2) + c = \sigma(U_1z_1 + b_1) - \sigma(U_2z_1 + b_2) + c < 0$. Similarly, $L(z_{-1}) < 0$.

Since $c < 0$, $\sigma(U_1z_2 + b_1) + c > 0$, and $\mathcal{F}(z_2) = 1$, we have $L(z_2) = (U_1z_2 + b_1) - (U_2z_2 + b_2) + c = \sigma(U_1z_2 + b_1) - (U_2z_2 + b_2) + c \geq \sigma(U_1z_2 + b_1) - \sigma(U_2z_2 + b_2) + c > 0$. Similarly, we have $L(z_{-2}) > 0$.

Hence $L(0) = (L(z_1) + L(z_{-1}))/2 < 0$ and $L(0) = (L(z_2) + L(z_{-2}))/2 > 0$, a contradiction, so $s_2 = 1$. □

We restate Theorem 3.4 here for convenience.

Theorem A.3. RobM$(\mathcal{D}, \alpha)$ *is NP-hard; that is, for* $\alpha \in \mathbb{R}_+$ *and a dataset* $\mathcal{D} \subset \mathcal{D}_{n,N,2}$*, it is NP-hard to decide whether there exists a robust network in* $\mathcal{H}_{n,2}$ *for* $\mathcal{D}$ *with budget* α*.*

Proof. Let $\varphi(k,m) = \wedge_{i=1}^m \varphi_i(k,m)$ be a 6-SAT for k variables, where $\varphi_i(k,m) = \vee_{j=1}^6 \tilde{x}_{i,j}$ and $\tilde{x}_{i,j}$ is either x_s or $\neg x_s$ for $s \in [k]$ (refer to Definition A.1).

For $i \in [k]$, define $Q_i^\varphi \in \mathbb{R}^k$ as follows: $Q_i^\varphi[j] = 1$ if x_j occurs in $\varphi_i(k,m)$; $Q_i^\varphi[j] = -1$ if $\neg x_j$ occurs in $\varphi_i(k,m)$; $Q_i^\varphi[j] = 0$ otherwise. Then six entries of Q_i^φ are 1 or -1 and all other entries are zero.

We define a binary classification dataset $\mathcal{D}(\varphi) = \{(x_i, y_i)\}_{i=1}^{m+4k} \subset \mathbb{R}^k \times \{-1, 1\}$ as follows

(1) For $i \in [k]$, $x_i = k\mathbf{1}_i$, $y_i = 1$.

(2) For $i \in \{k+1, k+2, \ldots, 2k\}$, $x_i = -k\mathbf{1}_{i-k}$, $y_i = 1$.

(3) For $i \in \{2k+1, 2k+2, \ldots, 3k\}$, $x_i = 2k\mathbf{1}_{i-2k}$, $y_i = 2$.

(4) For $i \in \{3k+1, 3k+2, \ldots, 4k\}$, $x_i = -2k\mathbf{1}_{i-3k}$, $y_i = 2$.

(5) For $i \in \{4k+1, 3k+2, \ldots, 4k+m\}$, $x_i = k/4 \cdot Q_{i-4k}^\varphi$, $y_i = 1$.

The size of $\mathcal{D}(\varphi)$ is $O((m+k)\log k)$ and $\mathcal{D}(\varphi)$ has separation bound $k/4.1 > 1$, because $k \geq 6$ for 6-SAT problem. Let the robustness radius be $\alpha = 0.5 - \gamma$, where $\gamma < \frac{1}{10k}$.

We claim that RobM$(\mathcal{D}(\varphi), 0.5-\gamma)$ **has a solution** $\mathcal{F}$ **if and only if the reversible 6-SAT** $\varphi(k,m)$ **has a solution** $J = \{x_j = v_j\}_{j=1}^k$**. Furthermore,** $\mathcal{F}$ **and** J **can be deduced from each other in polynomial time; that is,** RobM$(\mathcal{D}(\varphi), 0.5-\gamma)$ **is computationally equivalent to** $\varphi(k,m)$. Since reversible 6-SAT is NPC (Megiddo, 1988), by the claim, RobM$(\mathcal{D}(\varphi), 0.5-\gamma)$ is NPC, which implies that RobM$(\mathcal{D}(\varphi), \alpha)$ is NP-hard. This proves the theorem.

Before proving the claim, we first introduce a notation. Let $J = \{x_j = v_j\}_{j=1}^k$ be a solution to the reversible 6-SAT problem φ and $\varphi_i(k,m) = \vee_{j=1}^6 \tilde{x}_{i,j}$ a clause of φ, where $v_i \in \{-1, 1\}$. Then denote $q(J, \varphi_i)$ to be the number of $\tilde{x}_{i,j}$ which has value 1 on the solution J. If $q(J, \varphi_i) = 0$, then φ_i is not true. If $q(J, \varphi_i) = 6$, then $\neg\varphi_i$ is not true. Since J is a solution to the reversible 6-SAT problem φ, we have $1 \leq q(J, \varphi_i) \leq 5$. It is easy to see that $q(J, \varphi_i) = |\{j \in [k] : Q_i^\varphi[j] = v_j\}|$.

The claim will be proved in two steps.

Step 1. We prove that if $\varphi(k,m)$ **has a solution** $J = \{x_j = v_j\}_{j=1}^k$**, then** RobM$(\mathcal{D}(\varphi), 0.5-\gamma)$ **has a solution** $\mathcal{F}$**, where** $v_i \in \{-1, 1\}$**.** Let $U_1 = (v_1, v_2, \ldots, v_k)$, $U_2 = -(v_1, v_2, \ldots, v_k)$. Define $\mathcal{F} \in \mathcal{H}_{k,2}$ to be $\mathcal{F}(x) = \sigma(U_1 x - 1.5k) + \sigma(U_2 x - 1.5k) + 1.5 - \gamma$. It is clear that $\mathcal{F}$ can be obtained from J in Poly(k). We will show that $\mathcal{F}(x)$ is a robust memorization of $\mathcal{D}(\varphi)$ with budget $\alpha = 0.5 - \gamma$. The proof will be given in three steps: (c1) - (c3).

(c1). For $i \in [2k]$ and any $\epsilon \in [-0.5+\gamma, 0.5-\gamma]^k$, we have $U_1(x_i+\epsilon) - 1.5k \leq k|v_i| - 1.5k + (0.5-\gamma)||U_1||_1 = -\gamma k < 0$. Similarly, we have $U_2(x_i+\epsilon) - 1.5k < 0$. Then, for any $||\epsilon||_\infty \leq 0.5 - \gamma$, we have

$$\mathcal{F}(x_i+\epsilon) = \sigma(U_1(x_i+\epsilon) - 1.5k) + \sigma(U_2(x_i+\epsilon) - 1.5k) + 1.5 - \gamma < 0 + 0 + 1.5 = 1.5.$$

Thus $|\mathcal{F}(x) - 1| < |\mathcal{F}(x) - 2|$, so $\mathcal{F}$ is robust at x_i with budget $0.5 - \gamma$.

(c2). For $i \in \{2k+1, 2k+2, \ldots, 4k\}$, since $U_1 = -U_2$, at least one of the following two equations $U_1 x_i - 1.5k = -1.5k + |U_1||x_i| = 0.5k$ and $U_2 x_i - 1.5k = -1.5k + |U_2||x_i| = 0.5k$ is true, say the first one is true. Then, for any $||\epsilon||_\infty \leq 0.5 - \gamma$, we have

$$\mathcal{F}(x_i+\epsilon) = \sigma(U_1(x_i+\epsilon) - 1.5k) + \sigma(U_2(x_i+\epsilon) - 1.5k) - \gamma + 1.5 \geq \sigma(U_1(x_i+\epsilon) - 1.5k) - \gamma + 1.5.$$

We have $U_1(x_i+\epsilon) - 1.5k = 0.5k + U_1\epsilon \geq 0.5k - (0.5-\gamma)k = \gamma k$. So $\mathcal{F}(x_i+\epsilon) \geq \gamma k - \gamma + 1.5 > 1.5$, since $k > 1$.

Thus $|\mathcal{F}(x) - 1| > |\mathcal{F}(x) - 2|$, so $\mathcal{F}$ is robust at x_i with budget $0.5 - \gamma$.

(c3). Let $i \in \{4k+1, 4k+2, \ldots, 4k+m\}$. It is clear that $q(J, \varphi_{i-4k}) + q(J, \overline{\varphi}_{i-4k}) = 6$.

Then

$$\begin{aligned}
& \tfrac{k}{4}U_1Q^{\varphi}_{i-4k} \\
= \;& \textstyle\sum_{j\,:\,x_j\,\in\,\varphi_{i-4k}} \tfrac{k}{4}v_jQ^{\varphi}_{i-4k}[j] \\
= \;& \textstyle\sum_{j\,:\,x_j\,\in\,\varphi_{i-4k},\,\mathrm{Sgn}(Q^{\varphi}_{i-4k}[j])=\mathrm{Sgn}(v_j)} k/4 - \sum_{j\,:\,x_j\,\in\,\varphi_{i-4k},\,\mathrm{Sgn}(Q^{\varphi}_{i-4k}[j])\neq\mathrm{Sgn}(v_j)} k/4 \\
= \;& q(J,\varphi_{i-4k})k/4 - q(J,\overline{\varphi}_{i-4k})k/4 \\
\in \;& \{0, k/2, k, -k/2-k\}
\end{aligned}$$

which means $|\frac{k}{4}U_1Q^{\varphi}_{i-4k}| \leq k$. Similarly, we also have $|\frac{k}{4}U_2Q^{\varphi}_{i-4k}| \leq k$. As a consequence, $U_1x_i - 1.5k = -1.5k + U_1Q^{\varphi}_{i-4k}\cdot k/4 \leq -0.5k$, Since $||U_1||_1 = k$, for any $||\epsilon||_\infty \leq 0.5-\gamma$, we have $U_1(x_i+\epsilon)-1.5k \leq -0.5k+||U||_1(0.5-\gamma) = -\gamma k < 0$. Similarly, for U_2 we have that

$$\mathcal{F}(x_i+\epsilon) = \sigma(U_1(x_i+\epsilon)-1.5k) + \sigma(U_2(x_i+\epsilon)-1.5k) - \gamma + 1.5 \leq 0 + 0 - \gamma + 1.5 < 1.5.$$

Thus $\mathcal{F}$ is robust at x_i with budget 0.5.

From (c1) to (c3), $\mathcal{F}$ is a robustness memorization of $\mathcal{D}(\varphi)$ with budget $0.5-\gamma$, and Step 1 is proved.

Step 2. We prove that if $\mathrm{RobM}(\mathcal{D}(\varphi), 0.5-\gamma)$ **has a solution** $\mathcal{F}(x) = s_1\sigma(U_1x+b_1) + s_2\sigma(U_2x+b_2) + C \in \mathcal{H}_{k,2}$ **which is a robust memorization of** $\mathcal{D}(\varphi)$ **with budget** $\alpha = 0.5-\gamma$**, then** $\varphi(k,m)$ **has a solution.**

Since $a\sigma(b) = \mathrm{Sgn}(a)\sigma(|a|b)$, we can assume that $s_1, s_2 \in \{-1,1\}$. Moreover, if $\mathcal{F}(x) - 1.5 < 0$, then we have that $|\mathcal{F}(x)-1| < |\mathcal{F}(x)-2|$. Similarly, if $\mathcal{F}(x)-1.5 > 0$, then we have that $|\mathcal{F}(x)-1| > |\mathcal{F}(x)-2|$. So $\mathcal{F}(x)-1.5 > 0$ when $\widehat{\mathcal{F}}(x) = 2$, and $\mathcal{F}(x)-1.5 < 0$ when $\widehat{\mathcal{F}}(x) = 1$. Let $c = -1.5 + C$, which means $\mathcal{F}(x) - 1.5 = s_1\sigma(U_1x+b_1) + s_2\sigma(U_2x+b_2) + c$.

Step 2.1. Assuming $c \geq 0$**, we will show that** $J = \{x_i = \mathrm{Sgn}(U_1^{(i)})\}_{i=1}^k$ **is the solution to the reversible 6-SAT problem** $\varphi(k,m)$**.** We will prove Step 2.1 by proving six properties: (d1) - (d6).

(d1): We have $s_1s_2 = -1$. Without loss of generality, we let $s_1 = 1$ and $s_2 = -1$. (d1) can be proved by using Lemma A.2.

(d2): $-k|U_2^{(q)}| + b_2 + p||U_2||_1 > 0$ for any $p \in [-0.5+\gamma, 0.5-\gamma]$ and $q \in [k]$.

We just need to prove the case $q = 1$. Without loss of generality, we assume $U_2^{(1)} \leq 0$. We know that $\mathcal{F}(x_1 + p\mathrm{Sgn}(U_2)) < 1.5$, so

$$\begin{aligned}
& 0 \\
> \;& \sigma(U_1(x_1 + p\mathrm{Sgn}(U_2)) + b_1) - \sigma(U_2(x_1 + p\mathrm{Sgn}(U_2)) + b_2) + c \\
\geq \;& -\sigma(U_2(x_1 + p\mathrm{Sgn}(U_2)) + b_2) + c \\
= \;& -\sigma(-k|U_2^{(1)}| + p||U_2||_1 + b_2) + c \\
\geq \;& -\sigma(-k|U_2^{(1)}| + p||U_2||_1 + b_2) \quad (by\ c \geq 0)
\end{aligned}$$

which means $\sigma(-k|U_2^{(1)}| + p||U_2||_1 + b_2) > 0$. Then we have $-k|U_2^{(1)}| + p||U_2||_1 + b_2 > 0$. For $U_2^{(1)} \geq 0$, we just need to consider x_{k+1}. So (d2) is proved.

(d3): $U_1^{(q)}U_2^{(q)} > 0$ and $|U_1^{(q)}| > |U_2^{(q)}|$ for any $q \in [k]$.

We just need to prove it for $q = 1$. Firstly, we show that $U_2^{(1)} \neq 0$. If $U_2^{(1)} = 0$. Without loss of generality, let $U_1^{(1)} \leq 0$. Since $\mathcal{F}(x_1) < 1.5$ and $\mathcal{F}(x_{2k+1}) > 1.5$, we have

$$\begin{aligned}
0 &> \mathcal{F}(x_1) - 1.5 \\
&= \sigma(U_1 x_1 + b_1) - \sigma(U_2 x_1 + b_2) + c \\
&= \sigma(kU_1^{(1)} + b_1) - \sigma(b_2) + c \quad (by\ U_2^{(1)} = 0) \\
&\geq \sigma(2kU_1^{(1)} + b_1) - \sigma(b_2) + c \quad (by\ U_1^{(1)} \leq 0) \\
&= \sigma(2kU_1^{(1)} + b_1) - \sigma(2kU_2^{(1)} + b_2) + c \quad (by\ U_2^{(1)} = 0) \\
&= \sigma(U_1 x_{2k+1} + b_1) - \sigma(U_2 x_{2k+1} + b_2) + c \\
&= \mathcal{F}(x_{2k+1}) - 1.5 \\
&> 0
\end{aligned}$$

which means $0 > 0$, so the assumption incorrect, and thus $U_2^{(1)} \neq 0$. When $U_1^{(1)} \geq 0$, just need to consider x_{k+1} and x_{3k+1}.

Now we prove (d3). Let $h = 1$ if $U_2^{(1)} > 0$, $h = k+1$ if $U_2^{(1)} < 0$. Because $x_{h+2k} = 2x_h$ and $\mathcal{F}(x_{h+2k}) - 1.5 > 0 > \mathcal{F}(x_h) - 1.5$, we have that:

$$\begin{aligned}
& \mathcal{F}(x_{h+2k}) - 1.5 \\
= \ & \sigma(U_1 x_{h+2k} + b_1) - \sigma(U_2 x_{h+2k} + b_2) + c \\
= \ & \sigma(2U_1 x_h + b_1) - \sigma(2U_2 x_h + b_2) + c \\
= \ & \sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(2k|U_2^{(1)}| + b_2) + c \\
> \ & 0 \quad (by\ \mathcal{F}(x_{h+2k}) - 1.5 > 0) \\
> \ & \sigma(U_1 x_h + b_1) - \sigma(U_2 x_h + b_2) + c \quad (by\ \mathcal{F}(x_h) - 1.5 < 0) \\
= \ & \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(k|U_2^{(1)}| + b_2) + c
\end{aligned}$$

which means $\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(2k|U_2^{(1)}| + b_2) > \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(k|U_2^{(1)}| + b_2)$, so we have

$$\begin{aligned}
0 &< \sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(2k|U_2^{(1)}| + b_2) - (\sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(k|U_2^{(1)}| + b_2)) \\
&= (\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - (\sigma(2k|U_2^{(1)}| + b_2) - \sigma(k|U_2^{(1)}| + b_2)) \\
&= (\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - ((2k|U_2^{(1)}| + b_2) - (k|U_2^{(1)}| + b_2)) \quad (by\ (d2)) \\
&= (\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - kU_2^{(1)}.
\end{aligned}$$

Then $\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) > kU_2^{(1)} \geq 0$, which means $2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1 > 0$. And according to that, we have:

$$\begin{aligned}
0 &< (\sigma(2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - k|U_2^{(1)}| \\
&= ((2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - \sigma(kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - k|U_2^{(1)}| \\
&\leq ((2kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1) - (kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) + b_1)) - k|U_2^{(1)}| \\
&= kU_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) - k|U_2^{(1)}|.
\end{aligned}$$

So we get $U_1^{(1)}\mathrm{Sgn}(U_2^{(1)}) > |U_2^{(1)}| > 0$, which means $\mathrm{Sgn}(U_1^{(1)}) = \mathrm{Sgn}(U_2^{(1)})$, and $|U_1^{(1)}| > |U_2^{(1)}|$. (d3) is proved.

(d4): $2k|U_1^{(q)}| + b_1 + p||U_1||_1 > 0$ for any $p \in [-0.5 + \gamma, 0.5 - \gamma]$ and $q \in [k]$.

We just need to prove it for $q = 1$. Let $h = 1$ if $U_1^{(1)} > 0$, $h = k+1$ if $U_1^{(1)} < 0$. Because $\mathcal{F}(x_{h+2k} + p\mathrm{Sgn}(U_1)) - 1.5 >$

$0 > \mathcal{F}(x_h + p\text{Sgn}(U_1)) - 1.5$, we have that:

$$\begin{aligned}
& \mathcal{F}(x_{h+2k} + p\text{Sgn}(U_1)) - 1.5 \\
= \;& \sigma(U_1(x_{h+2k} + p\text{Sgn}(U_1)) + b_1) - \sigma(U_2(2x_{h+2k} + p\text{Sgn}(U_1)) + b_2) + c \\
= \;& \sigma(2k|U_1^{(1)}| + b_1 + p||U_1||_1) - \sigma(2k|U_2^{(1)}| + b_2 + p||U_2||_1) + c \quad (by\ (d3)) \\
= \;& \sigma(2k|U_1^{(1)}| + b_1 + p||U_1||_1) - (2k|U_2^{(1)}| + b_2 + p||U_2||_1) + c \quad (by\ (d2)) \\
> \;& 0 \quad (by\ \mathcal{F}(x_{h+2k} + p\text{Sgn}(U_1)) - 1.5 > 0) \\
> \;& \sigma(U_1(x_h + p\text{Sgn}(U_1)) + b_1) - \sigma(U_2(x_h + p\text{Sgn}(U_1)) + b_2) + c \quad (by\ 0 > \mathcal{F}(x_h + p\text{Sgn}(U_1)) - 1.5) \\
= \;& \sigma(k|U_1^{(1)}| + b_1 + p||U_1||_1) - \sigma(k|U_2^{(1)}| + b_2 + p||U_2||_1) + c \quad (by\ (d3)) \\
= \;& \sigma(k|U_1^{(1)}| + b_1 + p||U_1||_1) - (k|U_2^{(1)}| + b_2 + p||U_2||_1) + c \quad (by\ (d2))
\end{aligned}$$

which means $\sigma(2k|U_1^{(1)}| + b_1 + p||U_1||_1) - (2k|U_2^{(1)}| + b_2 + p||U_2||_1) + c > \sigma(k|U_1^{(1)}| + b_1 + p||U_1||_1) - (k|U_2^{(1)}| + b_2 + p||U_2||_1) + c$, so we have that:

$$\sigma(2k|U_1^{(1)}| + b_1 + p||U_1||_1) > \sigma(k|U_1^{(1)}| + b_1 + p||U_1||_1) + k|U_2^{(1)}| > 0.$$

(d4) is proved.

(d5): $\max_{z\in[k]}(|U_1^{(z)}| - |U_2^{(z)}|) < 2(1-2\gamma)(||U_1||_1 - ||U_2||_1)$.

For any $z \in [k]$, let $h = z$ if $U_1^{(z)} > 0$, or $h = z + k$ if $U_1^{(z)} < 0$. We have $\mathcal{F}(x_h + (0.5-\gamma)\text{Sgn}(U_1)) - 1.5 < 0$, which means

$$\begin{aligned}
& 0 \\
> \;& \sigma(U_1(x_h + (0.5-\gamma)\text{Sgn}(U_1)) + b_1) - \sigma(U_2(x_h + (0.5-\gamma)\text{Sgn}(U_1)) + b_2) + c \\
= \;& \sigma(k|U_1^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) - \sigma(k|U_2^{(z)}| + (0.5-\gamma)||U_2||_1 + b_2) + c \quad (by\ (d3)) \\
= \;& \sigma(k|U_1^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) - (k|U_2^{(z)}| + (0.5-\gamma)||U_2||_1 + b_2) + c \quad (by\ (d2)) \\
\geq \;& (k|U_1^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) - (k|U_2^{(z)}| + (0.5-\gamma)||U_2||_1 + b_2) + c \\
= \;& k|U_1^{(z)}| - k|U_2^{(z)}| + (0.5-\gamma)(||U_1||_1 - ||U_2||_1) + b_1 - b_2 + c.
\end{aligned}$$

We thus have $k|U_1^{(z)}| - k|U_2^{(z)}| < -b_1 + b_2 - c - (0.5-\gamma)(||U_1||_1 - ||U_2||_1)$. Then we have $\mathcal{F}(x_{h+2k} - (0.5-\gamma)\text{Sgn}(U_1)) - 1.5 > 0$, which means

$$\begin{aligned}
& 0 \\
< \;& \sigma(U_1(x_{2k+h} - (0.5-\gamma)\text{Sgn}(U_1)) + b_1) - \sigma(U_2(x_{2k+h} - (0.5-\gamma)\text{Sgn}(U_1)) + b_2) + c \\
= \;& \sigma(2k|U_1^{(z)}| - (0.5-\gamma)||U_1||_1 + b_1) - \sigma(2k|U_2^{(z)}| - (0.5-\gamma)||U_2||_1 + b_2) + c \quad (by\ (d3)) \\
= \;& (2k|U_1^{(z)}| - (0.5-\gamma)||U_1||_1 + b_1) - \sigma(2k|U_2^{(z)}| - (0.5-\gamma)||U_2||_1 + b_2) + c \quad (by\ (d4)) \\
\leq \;& (2k|U_1^{(z)}| - (0.5-\gamma)||U_1||_1 + b_1) - (2k|U_2^{(z)}| - (0.5-\gamma)||U_2||_1 + b_2) + c \\
= \;& 2k|U_1^{(z)}| - 2k|U_2^{(z)}| - (0.5-\gamma)(||U_1||_1 - ||U_2||_1) + b_1 - b_2 + c.
\end{aligned}$$

So, we have $k|U_1^{(z)}| - k|U_2^{(z)}| > \frac{-b_1+b_2-c+(0.5-\gamma)(||U_1||_1-||U_2||_1)}{2}$, and thus

$$\begin{aligned}
& k|U_1^{(z)}| - k|U_2^{(z)}| \\
< \;& -b_1 + b_2 - c - (0.5-\gamma)(||U_1||_1 - ||U_2||_1) \\
= \;& 2\frac{-b_1+b_2-c+(0.5-\gamma)(||U_1||_1-||U_2||_1)}{2} - (1-2\gamma)(||U_1||_1 - ||U_2||_1) \\
< \;& 2k|U_1^{(z)}| - 2k|U_2^{(z)}| - (1-2\gamma)(||U_1||_1 - ||U_2||_1)
\end{aligned}$$

which means $k|U_1^{(z)}| - k|U_2^{(z)}| > (1-2\gamma)(||U_1||_1 - ||U_2||_1)$ for any $z \in [k]$. Using this inequality, we have

$$\begin{aligned} & k|U_1^{(z)}| - k|U_2^{(z)}| \\ = \; & k(||U_1||_1 - ||U_2||_1) - \textstyle\sum_{z' \neq z}(k|U_1^{z'}| - k|U_2^{z'}|) \\ < \; & (k - (1-2\gamma)(k-1))(||U_1||_1 - ||U_2||_1) \\ < \; & 1.1(||U_1||_1 - ||U_2||_1) \quad (by\ (d3)\ and\ \gamma < 1/(10k)) \\ < \; & 2 * (1-2\gamma)(||U_1||_1 - ||U_2||_1) \quad (by\ (d3)) \end{aligned}$$

which proves (d5).

(d6): $\{x_i = \text{Sgn}(U_1^{(i)})\}_{i=1}^k$ is the solution to the reversible 6-SAT problem $\varphi(k, m)$.

If this not valid, there exists an $i \in [m]$ such that $q(\{\text{Sgn}(U_1^{(w)})\}_{w=1}^k, \phi_i) = 6$ or $q(\{\text{Sgn}(U_1^{(w)})\}_{w=1}^k, \phi_i) = 0$. We just need to consider the first case, because when $q(\{\text{Sgn}(U_1^{(w)})\}_{w=1}^k, \phi_i) = 0$, there exists a $j \in [m]$ such that $\overline{\phi_j} = \phi_i$, so $q(\{\text{Sgn}(U_1^{(w)})\}_{w=1}^k, \phi_j) = 6$.

Without loss of generality, we assume that the index of the six entries in ϕ_i are $1, 2, 3, 4, 5, 6$. By the definition of x_{4k+i}, we know that $U_1 x_{4k+i} = \frac{k}{4}\sum_{z=1}^6 |U_1^{(z)}|$, and by (d3), we know that $U_2 x_{4k+i} = \frac{k}{4}\sum_{z=1}^6 |U_2^{(z)}|$.

Using (d2), we know that

$$0 < -k|U_2^{(1)}| + b_2 + (0.5-\gamma)||U_2||_1 < \tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_2^{(z)}| + b_2 + (0.5-\gamma)||U_2||_1. \tag{24}$$

Without loss of generality, we assume $U_1^{(1)} > 0$. Since $\mathcal{F}(x_{2k+1} - (0.5-\gamma)\text{Sgn}(U_1)) - 1.5 > 0$, we have

$$\begin{aligned} & 0 \\ < \; & \sigma(U_1(x_{2k+1} - (0.5-\gamma)\text{Sgn}(U_1)) + b_1) - \sigma(U_2(2x_{2k+1} - (0.5-\gamma)\text{Sgn}(U_1)) + b_2) + c \\ = \; & \sigma(2k|U_1^{(1)}| + b_1 - (0.5-\gamma)||U_1||_1) - \sigma(2k|U_2^{(1)}| + b_2 - (0.5-\gamma)||U_2||_1) + c \quad (by\ (d3)) \\ \leq \; & \sigma(2k|U_1^{(1)}| + b_1 - (0.5-\gamma)||U_1||_1) - (2k|U_2^{(1)}| + b_2 - (0.5-\gamma)||U_2||_1) + c \\ = \; & (2k|U_1^{(1)}| + b_1 - (0.5-\gamma)||U_1||_1) - (2k|U_2^{(1)}| + b_2 - (0.5-\gamma)||U_2||_1) + c \quad (by\ (d4)). \end{aligned} \tag{25}$$

So $0 < (2k|U_1^{(1)}| + b_1 - (0.5-\gamma)||U_1||_1) - (2k|U_2^{(1)}| + b_2 - (0.5-\gamma)||U_2||_1) + c$. If $U_1^{(1)} < 0$. We just need to consider x_{3k+1}, and others are the same. For $U_1^2, \ldots, U_1^6$, the conclusion is the same.

Then because $\mathcal{F}(x_{4k+i} + (0.5-\gamma)\text{Sgn}(U_1)) - 1.5 < 0$, we have that:

$$\begin{aligned} & 0 \\ > \; & \sigma(U_1(x_{4k+i} + (0.5-\gamma)\text{Sgn}(U_1)) + b_1) - \sigma(U_2(x_{4k+i} + (0.5-\gamma)\text{Sgn}(U_1)) + b_2) + c \\ = \; & \sigma(U_1 x_{4k+i} + b_1 + (0.5-\gamma)||U_1||_1) - \sigma(U_2 x_{4k+i} + b_2 + (0.5-\gamma)||U_2||_1) + c \quad (by\ (d3)) \\ = \; & \sigma(\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_1^{(z)}| + b_1 + (0.5-\gamma)||U_1||_1) - \sigma(\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_2^{(z)}| + b_2 + (0.5-\gamma)||U_2||_1) + c \\ = \; & \sigma(\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_1^{(z)}| + b_1 + (0.5-\gamma)||U_1||_1) - (\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_2^{(z)}| + b_2 + (0.5-\gamma)||U_2||_1) + c \quad (by\ (24)) \\ \geq \; & \tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_1^{(z)}| + b_1 + (0.5-\gamma)||U_1||_1 - (\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_2^{(z)}| + b_2 + (0.5-\gamma)||U_2||_1) + c \\ = \; & \tfrac{1}{6}\textstyle\sum_{z=1}^6 (2k|U_1^{(z)}| + b_1 - (0.5-\gamma)||U_1||_1 - 2k|U_2^{(z)}| - b_2 + (0.5-\gamma)||U_2||_1 + c) \\ & -\tfrac{k}{12}\textstyle\sum_{k=1}^6 (|U_1^{(z)}| - |U_2^{(z)}|) + (1-2\gamma)(||U_1||_1 - ||U_2||_1) \\ \geq \; & -\tfrac{k}{12}\textstyle\sum_{k=1}^6 (|U_1^k| - |U_2^k|) + (1-2\gamma)(||U_1||_1 - ||U_2||_1) \quad (by\ (25)) \\ > \; & 0 \quad \text{(by (d5))} \end{aligned}$$

Which means $0 > 0$, a contradiction and Step 2.1 is proved.

Step 2.2. Assuming $c<0$, we will show that $J=\{x_i=\text{Sgn}(U_1^{(i)})\}_{i=1}^k$ is the solution to the reversible 6-SAT problem $\varphi(k,m)$. The proof is divided into six steps: (e1) - (e6).

(e1): There must be $s_1=s_2=1$.

Just use Lemma A.2.

(e2): $U_1^{(q)}U_2^{(q)}<0$ for any $q\in[k]$.

We just need to prove it for $q=1$. First we prove that $U_1^{(1)}\neq 0$. If not, that is $U_1^{(1)}=0$. Without loss of generality, let $U_2^{(1)}\leq 0$. Since $\mathcal{F}(x_1)-1.5<0$ and $\mathcal{F}(x_{2k+1})-1.5>0$, we have

$$\begin{aligned}
&0\\
>\;& \mathcal{F}(x_1)-1.5\\
=\;& \sigma(U_1x_1+b_1)+\sigma(U_2x_1+b_2)+c\\
=\;& \sigma(b_1)+\sigma(kU_2^{(1)}+b_2)+c \quad (by\ U_1^{(1)}=0)\\
\geq\;& \sigma(b_1)+\sigma(2kU_2^{(1)}+b_2)+c \quad (by\ U_2^{(1)}\leq 0)\\
=\;& \sigma(2kU_1^{(1)}+b_1)+\sigma(2kU_2^{(1)}+b_2)+c \quad (by\ U_1^{(1)}=0)\\
=\;& \sigma(U_1x_{2k+1}+b_1)-\sigma(U_2x_{2k+1}+b_2)+c\\
=\;& \mathcal{F}(x_{2k+1})-1.5\\
>\;& 0
\end{aligned}$$

which means $0>0$, a contradiction, and hence $U_1^{(1)}\neq 0$. When $U_2^{(1)}\geq 0$, we just need to consider x_{k+1} and x_{3k+1}.

Now we prove (e2), let $h=1+k$ if $U_1^{(1)}>0$, or $h=1$ if $U_1^{(1)}<0$. Then, we have that $\mathcal{F}(x_{h+2k})-1.5>0$ and $\mathcal{F}(x_h)-1.5<0$, which means

$$\begin{aligned}
&\sigma(U_1x_{h+2k}+b_1)+\sigma(U_2x_{h+2k}+b_2)-c\\
=\;& \sigma(-2k|U_1^{(1)}|+b_1)+\sigma(-2kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)-c\\
>\;& 0
\end{aligned}\tag{26}$$

and

$$\begin{aligned}
&\sigma(U_1x_h+b_1)+\sigma(U_2x_h+b_2)-c\\
=\;& \sigma(-k|U_1^{(1)}|+b_1)+\sigma(-kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)-c\\
<\;& 0.
\end{aligned}\tag{27}$$

These two inequalities illustrate that $\sigma(-k|U_1^{(1)}|+b_1)+\sigma(-kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)<\sigma(-2k|U_1^{(1)}|+b_1)+\sigma(-2kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)$. Furthermore, since $\sigma(-k|U_1^{(1)}|+b_1)\geq\sigma(-2k|U_1^{(1)}|+b_1)$, we have $\sigma(-kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)<\sigma(-2kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)$, which means $(-kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)<(-2kU_2^{(1)}\text{Sgn}(U_1^{(1)})+b_2)$. Then $U_2^{(1)}\text{Sgn}(U_1^{(1)})<0$, that is $U_1^{(1)}U_2^{(1)}<0$, which is what we want.

(e3): $k|U_2^{(q)}|>(1-2\gamma)||U_2||_1$ for any $q\in[k]$.

We just need to prove it for $q=1$. Let $h=1$ if $U_2^{(1)}>0$, or $h=k+1$ if $U_2^{(1)}<0$. We have $\mathcal{F}(x_{h+2k}-(0.5-\gamma)\text{Sgn}(U_2))-1.5>0$ and $\mathcal{F}(x_h+(0.5-\gamma)\text{Sgn}(U_2))-1.5<0$, which means

$$\begin{aligned}
&\sigma(U_1(x_{h+2k}-(0.5-\gamma)\text{Sgn}(U_2))+b_1)+\sigma(U_2(x_{h+2k}-(0.5-\gamma)\text{Sgn}(U_2))+b_2)-c\\
=\;& \sigma(-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \quad (by\ (e2))\\
>\;& 0
\end{aligned}\tag{28}$$

and

$$\begin{aligned} & \sigma(U_1(x_h+(0.5-\gamma)\mathrm{Sgn}(U_2))+b_1)+\sigma(U_2(x_h+(0.5-\gamma)\mathrm{Sgn}(U_2))+b_2)-c \\ = \;& \sigma(-k|U_1^{(1)}|-(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)-c \quad (by\ (e2)) \\ < \;& 0. \end{aligned} \tag{29}$$

Next, consider two situations:
(e3.1): If $-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1\le 0$, then we can prove that $k|U_2^{(1)}|>(1-2\gamma)||U_2||_1$.

By (28) and $-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1\le 0$, we have $\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c>0$.

By (29), we have $\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)-c\le\sigma(-k|U_1^{(1)}|-(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)-c<0$.

So $\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)<c<\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)$, which means $(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)<(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)$. Then we get that: $(1-2\gamma)||U_2||_1<k|U_2^{(1)}|$. This is what we want.
(e3.2): If $-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1>0$, then we can prove $-2k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2\le 0$ and $k|U_2^{(1)}|>(1-2\gamma)||U_2||_1$.

Since $\mathcal{F}(x_h-(0.5-\gamma)\mathrm{Sgn}(U_2))=-1<0$, we have that

$$\begin{aligned} & \sigma(U_1(x_h-(0.5-\gamma)\mathrm{Sgn}(U_2))+b_1)+\sigma(U_2(x_h-(0.5-\gamma)\mathrm{Sgn}(U_2))+b_2)-c \\ = \;& \sigma(-k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \quad (by\ (e2)) \\ < \;& 0. \end{aligned} \tag{30}$$

Since $-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1>0$, it holds $-k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1>0$. Then by (28) and (30) and $-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1>0$, we have that

$$\begin{aligned} & \sigma(-k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \\ = \;& (-k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \\ < \;& 0 \\ < \;& \sigma(-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \\ = \;& (-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \end{aligned} \tag{31}$$

which means $k|U_1^{(1)}|<\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-\sigma(k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)\le k|U_2^{(1)}|$ (Use $\sigma(x)-\sigma(y)\le|x-y|$ here). So $|U_1^{(1)}|<|U_2^{(1)}|$.

Similarly, if $-2k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2>0$, then we have $|U_1^{(1)}|>|U_2^{(1)}|$. But $|U_1^{(1)}|<|U_2^{(1)}|$ and $|U_1^{(1)}|>|U_2^{(1)}|$ cannot stand simultaneously, so $-2k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2>0$ can not stand. Then we have $-2k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2\le 0$.

Now using (28) and (29), we have

$$\begin{aligned} & \sigma(-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \\ = \;& (-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-c \\ > \;& 0 \\ > \;& \sigma(-k|U_1^{(1)}|-(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)-c \\ \ge \;& (-k|U_1^{(1)}|-(0.5-\gamma)||U_1||_1+b_1)+\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)-c \end{aligned}$$

which means $(-k|U_1^{(1)}|-(0.5-\gamma)||U_1||_1+b_1)-(-2k|U_1^{(1)}|+(0.5-\gamma)||U_1||_1+b_1)<\sigma(2k|U_2^{(1)}|-(0.5-\gamma)||U_2||_1+b_2)-\sigma(k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2)$. Since $-2k|U_2^{(1)}|+(0.5-\gamma)||U_2||_1+b_2\le 0$, similar to (e3.1), we have $(1-2\gamma)||U_1||_1<k|U_1^{(1)}|$.

So we can obtain

$$\begin{aligned} & 0 \\ < & -(1-2\gamma)||U_1||_1 + k|U_1^{(1)}| \quad (by\ equc3) \\ = & (-k|U_1^{(1)}| - (0.5-\gamma)||U_1||_1 + b_1) - (-2k|U_1^{(1)}| + (0.5-\gamma)||U_1||_1 + b_1) \\ < & \sigma(2k|U_2^{(1)}| - (0.5-\gamma)||U_2||_1 + b_2) - \sigma(k|U_2^{(1)}| + (0.5-\gamma)||U_2||_1 + b_2) \end{aligned}$$

which implies $\sigma(2k|U_2^{(1)}| - (0.5-\gamma)||U_2||_1 + b_2) > 0$. Then we have

$$\begin{aligned} & 0 \\ < & \sigma(2k|U_2^{(1)}| - (0.5-\gamma)||U_2||_1 + b_2) - \sigma(k|U_2^{(1)}| + (0.5-\gamma)||U_2||_1 + b_2) \\ = & (2k|U_2^{(1)}| - (0.5-\gamma)||U_2||_1 + b_2) - \sigma(k|U_2^{(1)}| + (0.5-\gamma)||U_2||_1 + b_2) \\ \leq & (2k|U_2^{(1)}| - (0.5-\gamma)||U_2||_1 + b_2) - (k|U_2^{(1)}| + (0.5-\gamma)||U_2||_1 + b_2) \\ = & k|U_2^{(1)}| - (1-2\gamma)||U_2||_1. \end{aligned}$$

This is what we want.

(e4): $k|U_1^{(q)}| > (1-2\gamma)||U_1||_1$ for any $q \in [k]$.

Similar to (e3).

(e5): $J = \{x_i = \text{Sgn}(U_1^{(i)})\}_{i=1}^k$ is the solution to the reversible 6-SAT problem $\varphi(k, m)$.

If not, as said in (d6), there is an $i \in [m]$ such that $q(\{\text{Sgn}(U_1^{(w)})\}_{w=1}^k, \phi_i) = 6$. And there is a $j \in [k]$ such that $\overline{\phi_j} = \phi_i$.

Without loss of generality, we assume that the index of the six entries in ϕ_i are $1, 2, 3, 4, 5, 6$. By the definition of x_{4k+i}, we know that $U_1 x_{4k+i} = \frac{k}{4}\sum_{z=1}^6 |U_1^{(z)}|$, and by (e2), we know that $U_2 x_{4k+i} = -\frac{k}{4}\sum_{z=1}^6 |U_2^{(z)}|$. By the definition of x_{4k+j}, we know that $U_1 x_{4k+j} = -\frac{k}{4}\sum_{z=1}^6 |U_1^{(z)}|$, and by (e2), we know that $U_2 x_{4k+j} = \frac{k}{4}\sum_{z=1}^6 |U_2^{(z)}|$.

As said in (e3.2), we have $-2k|U_1^z| + (0.5-\gamma)||U_1||_1 + b_1 < 0$ or $-2k|U_2^z| + (0.5-\gamma)||U_2||_1 + b_2 < 0$ stand for any $z \in [k]$. Let the last one stands for $z = 7$. (If the first one stands, it is similar.)

Now we will show that

$$\begin{aligned} & \sigma(2k|U_1^{(7)}| - (0.5-\gamma)||U_1||_1 + b_1) \\ < & \sigma(\tfrac{k}{4}\textstyle\sum_{z=1}^6 |U_1^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) + \sigma(-\sum_{z=1}^6 |U_2^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) \\ < & \sigma(2k|U_1^{(7)}| - (0.5-\gamma)||U_1||_1 + b_1), \end{aligned} \tag{32}$$

which lead to a contradiction.

(e5.1): We prove that $\sigma(\frac{k}{4}\sum_{z=1}^6 |U_1^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) + \sigma(-\frac{k}{4}\sum_{z=1}^6 |U_2^{(z)}| + (0.5-\gamma)||U_1||_1 + b_1) < \sigma(2k|U_1^{(7)}| - (0.5-\gamma)||U_1||_1 + b_1)$.

Let $h = 7$ if $U_1^{(7)} > 0$, and $h = k + 7$ if $U_1^{(7)} < 0$. Because $\mathcal{F}(x_{2k+h} - (0.5-\gamma)\text{Sgn}(U_1)) - 1.5 > 0$ and $-2k|U_2^{(7)}| + (0.5-\gamma)||U_2||_1 + b_2 - 1.5 < 0$, we have

$$\begin{aligned} & \sigma(U_1(x_{h+2k} - (0.5-\gamma)\text{Sgn}(U_1)) + b_1) + \sigma(U_2(x_{h+2k} - (0.5-\gamma)\text{Sgn}(U_1)) + b_2) + c \\ = & \sigma(2k|U_1^{(7)}| - (0.5-\gamma)||U_1||_1 + b_1) + \sigma(-2k|U_2^{(7)}| + (0.5-\gamma)||U_2||_1 + b_2) + c \quad (by\ (e2)) \\ = & \sigma(2k|U_1^{(7)}| - (0.5-\gamma)||U_1||_1 + b_1) + c \\ > & 0. \end{aligned} \tag{33}$$

Because $\mathcal{F}(x_{4k+i}+(0.5-\gamma)\text{Sgn}(U_1))-1.5<0$, we have that:

$$\begin{aligned} & \sigma(U_1(x_{4k+i}+(0.5-\gamma)\text{Sgn}(U_1))+b_1)+\sigma(U_2(x_{4k+i}+(0.5-\gamma)\text{Sgn}(U_1))+b_2)+c \\ = \; & \sigma(\tfrac{k}{4}\textstyle\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(-\sum_{z=1}^{6}|U_2^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)+c \quad (by\ (e2)) \\ < \; & 0. \end{aligned} \tag{34}$$

By (33) and (34), it holds $\sigma(\frac{k}{4}\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(-\sum_{z=1}^{6}|U_2^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)<\sigma(2k|U_1^{(7)}|-(0.5-\gamma)||U_1||_1+b_1)$. This is what we want.

(e5.2) We prove that $\sigma(\frac{k}{4}\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(-\sum_{z=1}^{6}|U_2^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)>\sigma(2k|U_1^{(7)}|-(0.5-\gamma)||U_1||_1+b_1)$.

By (e4), we have that:

$$\begin{aligned} & 2k|U_1^{(7)}|-(0.5-\gamma)||U_1||_1 \\ = \; & 2k(||U_1||_1-\textstyle\sum_{z\neq 7}|U_1^{(z)}|)-(0.5-\gamma)||U_1||_1 \\ < \; & 2k(||U_1||_1-(k-1)(1-2\gamma)||U_1||_1/k)-(0.5-\gamma)||U_1||_1 \quad (by\ (e4)) \\ = \; & (1.5+4\gamma k-3\gamma)||U_1||_1 \\ < \; & (1.5(1-2\gamma)+(0.5-\gamma))||U_1||_1 \quad (by\ \gamma<1/(10k)) \\ < \; & \tfrac{k}{4}\textstyle\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1 \quad (by\ (e4)). \end{aligned}$$

So $\sigma(\frac{k}{4}\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)>\sigma(2k|U_1^{(7)}|-(0.5-\gamma)||U_1||_1+b_1)$. Then $\sigma(\frac{k}{4}\sum_{z=1}^{6}|U_1^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)+\sigma(-\sum_{z=1}^{6}|U_2^{(z)}|+(0.5-\gamma)||U_1||_1+b_1)>\sigma(2k|U_1^{(7)}|-(0.5-\gamma)||U_1||_1+b_1)$. (e5.2) is proved.

From (e5.1) and (e5.2), the assumption is wrong and (e5) is proved. □

A.2. Proof of Proposition 3.6

We restate Proposition 3.6 for convenience.

Proposition A.4. *If* $\mathbf{H}=\{\mathcal{F}:\mathbb{R}^n\to\mathbb{R}, \text{width}(\mathcal{F})=w\}$ *is an optimal robust memorization of any dataset* $\mathcal{D}\in\mathcal{D}_{n,N,L}$ *with* $N>n$*, then* $\text{width}(\mathcal{F})=w\geq n$.

Proof. It suffices to show that there exists a dataset $\mathcal{D}$ such that, if $\mathcal{F}$ has width less than n and memorizes $\mathcal{D}$, then $\text{RA}_{\mathcal{D}}(\mathcal{F},0.4\lambda_{\mathcal{D}})\leq 1-\frac{1}{n+1}$; that is, $\mathcal{F}$ is not a robust memorization of $\mathcal{D}$ with budget $0.4\lambda_{\mathcal{D}}$.

Denote $\mathbf{1}$ to be the vector all of whose weights are 1 and $\mathbf{1}_k$ the vector whose k-th weight is 1 and all other weights are 0. Without loss of generality, let N satisfy $(n+1)|N$. We define a dataset $\mathcal{D}=\{x_i,y_i\}_{i=1}^N$ with separation bound 1 as follows:

(1) $x_1=0$ and $y_1=0$; $x_i=\mathbf{1}_{i-1}$ and $y_i=1$ for $i=2,3,\ldots,n+1$;

(2) for $i=k(n+1)+1,\ldots,k(n+1)+n+1$ and $k=1,\ldots,\frac{N}{n+1}-1$, $x_i=x_{\bar{i}}+\mathbf{1}$ and $y_i=y_{\bar{i}}$, where $\bar{i}=i \mod (n+1)$ if $(n+1)\not| i$ and $\bar{i}=n+1$ otherwise.
It is easy to see that $\lambda_{\mathcal{D}}=1$.

Let $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}$ be a network which memorizes $\mathcal{D}$ defined above. Let W_1 be the weight matrix of the first layer of $\mathcal{F}$. Then $W_1\in\mathbb{R}^{K\times n}$. We will show that, there exists an s in $[n]$ such that

$$\exists\delta_1,\delta_s\in\mathbb{R}^n, \text{ satisfying } ||\delta_1||_\infty<0.4, ||\delta_s||_\infty<0.4, W_1(x_1+\delta_1)=W_1(x_s+\delta_s).$$

Firstly, since $n>K$, $W_1\in\mathbb{R}^{K\times n}$ is not of full row rank, and hence there exists a vector $v\in\mathbb{R}^n$ such that $W_1v=0$ and $||v||_\infty=1$. For such a v, let $|v^{(s)}|=1$ for some $s\in[n]$. We define $\delta_1,\delta_s\in\mathbb{R}^n$ as follows:

$$\delta_1^{(s)}=1/3 \text{ and } \delta_1^{(k)}=-v^{(s)}v^{(k)}/3 \text{ for } k\neq s; \quad \delta_s^{(s)}=0 \text{ and } \delta_s^{(k)}=v^{(s)}v^{(k)}/3 \text{ for } k\neq s.$$

It is clearly that $||\delta_1||_\infty = \frac{1}{3} < 0.4$ and $||\delta_s||_\infty = \frac{1}{3} < 0.4$. Also, $x_s + \delta_s - x_1 - \delta_1 = \frac{2}{3}v^{(s)}v$. Thus, $W_1(x_1 + \delta_1) - W_1(x_s + \delta_s) = W_1(x_1 + \delta_1 - x_s - \delta_s) = W_1(\frac{2}{3}v^{(s)}v) = 0$.

It is easy to see that, for any $x, z \in \mathbb{R}^n$, $W_1 x = W_1 z$ implies $\mathcal{F}(x) = \mathcal{F}(z)$. Since $W_1(x_1 + \delta_1) = W_1(x_s + \delta_s)$, we have $\mathcal{F}(x_1 + \delta_1) = \mathcal{F}(x_s + \delta_s)$, and either $\mathcal{F}(x_1 + \delta_1) \neq 0$ or $\mathcal{F}(x_s + \delta_s) \neq 1$ must be valid. In other words, $\mathcal{F}$ cannot be robust at x_1 or x_s for the robust budget 0.4. Similarly, $\mathcal{F}$ cannot be robust for at least one point in $\{x_i\}_{i=k(n+1)+1}^{(k+1)(n+1)}$ for $k \in \{1, \ldots, \frac{N}{n+1} - 1\}$. In summary, $\mathcal{F}$ cannot be robust for at least $\frac{N}{n+1}$ points, so $\text{RA}_{\mathcal{D}}(\mathcal{F}, 0.4) \leq 1 - \frac{1}{n+1}$. □

A.3. Proof of Theorem 3.8

We restate Theorem 3.8 for convenience.

Theorem A.5. *For any dataset $\mathcal{D} \in \mathcal{D}_{n,N,L}$, the hypothesis space $\mathbf{H}_{n,2N+1,3n+1,O(Nn)}$ is an optimal robust memorization for $\mathcal{D}$.*

Proof. It suffices to show that for any $\mu < 0.5\lambda_{\mathcal{D}}$, there exists a network with depth $2N + 1$, width $3n + 1$, and $O(Nn)$ nonzero parameters, which can robustly memorize $\mathcal{D}$ with robust budget μ.

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subset \mathbb{R}^n \times [L]$. Let $C \in \mathbb{R}_+$ satisfy $C > |x_i^{(j)}| + \mu > 0$ for all $i \in [N]$ and $j \in [n]$.

$\mathcal{F}$ will be defined in three steps for an input x.

Step 1. The first layer is used to check whether $x \in \mathbb{B}(x_1, \mu)$. The second layer is used to compute $E_1(x)$ in Property 2 given below. The two layers are given below.

(1-1.1) $\mathcal{F}_1^0(x) = 0$;

(1-1.2) $\mathcal{F}_1^j(x) = \sigma(x_1^{(j)} - x^{(j)} - \mu)$, $\mathcal{F}_1^{n+j}(x) = \sigma(x^{(j)} - x_1^{(j)} - \mu)$, where $j \in [n]$;

(1-1.3) $\mathcal{F}_1^{2n+j}(x) = \sigma(x^{(j)} + C)$, where $j \in [n]$;

(1-2.1) $\mathcal{F}_2^0(x) = 0$;

(1-2.2) $\mathcal{F}_2^1(x) = \sigma(y_1 - \frac{y_1}{\lambda_{\mathcal{D}} - 2\mu} \sum_{k=1}^{2n} \mathcal{F}_1^k(x))$;

(1-2.3) $\mathcal{F}_2^{j+1}(x) = \sigma(\mathcal{F}_1^{2n+j}(x))$, where $j \in [n]$.

Step 2. For $i = 2, 3, \ldots, N$, the $(2i-1)$-th layer has width $3n + 1$ and is used to check whether $x \in \mathbb{B}(x_i, \mu)$. The $2i$ layer has width $n + 2$ and is used to compute $E_i(x)$ in Property 2 given below. These layers are given below.

(i-1.1) $\mathcal{F}_{2i-1}^0(x) = \sigma(\mathcal{F}_{2i-2}^0(x) + \mathcal{F}_{2i-2}^1(x))$;

(i-1.2) $\mathcal{F}_{2i-1}^j(x) = \sigma((x_i^{(j)} + C) - \mathcal{F}_{2i-2}^{j+1}(x) - \mu)$ and $\mathcal{F}_{2i-1}^{n+j}(x) = \sigma(\mathcal{F}_{2i-2}^{j+1}(x) - (x_i^{(j)} + C) - \mu)$, where $j \in [n]$;

(i-1.3) $\mathcal{F}_{2i-1}^{2n+j}(x) = \sigma(\mathcal{F}_{2i-2}^{j+1}(x))$, where $j \in [n]$;

(i-2.1) $\mathcal{F}_{2i}^0(x) = \sigma(\mathcal{F}_{2i-1}^0(x))$;

(i-2.2) $\mathcal{F}_{2i}^1(x) = \sigma(y_i - \frac{y_i}{\lambda_{\mathcal{D}} - 2\mu} \sum_{k=1}^{2n} \mathcal{F}_{2i-1}^k(x) - \mathcal{F}_{2i-1}^0(x))$;

(i-2.3) $\mathcal{F}_{2i}^{j+1}(x) = \sigma(\mathcal{F}_{2i-1}^{2n+j}(x))$, where $j \in [n]$.

Step 3. The output layer of $\mathcal{F}$ is $\mathcal{F}(x) = \mathcal{F}_{2N}^0(x) + \mathcal{F}_{2N}^1(x)$.

Next, we will show that $\mathcal{F}$ has the following properties.

Property 1. $\mathcal{F}_{2i}^{j+1}(x) = x^{(j)} + C$ for $i \in [N]$, $j \in [n]$, and $x \in \mathbb{R}^n$.

From (1-1.3) and (1-2.3), since $C + x_i^{(j)} > \mu > 0$ for all $i \in [N]$ and $j \in [n]$, we have that $\mathcal{F}_2^{j+1}(x) = \mathcal{F}_1^{2n+j}(x) = \sigma(x^j + C) = x^j + C$.

From (i-2.3) and (i-1.3), we have that $\mathcal{F}_{2i}^{j+1}(x) = \sigma(\mathcal{F}_{2i-1}^{2n+j}(x)) = \sigma(\mathcal{F}_{2i-2}^{j+1}(x)) = \cdots = \sigma(\mathcal{F}_2^{j+1}(x)) = x^{(j)} + C$, for all $i \in [N]$ and $j \in [n]$. Property 1 is proved.

Property 2. Let $E_i(x) = y_i - \frac{y_i}{\lambda_{\mathcal{D}}-2\mu}\sum_{j=1}^{2n}\mathcal{F}_{2i-1}^j(x)$ for $i \in [N]$. Then $E_i(x) = y_i$ for $x \in \mathbb{B}_\infty(x_i, \mu)$, and $E_i(x) < y_i$ for $x \notin \mathbb{B}_\infty(x_i, \mu)$.

Due to Property 1, for $j \in [n]$, step (i-1.2) becomes

$$\begin{aligned}&\mathcal{F}_{2i-1}^j(x) = \sigma((x_i^{(j)} + C) - \mathcal{F}_{2i-2}^{j+1}(x) - \mu)\\ &= \sigma(x_i^{(j)} - x^{(j)} - \mu)\\ &\mathcal{F}_{2i-1}^{n+j}(x) = \sigma(\mathcal{F}_{2i-2}^{j+1}(x) - (x_i^{(j)} + C) - \mu)\\ &= \sigma(x^{(j)} - x_i^{(j)} - \mu).\end{aligned}$$

If $x \in \mathbb{B}_\infty(x_i, \mu)$, then $\sigma(x_i - x - \mu) = \sigma(x - x_i - \mu) = 0$, which means $\mathcal{F}_{2i-1}^j(x) = 0$ for $j \in [2n]$. Thus $E_i(x) = y_i$. If $x \notin \mathbb{B}_\infty(x_i, \mu)$, then $||x_i - x - \mu||_\infty > 0$ or $||x - x_i - \mu||_\infty > 0$ which means that $\mathcal{F}_{2i-1}^j(x) > 0$ for at least one $j \in [2n]$. Since $\mathcal{F}_i^j(x) \geq 0$ for all i and j, we have that $E_i(x) < y_i$.

Property 3. If $x \in \mathbb{B}_\infty(x_k, \mu)$ for $y_k \neq y_i$, then $E_i(x) \leq 0$.

Since $x \in \mathbb{B}_\infty(x_k, \mu)$ and $y_k \neq y_i$, we have that $||x_i - x - \mu||_\infty \geq \lambda_{\mathcal{D}} - 2\mu > 0$ or $||x - x_i - \mu||_\infty \geq \lambda_{\mathcal{D}} - 2\mu > 0$, since the separation bound is λ. Then $\mathcal{F}_{2i-1}^j(x) \geq \lambda_{\mathcal{D}} - 2\mu$ for at least one $j \in [2n]$ and thus $E_i(x) \leq y_i - \frac{y_i}{\lambda_{\mathcal{D}}-2\mu}(\lambda_{\mathcal{D}} - 2\mu) = 0$.

Property 4. $\mathcal{F}(x) = \max_{i\in[N]}\{E_i(x), 0\}$ for $x \in \mathbb{R}^n$.

Since $\max\{x, y\} = x + \sigma(y - x)$ for $x, y \in \mathbb{R}$ and $\mathcal{F}_i^j(x) \geq 0$ for all i and j, we have that

$$\begin{aligned}\sigma(\mathcal{F}_{2i}^0(x) + \mathcal{F}_{2i}^1(x)) &= \mathcal{F}_{2i}^0(x) + \mathcal{F}_{2i}^1(x)\\ &= \sigma(\mathcal{F}_{2i-1}^0(x)) + \sigma(E_{i-1}(x) - \mathcal{F}_{2i-1}^0(x))\\ &= \max\{\mathcal{F}_{2i-1}^0(x), E_i(x)\}\\ &= \max\{\sigma(\mathcal{F}_{2i-2}^0(x) + \mathcal{F}_{2i-2}^1(x)), E_i(x)\}.\end{aligned}$$

Using the above equation repeatedly, we have that $\mathcal{F}(x) = \sigma(\mathcal{F}_{2N}^0(x) + \mathcal{F}_{2N}^1(x)) = \max_{i=1}^N\{E_i(x), \mathcal{F}_2^0(x)\} = \max_{i=1}^N\{E_i(x), 0\}$.

We now show that $\mathcal{F}$ satisfies the conditions of the theorem. Let $x \in \mathbb{B}_\infty(x_s, \mu)$ for $s \in [N]$. By Property 2, $E_s(x) = y_s$; and if $i \neq s$ and $y_i = y_s$, then $E_i(x) < y_s$. By Property 3, if $y_i \neq y_s$, then $E_i(x) \leq 0$. By Property 4, $\mathcal{F}(x) = \max_{i\in[N]}\{E_i(x)\} = E_s(x) = y_s$; that is, $\mathcal{F}$ is robust at x_s with budget μ.

We now estimate the number of nonzero parameters. For $i \in [N]$, constructions (i-1.1) and (i-2.1) need 3 parameters; (i-1.2) needs $8n$ parameters; (i-1.3) and (i-2.3) need $2n$ parameters; (i-2.2) need $2n + 2$ parameters. Totally, $(N - 1)(12n + 5) + 2$ parameters are needed. □

A.4. Proofs for Theorem 3.10

We give a lemma below.

Lemma A.6. *There exists a network $\mathcal{F} \in \mathbf{H}_{n,O(\log n),O(n),O(n)}$ such that $\mathcal{F}(x) = ||x||_\infty$; that is, there exists a network $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}$ with depth $O(\log n)$, width $O(n)$, and $O(n)$ nonzero parameters such that $\mathcal{F}(x) = ||x||_\infty$.*

Proof. Let $e = \lceil \log_2 n \rceil$. Without loss of generality, we assume that $n = 2^e$. Then $\mathcal{F}$ has depth $2e$ and for $i \in [e + 1]$, the $(2i - 1)$-th layer has width 2^{e-i+2}, and the $2i$-th layer has width 2^{e-i+1}.

Denote W_i and b_i to be the weight matrix and the bias of the i-th layer of $\mathcal{F}$. The first and second layers will change x to $|x|$. The first layer has width 2^{e+1} and the second layer has width 2^e, which are defined below.

$W_1^{2i,i} = 1$ and $W_1^{2i+1,i} = -1$; other entries of W_1 are 0. $b_1 = 0$.

$W_2^{i,2i} = 1$ and $W_2^{i,2i+1} = 1$; other entries of W_2 are 0. $b_2 = 0$.

Since $\sigma(x) + \sigma(-x) = |x|$ for any $x \in \mathbb{R}$, it is easy to check that $\mathcal{F}_2(x) = \sigma(W_2\sigma(W_1x)) = |x|$.

For $i \in [e]$, the $(2i+1)$-th and the $(2i+2)$-th layers are defined below.

$\mathcal{F}_{2i+1}^{2m}(x) = \sigma(\mathcal{F}_{2i}^{2m}(x))$, where $m = 0, 1, \ldots, 2^{e-i} - 1$.

$\mathcal{F}_{2i+1}^{2m+1}(x) = \sigma(\mathcal{F}_{2i}^{2m+1}(x) - \mathcal{F}_{2i}^{2m}(x))$, where $m = 0, 1, \ldots, 2^{e-i} - 1$.

$\mathcal{F}_{2i+2}^{m}(x) = \sigma(\mathcal{F}_{2i+1}^{2m}(x) + \mathcal{F}_{2i+1}^{2m+1}(x))$, where $m = 0, 1, \ldots, 2^{e-i} - 1$.

For $i \in [e+1]$, using $\sigma(x-y)+y = \max\{x, y\}$ for any $x, y \in \mathbb{R}$, we have that

$$\begin{aligned} & \mathcal{F}_{2i+2}^{m}(x) \\ = \; & \sigma(\mathcal{F}_{2i+1}^{2m}(x) + \mathcal{F}_{2i+1}^{2m+1}(x)) \\ = \; & \mathcal{F}_{2i+1}^{2m}(x) + \mathcal{F}_{2i+1}^{2m+1}(x) \\ = \; & \sigma(\mathcal{F}_{2i}^{2m}(x)) + \sigma(\mathcal{F}_{2i}^{2m+1}(x) - \mathcal{F}_{2i}^{2m}(x)) \\ = \; & \mathcal{F}_{2i}^{2m}(x) + \sigma(\mathcal{F}_{2i}^{2m+1}(x) - \mathcal{F}_{2i}^{2m}(x)) \\ = \; & \max\{\mathcal{F}_{2i}^{2m}(x), \mathcal{F}_{2i}^{2m+1}(x)\}. \end{aligned}$$

The $(2e+2)$-th layer has width 1 and is the output

$$\begin{aligned} \mathcal{F}(x) &= \mathcal{F}_{2e+2}^{1}(x) \\ &= \max\{\mathcal{F}_{2e}^{2}(x), \mathcal{F}_{2e}^{1}(x)\} \\ &= \max\{\mathcal{F}_{2e-2}^{4}(x), \mathcal{F}_{2e-2}^{3}(x), \mathcal{F}_{2e-2}^{2}(x), , \mathcal{F}_{2e-2}^{1}(x)\} \\ &= \ldots \\ &= \max\{\mathcal{F}_{2}^{2^e}(x), \mathcal{F}_{2}^{2^e-1}(x), \ldots, \mathcal{F}_{2}^{2}(x), , \mathcal{F}_{2}^{1}(x)\} \\ &= ||x||_\infty. \end{aligned}$$

We now estimate the number of parameters. The first two layers need $4d$ nonzero parameters. For $i \in [e]$, the $(2i+1)$-th layer and $(2i+2)$-th layer need $5 \cdot 2^{e-i}$ parameters. So, we need $\sum_{i=1}^{e} 5 \cdot 2^{e-i} = O(2^e) = O(n)$ parameters. Then the lemma is proved. □

We restate the theorem for convenience.

Theorem A.7. *For any dataset $\mathcal{D} \in \mathcal{D}_{n,N,2}$, the hypothesis space $\mathbf{H}_{n,O(N\log(n)),O(n),O(Nn\log(n))}$ contains a network $\mathcal{F}$ which is an optimal robust memorization of $\mathcal{D}$ via Lipschitz; that is, $\mathcal{F}$ is a memorization of $\mathcal{D}$ and $\mathrm{Lip}_\infty(\mathcal{F}) = 1/\lambda_{\mathcal{D}}$.*

Proof. Let $\mathcal{D} = \{(x_i, l_{x_i})\}_{i=1}^{N} \in \mathcal{D}_{n,N,2}$ and $C \in \mathbb{R}_+$ satisfy $C + x_i^{(k)} - 0.5\lambda_{\mathcal{D}} > 0$ for all $i \in [N]$, $k \in [n]$. Let $y_i = 2(l_{x_i} - 1.5)$ for any $i \in [N]$, that is $y_i = -1$ if $l_{x_i} = 1$ and $y_i = 1$ if $l_{x_i} = 2$. The network has $N(2\lceil\log(n)\rceil + 5) + 1$ hidden layers which will be defined below.

Step 1. The first layer has width $n+1$: $\mathcal{F}_1^0(x) = 2$ and $\mathcal{F}_1^j(x) = \sigma(x^{(j)} + C) = x^{(j)} + C$, where $j \in [n]$.

Step 2. Let $s_k = (2\lceil\log(n)\rceil + 5)(k-1) + 2$. For $k \in [N]$, we will use the s_k-th layer to the $(s_k + 2\lceil\log(n)\rceil + 4)$-th layer to check if $||x - x_k||_\infty < 0.5\lambda_{\mathcal{D}}$. Step 2 consists of three sub-steps.

Step 2a. We use the s_k-th layer and the (s_k+1)-th layer to calculate $|x - x_k|$. The s_k-th layer has width $3n+1$ and is defined below.

$\mathcal{F}_{s_k}^0(x) = \sigma(\mathcal{F}_{s_k-1}^0(x))$;

$\mathcal{F}_{s_k}^j(x) = \sigma(\mathcal{F}_{s_k-1}^j(x) - x_k^{(j)} - C)$, where $j \in [n]$;

$\mathcal{F}_{s_k}^{n+j}(x) = \sigma(-\mathcal{F}_{s_k-1}^j(x) + x_k^{(j)} + C)$, where $j \in [n]$;

$\mathcal{F}_{s_k}^{2n+j}(x) = \sigma(\mathcal{F}_{s_k-1}^j(x))$, where $j \in [n]$.

The (s_k+1)-th layer has width $2n+1$ and is defined below.

$$\mathcal{F}^0_{s_k+1}(x) = \sigma(\mathcal{F}^0_{s_k}(x));$$

$$\mathcal{F}^j_{s_k+1}(x) = \sigma(\mathcal{F}^j_{s_k}(x) + \mathcal{F}^{n+j}_{s_k}(x)), \text{ where } j \in [n];$$

$$\mathcal{F}^{n+j}_{s_k+1}(x) = \sigma(\mathcal{F}^{2n+j}_{s_k}(x)), \text{ where } j \in [n].$$

The s_k-th layer needs $5n+1$ nonzeros parameters and (s_k+1)-th layer needs $3n+1$ nonzeros parameters.

Step 2b. Lemma A.6 is used to calculate $||x - x_k||_\infty$. By Lemma A.6, there exists a network $\mathcal{H}: \mathbb{R}^n \to \mathbb{R}$ with $2\lceil\log(n)\rceil$ hidden layers, width $O(n)$, and $O(n)$ nonzero parameters to compute $\mathcal{H}(x) = ||x||_\infty$ for $x \in \mathbb{R}^n$. Since $\mathcal{H}$ has $2\lceil\log(n)\rceil$ hidden layers, we set the output of the $(s_k + 2\lceil\log(n)\rceil + 1)$-th layer to be

$$\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x) = \sigma(\mathcal{F}^0_{s_k+1}(x));$$

$$\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x) = \mathcal{H}(\mathcal{F}^1_{s_k+1}(x), \ldots, \mathcal{F}^n_{s_k+1}(x)) = ||\mathcal{F}_{s_k+1}(x)||_\infty;$$

$$\mathcal{F}^{j+1}_{s_k+2\lceil\log(n)\rceil+1}(x) = \sigma(\mathcal{F}^{n+j}_{s_k+1}(x)), \text{ where } j \in [n].$$

Step 2c. Use the $(s_k + 2\lceil\log(n)\rceil + 2)$-th to the $(s_k + 2\lceil\log(n)\rceil + 4)$-th layers to check if $||x - x_k||_\infty < 0.5\lambda_{\mathcal{D}}$. The $(s_k + 2\lceil\log(n)\rceil + 2)$-th layer has width $n+4$ and is defined below

$$\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x));$$

$$\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x) + 1);$$

$$\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x) - 2);$$

$$\mathcal{F}^3_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x) + 2);$$

$$\mathcal{F}^{j+3}_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(\mathcal{F}^{j+1}_{s_k+2\lceil\log(n)\rceil+1}(x)), \text{ where } j \in [n].$$

The $(s_k + 2\lceil\log(n)\rceil + 3)$-th layer has width $n+3$ and is defined below

$$\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) + y_k\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x));$$

$$\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2});$$

$$\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2} - (\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+2}(x) + \mathcal{F}^3_{s_k+2\lceil\log(n)\rceil+2}(x)));$$

$$\mathcal{F}^{j+2}_{s_k+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^{j+3}_{s_k+2\lceil\log(n)\rceil+2}(x)), \text{ where } j \in [n].$$

The $(s_k + 2\lceil\log(n)\rceil + 4)$-th layer has width $n+1$ and is defined as

$$\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) = \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+3}(x) - y_k(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+3}(x) - \mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+3}(x)));$$

$$\mathcal{F}^j_{s_k+2\lceil\log(n)\rceil+4}(x) = \sigma(\mathcal{F}^{j+2}_{s_k+2\lceil\log(n)\rceil+3}(x)), \text{ where } j \in [n].$$

It is easy to check that if $\mathcal{F}^j_{s_k+1}(x) = |x^{(j)} - x_k^{(j)}|$. Then

$$\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x) + 1) > 0$$

if and only if $||x - x_k||_\infty < 0.5\lambda_{\mathcal{D}}$. These three layers need $3n+16$ nonzeros parameters.

Step 3. The output is $\mathcal{F}(x) = 0.5(\mathcal{F}^0_{s_N+2\lceil\log(n)\rceil+4}(x) - 2) + 1.5$. The network $\mathcal{F}$ has width $O(n)$, depth $O(N\log(n))$, and $O(Nn\log(n))$ nonzeros parameters.

We now show that $\mathcal{F}$ satisfies the condition of the theorem; that is $\mathcal{F}$ memorizes $\mathcal{D}$ and satisfies $\text{Lip}_\infty(\mathcal{F}) = 2/\lambda_{\mathcal{D}}$.

Property 1. $\mathcal{F}^j_{s_k-1}(x) = x^{(j)} + C$ for $j \in [n]$ and $k \in [N]$. When $k=1$, $s_k - 1 = 1$. By Step 1, we have that

$\mathcal{F}^j_{s_1-1}(x) = \mathcal{F}^j_1(x) = x^{(j)} + C$. When $k > 1$, we have that

$$\begin{aligned} & \mathcal{F}^j_{s_{k+1}-1}(x) \\ & = \sigma(\mathcal{F}^j_{s_k+2\lceil\log(n)\rceil+4}(x)) = \sigma(\mathcal{F}^{j+2}_{s_k+2\lceil\log(n)\rceil+3}(x)) \\ & = \sigma(\mathcal{F}^{j+3}_{s_k+2\lceil\log(n)\rceil+2}(x)) = \sigma(\mathcal{F}^{j+1}_{s_k+2\lceil\log(n)\rceil+1}(x)) \\ & = \sigma(\mathcal{F}^{n+j}_{s_k+1}(x)) = \sigma(\mathcal{F}^{2n+j}_{s_k}(x)) = \sigma(\mathcal{F}^j_{s_k-1}(x)) \\ = \quad & \mathcal{F}^j_{s_k-1}(x). \end{aligned}$$

Then, $\mathcal{F}^j_{s_{k+1}-1}(x) = \mathcal{F}^j_{s_k-1}(x) = \cdots = \mathcal{F}^j_{s_1-1}(x) = \mathcal{F}^j_1(x) = x^{(j)} + C$.

Property 2. $\mathcal{F}^j_{s_k+1}(x) = |x^{(j)} - x^{(j)}_k|$ and $\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x) = ||x - x_k||_\infty$ for $j \in [n]$.

Since $\sigma(x) + \sigma(-x) = |x|$ for any $x \in \mathbb{R}$, from Step 2a, $\mathcal{F}^j_{s_k+1}(x) = |\mathcal{F}^j_{s_k-1}(x) - x^{(j)}_k - C|$ for $j \in [n]$. By Property 1, $\mathcal{F}^j_{s_k-1}(x) = x^{(j)} + C$ for $j \in [n]$. Then, $\mathcal{F}^j_{s_k+1}(x) = |x^{(j)} - x^{(j)}_k|$ for $j \in [n]$. From Step 2b, we have that $\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x) = ||x - x_k||_\infty$ for $j \in [n]$.

Property 3. $\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) = 2 + y_{w_k}\sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_k}||_\infty)$, where $w_k = \text{argmin}_{i\in[k]}||x - x_i||_\infty$.

We prove the property by induction on k. We first show that the statement is valid for $k = 1$. We have that $w_k = 1$ and $\mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+2}(x) = \mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+1}(x) = \mathcal{F}^0_{s_1+1}(x) = \mathcal{F}^0_{s_1}(x) = \mathcal{F}^0_{s_1-1}(x) = 2$. From Step 2c and Property 2,

$$\begin{aligned} & \mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+3}(x) \\ = \quad & \sigma(\mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+2}(x) + y_0\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+2}(x)) \\ = \quad & \sigma(2 + y_0\sigma(1 - 2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+1}(x))) \\ = \quad & 2 + y_0\sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+1}(x) + 1) \\ = \quad & 2 + y_0\sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_0||_\infty). \end{aligned}$$

Since $\mathcal{F}^2_{s_1+2\lceil\log(n)\rceil+2}(x) = \sigma(\mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+1}(x) - 2) = \sigma(2 - 2) = 0$ and $\mathcal{F}^3_{s_1+2\lceil\log(n)\rceil+2}(x) = \sigma(-\mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+1}(x) + 2) = \sigma(2 - 2) = 0$, we have that $\mathcal{F}^2_{s_1+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+2} - (\mathcal{F}^2_{s_1+2\lceil\log(n)\rceil+2}(x) + \mathcal{F}^3_{s_1+2\lceil\log(n)\rceil+2}(x))) = \sigma(\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+2}) = \mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+3}$. Then

$$\begin{aligned} & \mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+4}(x) \\ = \quad & \sigma(\mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+3}(x) - y_0(\mathcal{F}^1_{s_1+2\lceil\log(n)\rceil+3}(x) \\ & -\mathcal{F}^2_{s_1+2\lceil\log(n)\rceil+3}(x))) \\ = \quad & \mathcal{F}^0_{s_1+2\lceil\log(n)\rceil+3}(x) \\ = \quad & 2 + y_0\sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_0||_\infty). \end{aligned}$$

We have proved the statement for $k = 1$.

Assume that the statement is valid for $k - 1$; that is, $\mathcal{F}^0_{s_{k-1}+2\lceil\log(n)\rceil+4}(x) = 2 + y_{w_{k-1}}\sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty)$. We have that $\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) = \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x) = \mathcal{F}^0_{s_k+1}(x) = \mathcal{F}^0_{s_k}(x) = \mathcal{F}^0_{s_k-1}(x) = 2 + y_{w_{k-1}}\sigma(1 - 2/\lambda_{\mathcal{D}}||x -$

$x_{w_{k-1}}||_\infty) \geq 1$, and we also have that $\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x)+1) \leq 1$. Then

$$\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+3}(x) \\
= \;& \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) + y_k\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x)) \\
= \;& \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& +y_k\sigma(1-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x))) \\
= \;& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& +y_k\sigma(1-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x)) \\
= \;& \mathcal{F}^0_{s_k-1}(x) + y_k\sigma(1-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x)) \\
= \;& \mathcal{F}^0_{s_k-1}(x) + y_k\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x)
\end{aligned} \tag{35}$$

Since $\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x)-2)$ and $\mathcal{F}^3_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x)+2)$, we have that $\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+3}(x) = \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2} - (\mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+2}(x) + \mathcal{F}^3_{s_k+2\lceil\log(n)\rceil+2}(x))) = \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2} - |\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+1}(x)-2|)$. Then

$$\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) \\
= \;& \sigma(\mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+3}(x) - y_k(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+3}(x) - \mathcal{F}^2_{s_k+2\lceil\log(n)\rceil+3}(x))) \\
= \;& \sigma(\mathcal{F}^0_{s_k-1}(x) + y_k\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) - y_k(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& -\sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) - |\mathcal{F}^0_{s_k-1}(x)-2|))).
\end{aligned}$$

We divide the proof into two cases.

Case 1. If $x \notin \mathbb{B}_\infty(x_k, 0.5\lambda_{\mathcal{D}})$, then $\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x)+1) = \sigma(1-2/\lambda_{\mathcal{D}}||x - x_k||_\infty) = 0$ and

$$\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) \\
= \;& \sigma(\mathcal{F}^0_{s_k-1}(x) + y_k\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& -y_k(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x)- \\
& \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) - |\mathcal{F}^0_{s_k-1}(x)-2|))) \\
= \;& \mathcal{F}^0_{s_k-1}(x) \\
= \;& \mathcal{F}^0_{s_{k-1}+2\lceil\log(n)\rceil+4}(x) \\
= \;& 2 + y_{w_{k-1}}\sigma(1-2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty) \\
= \;& 2 + y_{w_k}\sigma(1-2/\lambda_{\mathcal{D}}||x - x_{w_k}||_\infty).
\end{aligned}$$

Case 2. If $x \in \mathbb{B}_\infty(x_k, 0.5\lambda_{\mathcal{D}})$, then $\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) = \sigma(-2/\lambda_{\mathcal{D}}\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+1}(x)+1) = \sigma(1-2/\lambda_{\mathcal{D}}||x - x_k||_\infty) \geq 0$ and using equation 35:

$$
\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) \\
= \;& \sigma(\mathcal{F}^0_{s_k-1}(x) + y_k \mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& -y_k(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) - \sigma(\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& -|\mathcal{F}^0_{s_k-1}(x) - 2|))) \\
= \;& \sigma(\mathcal{F}^0_{s_k-1}(x) + y_k \mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x) \\
& -y_k(\min\{\mathcal{F}^1_{s_k+2\lceil\log(n)\rceil+2}(x), |2 - \mathcal{F}^0_{s_k-1}(x)|\})) \\
= \;& \sigma(2 + y_{w_{k-1}} \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty) + \\
& y_k(1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty) \\
& -y_k(\min\{1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty, \\
& \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty)\})).
\end{aligned}
$$

Consider two sub-cases:

Case 2.1. If $||x - x_{w_{k-1}}||_\infty > 0.5\lambda_{\mathcal{D}}$, then $w_k = k$ and hence

$$
\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) \\
= \;& \sigma(2 + y_{w_{k-1}} \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty) + \\
& y_k(1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty) \\
& -y_k(\min\{1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty, \\
& \quad \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty)\})) \\
= \;& \sigma(2 + y_k(1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty)) \\
= \;& 2 + y_k(1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty) \\
= \;& 2 + y_{w_k}(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_k}||_\infty).
\end{aligned}
$$

Case 2.2. If $||x - x_{w_{k-1}}||_\infty \leq 0.5\lambda_{\mathcal{D}}$, then $y_{w_{k-1}} = y_k$ and hence

$$
\begin{aligned}
& \mathcal{F}^0_{s_k+2\lceil\log(n)\rceil+4}(x) \\
= \;& \sigma(2 + y_{w_{k-1}} \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty) + \\
& y_k(1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty) \\
& -y_k(\min\{1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty, \\
& \quad \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty)\})) \\
= \;& \sigma(2 + y_{w_{k-1}}(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty) \\
& \quad + y_k(1 - 2/\lambda||x - x_k||_\infty) \\
& -y_k(\min\{1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty, \\
& \quad 1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty\})) \\
= \;& 2 + y_k \max\{1 - 2/\lambda_{\mathcal{D}}||x - x_k||_\infty, \\
& 1 - 2/\lambda_{\mathcal{D}}||x - x_{w_{k-1}}||_\infty\} \\
= \;& 2 + y_{w_k} \sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_k}||_\infty).
\end{aligned}
$$

The property is proved.

Property 4. $\mathcal{F}$ is a memorization $\mathcal{D}$ and has $\text{Lip}_\infty(\mathcal{F}) = 1/\lambda_{\mathcal{D}}$.

By Property 3, the output is

$$\mathcal{F}(x) = 0.5(\mathcal{F}^1_{s_N+2\lceil\log(n)\rceil+4}(x) - 2) + 1.5 = 0.5y_{w_N}\sigma(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_N}||_\infty) + 1.5$$

where $w_N = \text{argmin}_{i\in[N]}||x - x_i||_\infty$.

If $x = x_s$, then $w_N = s$ and $\mathcal{F}(x) = 0.5y_s + 1.5 = l_{x_s}$; that is, $\mathcal{F}$ memorizes $\mathcal{D}$. If $x \in \mathbb{B}(x_s, 0.5\lambda_{\mathcal{D}})$ for some $s \in [N]$, then $w_N \in [N]$ and $\mathcal{F}(x) = 0.5y_{w_N}(1 - 2/\lambda_{\mathcal{D}}||x - x_{w_N}||_\infty) + 1.5$ such that the local $\text{Lip}_\infty(\mathcal{F}) = 1/\lambda_{\mathcal{D}}$ over $\mathbb{B}(x_{w_N}, 0.5\lambda_{\mathcal{D}})$. If x is not in $\cup_{i=1}^N \mathbb{B}(x_s, 0.5\lambda_{\mathcal{D}})$, then $||x - x_{w_N}||_\infty > 0.5\lambda_{\mathcal{D}}$. Hence $\mathcal{F}(x) = 0$ and the local $\text{Lip}_\infty(\mathcal{F}) = 0$. The theorem is proved. □