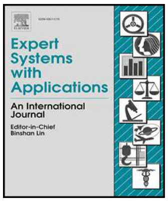
Contents lists available at ScienceDirect

Expert Systems With Applications

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



Training generalizable quantized deep neural nets

Charles Hernandez[1], Bijan Taslimi [*], Hung Yi Lee, Hongcheng Liu[2], Panos M. Pardalos

Department of Industrial and Systems Engineering, University of Florida, Gainesville FL, 32611, USA

ARTICLE INFO

MSC:
62M45
68Q32

Keywords:
Deep learning
Quantized neural networks
Generalizability

ABSTRACT

While a number of practical methods for training quantized DL models have been presented in the literature, there exists a critical gap in the theoretical generalizability results for such approaches. Although empirical evidence often suggests a high tolerance of DL architectures to variations of training procedures, existing theoretical generalization analyses are often contingent on the specific designs of training algorithms, e.g., in stochastic gradient descent (SGD). This specialization makes such generalizability results inapplicable to the case of quantized DL models. In view of this critical vacuum, this paper provides several almost-algorithm-independent results to ensure the generalizability of a quantized neural network at different levels of optimality. These results include the characterizations of a computable, quantized local solution that ensures the generalization performance and an algorithm that is provably convergent to such a local solution.

1. Introduction

Deep Learning methods have been successfully used in a wide spectrum of applications, achieving impressive results (Affonso, Rossi, Vieira, de Leon Ferreira, et al., 2017; Ahn, Cho, & Kim, 2000; Dosovitskiy et al., 2020; Edunov, Ott, Auli, & Grangier, 2018; Foret, Kleiner, Mobahi, & Neyshabur, 2020; Guresen, Kayakutlu, & Daim, 2011; Kabir, Abdar, Jalali, Khosravi, Atiya, Nahavandi, et al., 2020; Liu, Duh, Liu, & Gao, 2020; Tao, Sapra, & Catanzaro, 2020; Tsai & Wu, 2008; Zhang et al., 2020). However, the resulting models can be prohibitively expensive to use in application areas where computational resources are limited. Recent technologies like facial recognition (Zhao & Tsai, 2015), OCR (Laine & Nevalainen, 2006) and text translation (Fragoso, Gauglitz, Zamora, Kleban, & Turk, 2011) are rapidly being adopted as core features on many mobile devices, a trend that has only emphasized the need for technologies that can transform these state-of-the-art models into something that can be more portably utilized.

Numerous works have shown how to turn this desire into a reality, which we will discuss in more detail in Section 1.1, but for the purposes of this paper, we focus on methods involving quantized networks. A quantized network is one in which all parameter values have been discretized in order to significantly reduce the number of candidate weight values admissible to the network. As an example, rounding all the weights of a network to integers would be considered a network that has been quantized to integer values. A more rigorous definition of quantization can be found at the end of Section 1.2.

Though there have been many works establishing practical schemes for training quantized DL models, there remain significant gaps in our theoretical understanding of such problems. Two of these critical vacua are as below:

- *Gap 1.* While Petersen and Voigtlaender (2018) and Ding, Liu, Xiong, and Shi (2018) have proven the expressive power of quantized networks and Bu, Gao, Zou, and Veeravalli (2019) have analyzed the impact of model compression on generalization performance in general, limited theoretical generalizability guarantees on quantized DL models are available from the literature.
- *Gap 2.* Although, for non-quantized DL models, there are many existing results on the generalization error bounds, they tend to be contingent on the specific algorithmic designs, e.g., of the stochastic gradient descent (SGD) (Brutzkus, Globerson, Malach, & Shalev-Shwartz, 2017; Brutzkus et al., 2017; Cao & Gu, 2019a, 2019b; Daniely, 2017; Li & Liang, 2018; Wang, Giannakis, & Chen, 2019). Because most provably generalizability-guaranteeing algorithms cannot be directly used for quantization, it is unknown how the corresponding algorithm-specific generalizability results can be readily applicable to a quantized DL model.

In view of these critical gaps, this paper presents several results on the generalization performance of a quantized DL model. We first

[*] Corresponding author.
E-mail addresses: cdhernandez@ufl.edu (C. Hernandez), b.taslimi@ufl.edu (B. Taslimi), hungyilee@ufl.edu (H.Y. Lee), hliu@ise.ufl.edu (H. Liu), pardalos@ise.ufl.edu (P.M. Pardalos).

[1] Supported by National Science Foundation, USA grant CMMI-2016571.

[2] Supported by National Science Foundation, USA grant CMMI-2016571 and University of Florida, USA AI Catalyst Grant OR-DRD-AI2020.

https://doi.org/10.1016/j.eswa.2022.118736
Received 23 January 2021; Received in revised form 9 July 2022; Accepted 29 August 2022
Available online 5 October 2022

present a completely algorithm-independent generalization error bound for globally optimal solutions to the training formulation of a quantized DL. To our knowledge, this is among the first set of theoretical results that explicate the impact of quantization to generalization performance.

Further, we present an almost-algorithm-independent result for the locally trained quantized DL models. We show the existence of tractable local solutions where a significant portion of connection weights of a DL model can be quantized without introducing any compromise to the generalizability. Meanwhile, if all weights are quantized, then the generalization error can still be well controlled. As an additional contribution to Gap 2. , a byproduct of our local result can be found in Section 2.4. Such a result is readily applicable to analyzing a non-quantized DL model.

Based on our new theoretical insights, we further propose a quantized iterative shrinkage-thresholding algorithm (Quantized ISTA), which is a first-order method with provably tractable computational complexity in solving for those generalizable, quantized local solutions. Our algorithm substantially modifies the ISTA traditionally for sparse recovery, as discussed by, e.g., Beck and Teboulle (2009), and is a new addition to existing quantized training algorithms as further explored in Section 1.1. We finally demonstrate this algorithm's performance in numerical experiments including both synthetic and real world data, thereby testing our theoretical results and the practical performance of our approach.

The rest of this paper is organized as below: We conclude our introduction with a review of Network Compression Literature and an explanation of our notations in Sections 1.1 and 1.2, respectively. Section 2 presents our main theoretical results. Section 3 briefly discusses our numerical experiments to verify our theory (with details provided in the supplementary document). Section 4 concludes this paper. All our proofs can be found in Appendix.

1.1. Related works on quantized DL

A variety of methods have been used to convert a trained DL model into smaller or more computationally efficient forms. Han, Mao, and Dally (2015) demonstrated the effectiveness of a combination of pruning and weight clustering on compressing a large network like the AlexNet into a model with 1/40th of the original size, without significant accuracy loss. Others have focused on designing entirely new architectures from the ground up to optimize the tradeoff between model size and accuracy. Such an approach has been demonstrated by Howard et al. (2017), Zhang, Zhou, Lin, and Sun (2018), and Iandola, Han, Moskewicz, Ashraf, Dally, and Keutzer (2016) as a method of learning a computationally efficient model while Meller, Finkelstein, Almog, and Grobman (2019) and Goncharenko, Denisov, Alyamkin, and Terentev (2019) have iterated and improved on those ideas. Knowledge distillation was first demonstrated by Hinton, Vinyals, and Dean (2015) and has been used in concert with other compression methods to train reduced-size networks as in Mishra and Marr (2017), Wu, Leng, Wang, Hu, and Cheng (2016), Polino, Pascanu, and Alistarh (2018), and Tann, Hashemi, Bahar, and Reda (2017).

A key development in the methodology of training quantized DL models is the Quantization Aware Training (QAT) algorithm which first used by Courbariaux and Bengio (2016) to retrain their quantized networks. This method utilizes the straight-through estimator, discussed by Hinton, Srivastava, and Swersky (2012) and Bengio, Léonard, and Courville (2013) to propagate gradients across the discontinuous quantization operation.

A variety of works have innovated on top of this QAT algorithm in order to improve the quantized DL model training process. Goncharenko, Denisov, Alyamkin, and Terentev (2018) focused on quantization methods that only require retraining with limited data. Baskin et al. (2018) used random perturbations to regularize the quantization process in order to achieve high test accuracy with only 3-bit weights. Li, Zhang, and Liu (2016), Courbariaux and Bengio (2016), Zhou, Wu, Ni, Zhou, Wen, and Zou (2016), Hubara, Courbariaux, Soudry, El-Yaniv, and Bengio (2017), and Tann et al. (2017) go a step further, quantizing networks to use only to binary, ternary or power-of-two valued weights that can eliminate costly multiplication operations in favor of inexpensive addition and/or bit shift operations. However, these more efficient operations require custom hardware to realize the theoretical gains they promise. Jacob et al. (2018), in contrast, focuses on practical usage and methods directly transferable to existing hardware.

Despite the myriad aforementioned results using different methods to train quantized DL models, they universally employ QAT as a fundamental component of their approach. In contrast to the many practical results that use QAT, there exists only limited theoretical guarantees of its performance, although some asymptotic convergence analysis related to the QAT is presented by Yin, Lyu, Zhang, Osher, Qi, and Xin (2019) under the assumption of an infinite training sample size. In that sense, we believe the QAT algorithm to be ripe for innovation. Thus, we position the algorithm described in Section 2.3 as a direct alternative to QAT, one with significantly stronger theoretical results as demonstrated in Sections 2.2 and 2.3.

Both QAT and our proposed Quantized ISTA algorithms attempt to transform a discrete optimization problem into a continuous optimization problem that is easier to solve, but how the two methods achieve quantization is quite different between the QAT and the Quantized ISTA. More specifically, the QAT incorporates the straight-through estimator by Hinton et al. (2012) and Bengio et al. (2013). This estimator approximates the gradient of any point (vector of fitting parameters) by the gradient of the closest quantized point. The QAT then follows a stochastic gradient descent (SGD)-like continuous optimization algorithm with the aforementioned gradient approximation. The output of this SGD-like procedure is then further discretized to yield the final quantized model. The Quantized ISTA, on the other hand, employs a penalty to induce quantization. Our numerical experimental results to be presented in Section 3 indicate a significant difference in the performance between the proposed Quantized ISTA and the QAT scheme.

1.2. Notation and network architecture

We will denote by $|\cdot|$ and $\|\cdot\|$, respectively, the 1-norm and 2-norm, while $|\cdot|$ also represents the cardinality of a finite set, if it is the argument. $\mathbb{1}(\cdot)$ denotes the index function that takes value 1 if the conditions in the argument "$\cdot$" is satisfied; otherwise, the value of this index function is zero. $vec(\cdot)$ represents the vector that collects all the elements in the argument "$\cdot$".

Let $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ be a pair of random input (features) and output (labels) with support $supp(\mathbf{x}, y)$, where $\mathcal{X} := \{\mathbf{x} \in \mathfrak{R}^d : \|\mathbf{x}\| = 1\}$, for some integer $d > 0$, and $\mathcal{Y} := \{-1, 1\}$. Let n be the sample size. Suppose that we have the knowledge of a collection of training data $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \subset \mathcal{X} \times \{-1, 1\}$, which are i.i.d. samples of $(\mathbf{x}, y)$. For convenience of analysis, we let $n \geq 2$.

Denote by $F_{NN}(\mathbf{x}, \mathbf{W})$ the NN of consideration, where $F_{NN} : \mathcal{X} \times \mathfrak{R}^p \to \mathfrak{R}$ captures the output of the NN for a given input $\mathbf{x} \in \mathcal{X}$ and fitting parameters $\mathbf{W} \in \mathfrak{R}^p$ (that is, connection weights and biases). Here, $p > 2$ is the network size. Suppose the NN has L-many hidden layers ($L \geq 3$ is an integer) and, only for notational simplicity, each hidden layer has exactly K hidden neurons ($K \geq 2$ is an integer). The output of this network, given $\mathbf{x}$ and $\mathbf{W} := [vec(\mathbf{W}_{0,\ell} : \ell = 1, \ldots, L-1); vec(\mathbf{b}_\ell : \ell = 1, \ldots, L-1); vec(\mathbf{W}_\ell : \ell = 2, \ldots, L-1); vec(\mathbf{w}_{\ell,L} : \ell = 0, \ldots, L)]$, is calculated as below, where $\mathbf{W}_{0,\ell} \in \mathfrak{R}^{d\times K}$, $\mathbf{b}_\ell \in \mathfrak{R}^K$, $\mathbf{W}_\ell \in \mathfrak{R}^{K\times K}$, $\mathbf{w}_{\ell,L} \in \mathfrak{R}^K$, and $\sigma(\mathbf{v}) = [\max\{v_1, 0\}; \max\{v_2, 0\}; \ldots]$, for any vector $\mathbf{v} = (v_j)$, represents the ReLU activation.

$$F_{NN}(\mathbf{x}, \mathbf{W}) := \mathbf{w}_{0,L}^\top \mathbf{x} + \sum_{\ell=1}^{L-1} \mathbf{w}_{\ell,L}^\top \mathbf{z}_\ell(\mathbf{x}); \quad \mathbf{z}_1(\mathbf{x}) := \sigma\left(\mathbf{W}_{0,1}^\top \mathbf{x} + \mathbf{b}_1\right);$$

$$\text{and} \quad \mathbf{z}_\ell(\mathbf{x}) := \sigma\left(\mathbf{W}_\ell^\top \mathbf{z}_{\ell-1}(\mathbf{x}) + \mathbf{W}_{0,\ell}^\top \mathbf{x} + \mathbf{b}_\ell\right), \quad \forall \ell = 2, \ldots, L-1. \quad (1.1)$$

This NN is trained by minimizing a loss function commonly discussed for binary classification (Cao & Gu, 2019a, 2019b), namely, $\mathscr{F}(z) := \ln[1 + \exp(-z)]$. Thus, the training formulation is given as

$$\inf_{\mathbf{W}=(W_j)\in\mathfrak{R}^q} \left\{ n^{-1} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right) \right.$$
$$\left. = n^{-1} \sum_{i=1}^{n} \ln\left[1 + \exp\left(-y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right] \,:\, W_j \in \mathbb{S}_\Delta,\, j \in \mathcal{Q} \right\}. \tag{1.2}$$

From (1.1), it should be noted that the NN has "skip connections" both from the input layer to each hidden neuron and from each hidden neuron to the output layer. For quantized networks, we let $\mathbb{S}_\Delta := \{\Delta \cdot k \mid k = 0, \pm 1, \pm 2, \dots\}$ be a countable set of quantization grids with an Δ-resolution ($\Delta > 0$) and we quantize the set of parameters indexed by $\mathcal{Q} \subseteq \{1, \dots, p\}$. When $\mathcal{Q} = \{1, \dots, p\}$, we say that the network is "fully quantized". Note that

2. Main results

We are now ready to present our theoretical results. Theorem 2.20 provides the promised, almost-algorithm-independent theory on generalizability.

2.1. Generalizability of globally optimal quantized solutions

Given architecture as described in Section 1.2, training an NN consists of minimizing objective function (1.2) subject to constraints (1.1). This problem can be viewed as a stochastic optimization problem with n observations in the form of (x_i, y_i). Therefore, the Sample Average Approximation (SAA) method, a well-known technique in stochastic programming (see Shapiro, Dentcheva, & Ruszczyński, 2014, Kim, Pasupathy, & Henderson, 2015, and Bertsimas, Gupta, & Kallus, 2018), can be utilized to solve this problem . In this section, we show that if the problem of training an NN is handled like an SAA problem with quantized solutions, and solved to optimality, the generalizability error of the NN is bounded as illustrated in Theorem 2.4. We make the following assumption on the data considered in this section.

Assumption 2.1. We assume that there exists an unknown, deterministic, and measurable separating function $g : \mathcal{X} \to \mathfrak{R}$ such that $\inf_{(x,y)\in supp(\mathbb{D})}\{y \cdot g(\mathbf{x})\} \geq v$ for some $v \in (0, 1)$; which means the two categories of data are separable by function g. We also assume that $\mathbb{E}[|g(\mathbf{x})|] < \infty$.

Let Ω be the model misspecification error of the NN representing the separating function g. So, we have

$$\Omega \geq \inf_{\mathbf{W}:\|\mathbf{W}\|_0 \leq p} \mathbf{E}\left[\left|F_{NN}(\mathbf{x}, \mathbf{W}) - g(\mathbf{x})\right|\right]. \tag{2.3}$$

We assume that $\mathbf{E}[|F_{NN}(\mathbf{x}, \mathbf{W})|] < \infty$ for all $\mathbf{W} : \|\mathbf{W}\|_\infty \leq R_\Omega$ for some $R_\Omega > 0$ where R_Ω is large enough to satisfy the following assumption.

Assumption 2.2. When the fitting parameters are bounded from the above by R_Ω, the NN can represent the separating function g with a model misspecification error that does not exceed Ω. In the other words, we have $\{\mathbf{W} \in \mathscr{R}^p : \mathbf{E}[|g(\mathbf{x}) - F_{NN}(\mathbf{x}, \mathbf{W})|] \leq \Omega\} \cap [-R_\Omega, R_\Omega]^p \neq \emptyset$.

We also consider the following non-critical condition on the architecture of an NN which can be verified it holds for many NN architectures including NNs with linear or ReLU activation functions in the output layer.

Assumption 2.3. For any constant C and fitting parameters $\mathbf{W}_1 \in \mathscr{R}^p : \|\mathbf{W}_1\|_\infty \leq R_\Omega$, It holds that $F_{NN}(\mathbf{x}, \mathbf{W}_1) \cdot C = F_{NN}(\mathbf{x}, \mathbf{W}_2)$ for some $\mathbf{W}_2 \in \mathscr{R}^p : \|\mathbf{W}_2\|_\infty \leq C \cdot R_\Omega$.

Theorem 2.4. *Suppose that Assumptions 2.1–2.3 hold. Consider a neural network $F_{NN}(\mathbf{x}, \mathbf{W})$ defined as in (1.1), then the expected 0–1 loss is bounded by*

$$\mathbf{E}\left[\mathbf{1}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}^{\mathbf{SAA}}) < 0\right)\right]$$
$$\leq \sqrt{\frac{8\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}} + \frac{\ln n}{v} \cdot \Omega + \frac{2}{\sqrt{n}} \tag{2.4}$$

with probability at least $1 - \frac{1}{n^p}$.

Remark 2.5. The following remarks are obtained from Theorem 2.4:

1. The architecture of NN considered in this theorem is prevalent for the binary classification problem. Further, Assumptions 2.2 and 2.3 are very generic allowing this result to be applicable to many NNs.
2. While the existing studies on generalization bounds are mainly driven for particular training algorithms, our result is not contingent on the algorithm design. Additionally, our result is specifically obtained for quantized NNs in contrast to previous works that presented their results for non-quantized NNs. Thus, our derivation can be applied to numerous algorithms specialized for quantized solutions regardless of their training approach.
3. Ignoring the first and third term of the bound, we observe that the generalization error increases logarithmically in the number of observations. In the other words, it is in the order of $\widetilde{\mathcal{O}}(\ln n)$.
4. The generalization error increases linearly with respect to the misspecification error of the NN. It implies that an architecture of NN that results in small amount of misspecification error can significantly decrease the generalization error. Note that the value of Ω is affected by various factors such as the number of hidden layers, number of neurons, activation function, and loss function of the NN as well as the properties of the approximating function g. Therefore, to obtain a more precise representation of this error, we need to restrict the problem to particular classes of NNs and functions such as what we provide in Corollary 2.6.

Corollary 2.6. *Suppose the approximating function g is a piecewise C^β function $g : [-\frac{1}{2}, \frac{1}{2}]^d \to \mathrm{R}$, and the approximating accuracy is $\xi \in (0, \frac{1}{2})$. If the NN has as many as $c \cdot \xi^{-2(d-1)/\beta}$-many weights and $c' \cdot \log_2(\beta+2) \cdot (1 + \frac{\beta}{d})$ layers, then the generalization error would be*

$$\mathbf{E}\left[\mathbf{1}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}^{\mathbf{SAA}}) < 0\right)\right]$$
$$\leq \sqrt{\frac{8\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}} + \frac{\ln n}{v} \cdot \xi + \frac{2}{\sqrt{n}} \tag{2.5}$$

with probability at least $1 - \frac{1}{n^p}$.

2.2. Generalizability of computable quantized solutions

This subsection is focused on extending the aforementioned results to quantized solutions which are not globally optimal but tractably computable (local) solutions. Specifically, we show that solutions that satisfy a set of weak second-order necessary conditions (wSONC) of a modified DL training formulation are both quantized and generalizable. Consequently, those wSONC conditions can serve as characterizations of the desired solutions to allow for simultaneous training and quantizing. However, to do this we require additional constraints on our data and network initialization.

2.2.1. The data generation process

Hereafter, we will strengthened our data Assumption 2.1 by limiting the separating function to the following form:

Assumption 2.7. For any $(\mathbf{x}, y) \in \text{supp}(\mathbf{x}, y)$, there exists a constant $\upsilon > 0$ and

$$g(\cdot) \in \left\{ G(\cdot) : G(\mathbf{x}) = \int_{\mathfrak{R}^d} C_g(\mathbf{u}) \cdot \max\left\{0, \mathbf{u}^\top \mathbf{x}\right\} \cdot P(\mathbf{u}) d\mathbf{u} : \sup_{\mathbf{u}} |C_g(\mathbf{u})| \le 1 \right\},$$

where $P(\mathbf{u})$ is the density of the standard Gaussian vectors, such that $y \cdot g(\mathbf{x}) \ge \upsilon$ for all $(\mathbf{x}, y) \in \mathscr{X} \times \mathscr{Y}$.

Remark 2.8. Assumption 2.7 follows the same settings as in Cao and Gu (2019b) and is more general than Wang et al. (2019).

2.2.2. Initialization

In general our objective (1.2) is nonconvex and thus the training quality can be dependent on the initialization. Several effective initialization schemes have been discussed by LeCun, Bottou, Orr, and Müller (2012), Hinton and Salakhutdinov (2006), Erhan, Bengio, Courville, Manzagol, Vincent, and Bengio (2010), Glorot and Bengio (2010), Glorot, Bordes, and Bengio (2011), Mishkin and Matas (2015), Saxe, McClelland, and Ganguli (2013), and Xiao, Bahri, Sohl-Dickstein, Schoenholz, and Pennington (2018). In this research, we consider the initialization scheme in Algorithm 1 below. Our following theories show the generalizability of any solution that has a better empirical risk than the initial solutions generated by Algorithm 1.

Algorithm 1. A tractable initialization scheme

Step 1. Fix a desired tolerance $\varsigma \ge 0$. Generate each entry of $\mathbf{W}_{0,\ell}^{initial}$, for all $\ell = 1, ..., L-1$ from an independent standard normal distribution and set $\mathbf{W}_{\ell}^{initial} = \mathbf{0}$ and $\mathbf{b}_\ell = \mathbf{0}$, $\ell = 1, ..., L-1$.

Step 2. Solve the following convex optimization problem approximately:

$$\inf_{(\tilde{\mathbf{w}}_{\ell,L} : \ell=0,\ldots,L-1)} \frac{1}{n} \sum_{i=1}^{n} \mathcal{F}\left(y_i \cdot \left(\mathbf{w}_{0,L}^\top \mathbf{x} + \sum_{\ell=1}^{L-1} \mathbf{w}_{\ell,L}^\top \mathbf{z}_\ell(\mathbf{W}) \right)\right). \tag{2.6}$$

Let $\left(\mathbf{w}_{\ell,L}^{initial} : \ell = 0, ..., L-1\right)$ be the ς-suboptimal solution to (2.6). (See Remark 2.10.)

Step 3. Output $\mathbf{W}^{initial} := [vec(\mathbf{W}_{0,\ell}^{initial} : \ell = 1, ..., L-1); vec(\mathbf{b}_{\ell}^{initial} : \ell = 1, ..., L-1); vec(\mathbf{W}_{\ell}^{initial} : \ell = 2, ..., L-1); vec(\mathbf{w}_{\ell,L}^{initial} : \ell = 0, ..., L)]$.

Remark 2.9. In Step 3 of Algorithm 1, Problem (2.6) is essentially the training formulation for a subnetwork of the original NN. This subnetwork is constructed as per the following: (i) All the weights and biases for connections between the input layer and each hidden layer are fixed at values determined as in Steps 1 and 2 of this algorithm. (ii) All the weights for connections between each hidden layer and the output layer are to be trained by solving Problem (2.6). (iii) All other fitting parameters are set to be zero (and thus the corresponding connections and biases are disabled).

Remark 2.10. The ς-suboptimal solution, for $\varsigma \ge 0$, refers to any solution in the ς-sublevel set of the objective function of (2.6); that is, a ς-suboptimal solution to (2.6) is any solution in the set below:

$$\left\{ (\mathbf{w}_{\ell,L} : \ell = 0, \ldots, L-1) : \frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot \left(\mathbf{w}_{0,L}^\top \mathbf{x} + \sum_{\ell=1}^{L-1} \mathbf{w}_{\ell,L}^\top \mathbf{z}_l \right)\right) \right.$$

$$\left. \le \inf_{(\tilde{\mathbf{w}}_{\ell,L} : \ell=0,\ldots,L-1)} \frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot \left(\tilde{\mathbf{w}}_{0,L}^\top \mathbf{x} + \sum_{\ell=1}^{L-1} \tilde{\mathbf{w}}_{\ell,L}^\top \mathbf{z}_l \right)\right) + \varsigma \right\}. \tag{2.7}$$

For any arbitrary choice of $\varsigma > 0$, Algorithm 1 is tractable, because not only is (2.6) convex but it is actually strongly convex on a bounded domain.

2.2.3. Weak second-order necessary conditions

To introduce the wSONC, We consider the following variation of (1.2):

$$\min_{\mathbf{W} \in \mathfrak{R}^p} \frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}, \mathbf{W})\right) + R(\mathbf{W}; \mathscr{Q}), \tag{2.8}$$

where $R(\mathbf{W}; \mathscr{Q}) := \sum_{j \in \mathscr{Q}} P_\lambda \left(\min_{q_j \in \mathbb{S}_\Delta} |W_j - q_j| \right)$ is a construct designed to enable quantization. Here, P_λ is the minimax concave penalty (MCP) formulated as $P_\lambda(t) := \int_0^t \frac{\max\{0, \theta - \varrho \cdot \lambda\}}{\varrho} d\theta$ (with $\lambda > 0$ and $\varrho > 0$ are MCP's hyper-parameters), for any $t \ge 0$, and $\mathscr{Q} \subseteq \{1, \ldots, p\}$ is the index set for parameters to be quantized.

Hereafter, we define the derivative of the activation function to be $\frac{d\sigma(z)}{dz} := \mathbb{1}(z > 0)$, which ignores the case where $z = 0$. This is a common bypass used in the literature in view of the presence of the non-differentiable point at $z = 0$. Accordingly, the gradient of the training formulation is written as

$$\frac{\partial \left[n^{-1} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right]}{\partial W_j} := n^{-1} \sum_{i=1}^{n} \left[\frac{d\mathscr{F}(z)}{dz} \right]_{z = y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})} \cdot y_i \cdot \frac{\partial F_{NN}(\mathbf{x}_i, \mathbf{W})}{\partial W_j},$$

where $\frac{\partial F_{NN}(\mathbf{x}_i, \mathbf{W})}{\partial W_j}$ can be further explicated by invoking the chain rule, which provably holds according to Berner, Elbrächter, Grohs, and Jentzen (2019). Correspondingly, we also have (c.f., $y_i \in \{-1, 1\}$ for all i) that

$$\frac{\partial^2 \left[n^{-1} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right]}{\partial W_j^2} := n^{-1} \sum_{i=1}^{n} \left[\frac{d^2\mathscr{F}(z)}{dz^2} \right]_{z = y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})} \cdot \left[\frac{\partial F_{NN}(\mathbf{x}_i, \mathbf{W})}{\partial W_j} \right]^2.$$

The aforementioned wSONC is then defined as below:

Definition 2.11. We say that $\widetilde{\mathbf{W}}$ is an wSONC solution to (2.8) for a given $\mathscr{Q}$, if it satisfies that

$$\left[\frac{\partial^2 \left[n^{-1} \sum_{i=1}^{n} \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right]}{\partial W_j^2} \right]_{\mathbf{W} = \widetilde{\mathbf{W}}} + \frac{\partial^2 R(\widetilde{\mathbf{W}}, \mathscr{Q})}{\partial \widetilde{W}_j^2} \ge 0, \quad \forall j \in \{ j \in \mathscr{Q} : W_j \notin \mathbb{S}_\Delta \}.$$

Remark 2.12. Recall that the standard second-order KKT (SO-KKT) conditions include that the hessian matrix (if it exists) of the objective function is positive semidefinite. Because the quantity $\left[\frac{\partial^2 \left[n^{-1} \sum_{i=1}^{n} \mathscr{F}(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W}))\right]}{\partial W_j^2} \right]_{\mathbf{W} = \widetilde{\mathbf{W}}} + \frac{\partial^2 R(\widetilde{\mathbf{W}}; \mathscr{Q})}{\partial \widetilde{W}_j^2}$ is on the diagonal of a hessian matrix, 2.11 is then implied by the standard SO-KKT.

Remark 2.13. While the wSONC conditions 2.11 seem technical, they admit tractable solution schemes to be explained in Section 2.3 below. The wSONC solutions can be shown to be both quantized and generalizable as in Section 2.2.4.

2.2.4. Theoretically generalizable quantization

The wSONC solutions can be shown to be both quantized and generalizable as in the following results. We start by introducing an additional assumption.

Assumption 2.14. There exists some $\mathscr{U}_{\mathscr{F}} > 0$ such that, for all $\mathbf{W} = (W_j) : \|\mathbf{W}\| \leq R_{\ell_2}$, $\frac{\partial^2 \left[n^{-1} \sum_{i=1}^{n} \mathscr{F}(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W}))\right]}{\partial W_j^2} \leq \mathscr{U}_{\mathscr{F}}$.

Again, in this assumption, the second derivative is based on the aforementioned definition that $\frac{d\sigma(z)}{dz} := \mathbb{1}(z > 0)$ to avoid the non-differentiable kink point.

Let $\mathcal{Q} := \mathcal{Q}_1 = \{j : W_j^{initial} \neq 0\}$ in (2.8), where $W_j^{initial}$ is the jth entry of $\mathbf{W}^{initial}$ generated by Algorithm 1. Then, the following corollary shows that an NN whose majority of fitting parameters are quantized can achieve the same generalization error as in (2.16); that is, no compromise in generalizability is introduced by quantizing those fitting parameters at any resolution.

Theorem 2.15. *Suppose that Assumptions 2.7 and 2.14 hold. Consider a neural network $F_{NN}(\mathbf{x}, \mathbf{W})$ defined as in (1.1). Let $\mathbf{W}^{initial}$ be the initial weights generated by Algorithm 1 and let $\widetilde{\mathbf{W}} := (\widetilde{W}_j) \in \Re^p : \|\widetilde{\mathbf{W}}\| \leq R_{\ell_2}$, for some $R_{\ell_2} \geq 1$, be a wSONC solution to (2.8) with $\mathcal{Q} := \mathcal{Q}_1$. Assume that $\frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}, \widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}}; \mathcal{Q}_1) \leq \frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}, \mathbf{W}^{initial})\right) + R(\mathbf{W}^{initial}; \mathcal{Q}_1)$, w.p.1. For any given $\Delta > 0$, if $\varrho < \mathscr{U}_{\mathscr{F}}^{-1}$, then there is a universal constant $C_4 > 0$, such that, if $K = \lceil n^{1/4} \rceil$, then*

$$\mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widetilde{\mathbf{W}}) < 0\right)\right] \leq C_4 \cdot \left(\frac{1}{n} + \frac{\ln n}{n} \cdot \sqrt{\frac{d \ln(d/L) + \ln n}{\upsilon^2}}\right)$$
$$+ C_4 \cdot \left(\frac{d \ln(d/L) \cdot (\ln n)^3}{n^{1/4} \cdot \upsilon^2} + \frac{\left[\ln n + \ln(L \cdot R_{\ell_2})\right]^{1/2}}{n^{1/4}}\right), \tag{2.9}$$

with probability at least $1 - \exp\left(-p \ln n - \frac{pL}{C_4} \ln(L \cdot R_{\ell_2})\right) - C_4 \exp\left(-d \ln\left(\frac{dn}{C_4 \cdot L}\right)\right)$. Furthermore, with probability one, all fitting parameters in $\mathcal{Q}_1$ are quantized; that is, $\mathbb{P}[\widetilde{W}_j \in \mathbb{S}_\Delta, \ \forall j \in \mathcal{Q}_1] = 1$.

Remark 2.16. Notice that in the corollary above, $|\mathcal{Q}_1| := (K^2 + K) \cdot L$-many fitting parameters are quantized into $\mathbb{S}_\Delta$. Meanwhile, there are $(K^2 + (d+2) \cdot K)$-many fitting parameters in the NN of consideration in total. Therefore, if $\frac{K^2+K}{K^2+(d+2)\cdot K} \geq 95\% \iff 19d + 18 \leq K$ (c.f., the error bound in (2.9) is equivalent to (2.18)), then the above corollary means that more than 95% of the fitting parameters in the DL model can be quantized without compromising the generalization performance. We thus conclude that, for a neural network that has a large width, the majority of the NN can be quantized without any compromise.

Below, we present the results on fully quantized networks. Let $\mathcal{Q} := \mathcal{Q}_2 = \{1, \ldots, p\}$ in (2.8). Then, the following theorem shows that a fully quantized DL model has a bounded generalization error. In this new result, the trade-off between generalizability and quantization is explicated.

Theorem 2.17. *Suppose that Assumptions 2.7 and 2.14 hold. Consider a neural network $F_{NN}(\mathbf{x}, \mathbf{W})$ defined as in (1.1). Let $\mathbf{W}^{initial}$ be the initial weights generated by Algorithm 1 and let $\widetilde{\mathbf{W}} = (\widetilde{W}_j) \in \Re^p : \|\widetilde{\mathbf{W}}\| \leq R_{\ell_2}$, for some $R_{\ell_2} \geq 1$, be a wSONC solution to (2.8). Assume that $\frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}, \widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}}; \mathcal{Q}_2) \leq \frac{1}{n} \sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}, \mathbf{W}^{initial})\right) + R(\mathbf{W}^{initial}; \mathcal{Q}_2)$, w.p.1. For any given $\Delta > 0$, if $\varrho < \mathscr{U}_{\mathscr{F}}^{-1}$, then there exists a universal constant $C_5 > 0$, such that, if $K = \lceil n^{1/4} \rceil$, then*

$$\mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widetilde{\mathbf{W}}) < 0\right)\right] \leq C_5 \cdot \left(\frac{1}{n} + \frac{\ln n}{n} \cdot \sqrt{\frac{d \ln(d/L) + \ln n}{\upsilon^2}}\right)$$
$$+ C_5 \cdot \left(\frac{d \ln(d/L) \cdot (\ln n)^3}{n^{1/4} \cdot \upsilon^2} + \frac{\left[\ln n + \ln(L \cdot R_{\ell_2})\right]^{1/2}}{n^{1/4}}\right)$$
$$+ C_5 \cdot \varrho \cdot \Delta^2 \cdot L \cdot K \cdot d, \tag{2.10}$$

with probability at least $1 - \exp\left(-p \ln n - \frac{pL}{C_5} \ln(L \cdot R_{\ell_2})\right) - C_5 \exp\left(-d \ln\left(\frac{dn}{C_5 \cdot L}\right)\right)$. Furthermore, $\mathbb{P}[\widetilde{W}_j \in \mathbb{S}_\Delta, \ \forall j \in \mathcal{Q}_2] = 1$.

Remark 2.18. For a fixed ϱ (which can be set to be $\varrho = O(1) \cdot \mathscr{U}_{\mathscr{F}}^{-1}$), the generalization performance of a fully quantized NN model may deteriorate linearly in both the network width K and the depth L. Meanwhile, the same performance metric improves quadratically as the resolution improves (namely, Δ vanishes). Notice that this result is established on a tractable local solution (a tractable algorithm will be presented subsequently). To our knowledge, this is the first explication between the tradeoff between generalization performance and resolution in quantization at a tractably computable solution to the training formulation of DL.

2.3. A novel quantization algorithm and its theoretical guarantee

This section presents a solution scheme that generates a solution to satisfy the aforementioned characterization of generalizable and quantized solutions. For ease of presentation, we consider an abstract optimization problem with evident correspondence with problems (2.8) with $f : \Re^p \to \Re$,

$$\min_{\mathbf{W}=(W_j)\in\Re^p} f_\lambda(\mathbf{W}) := f(\mathbf{W}) + R(\mathbf{W}; \mathcal{Q}) \tag{2.11}$$

The computing procedures of this solution scheme is provided as below:

Algorithm 2. A simultaneous quantization and training algorithm

Step 1. Initialize fitting parameters $\mathbf{W}^0 = \mathbf{W}^{\frac{1}{2}}$ and set the iteration count $\kappa := 0$. Choose termination tolerance $\epsilon > 0$.

Step 2. Let $q_j^\kappa \in \arg\min\left\{q_j \in \mathbb{S}_\Delta : |q_j - W_j^\kappa|\right\}$ for all $j \in \mathcal{Q}$. Solve the following problem

$$\mathbf{W}^{\kappa+1} \in \underset{\mathbf{W}=(W_j)\in\Re^p}{\arg\min} \left\langle \mathbf{g}^k, \mathbf{W} - \mathbf{W}^\kappa \right\rangle + \frac{U_L}{2}\|\mathbf{W} - \mathbf{W}^\kappa\|^2$$
$$+ \sum_{j\in\mathcal{Q}} P_\lambda'(|W_j^\kappa - q_j^\kappa|) \cdot |W_j - q_j^\kappa|. \tag{2.12}$$

where $\mathbf{g}^k : \|\mathbf{g}^k - \nabla f(\mathbf{W}^\kappa)\| \leq \vartheta$ is an arbitrary approximation to the gradient.

Step 3. Solve the following problem

$$\mathbf{W}^{\kappa+\frac{3}{2}} \in \underset{\mathbf{W}\in\Re^p}{\arg\min} \left\langle \nabla f(\mathbf{W}^{\kappa+1}), \mathbf{W} - \mathbf{W}^{\kappa+1} \right\rangle$$
$$+ \frac{U_L}{2}\|\mathbf{W} - \mathbf{W}^{\kappa+1}\|^2 + R(\mathbf{W}; \mathcal{Q}). \tag{2.13}$$

Here U_L is a user-specified hyper-parameter such that $U_L \geq U_F$ and U_F is defined as in Assumption 2.14.

Step 4. If the stopping criteria are not met, let $\kappa := \kappa + 1$ and go to Step 2.

A viable termination criterion can be to step the algorithm when

$$\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 < \frac{\epsilon^2}{U_L^2} \tag{2.14}$$

holds for the first time. Alternatively, one may also terminate the algorithm when it reaches a user-specified maximal iteration number.

Note that both subproblems (2.12) and (2.13) are highly tractable. In particular, the first problem is essentially one iteration of iterative shrinkage thresholding algorithm and thus admits a closed form. To be more specific, let $t = W_j^\kappa - q_j^\kappa - \frac{1}{U_L} g_j^k$. Then, we have $\mathbf{W}^{\kappa+1} = \left(\text{sign}(t) \cdot \max\left\{0, |t| - \frac{P_\lambda'\left(\left|W_j^\kappa - q_j^\kappa\right|\right)}{U_L}\right\} + q_j^\kappa : j = 1, \ldots, p\right)$. (In our implementation, we further let $U_L = 1$ and $\alpha_j = P_\lambda'\left(\left|W_j^\kappa - q_j^\kappa\right|\right)$. Then, $W_j^{\kappa+1} = \text{sign}(t) \cdot \max\left\{0, |t| - \alpha_j\right\} + q_j^\kappa$.) The second problem admits a semi-closed form solution given in supplementary documents. This semi-closed form is computable in (strongly) polynomial time.

Also observe that Step 2 does not require exact gradient. Instead, any vector that approximates the gradient with ϑ accuracy suffices. One viable approach is to let $\mathbf{g}^k$ be the gradient of $\mathbf{W}^{\kappa+\frac{1}{2}}$ instead of that of $\mathbf{W}^{\kappa}$ at the κth iteration.

Theorem 2.19. *Suppose that* $f_\lambda^* := \inf_{\mathbf{W}} f_\lambda(\mathbf{W}) > -\infty$ *and the gradient* ∇f_λ *is globally Lipschitz continuous with constant* $\mathscr{U}_{\mathscr{F}}$. *Let* $\varrho < U_L^{-1} \leq \mathscr{U}_{\mathscr{F}}^{-1}$, $\varrho \cdot \lambda = \Delta$. *For any* $\epsilon \in (0, a\lambda)$ *and* $\vartheta \in [0, \frac{\epsilon}{4})$, *Algorithm 2, with the termination criterion chosen as in (2.14), stops at iteration* $\mathscr{K}^* \leq \left\lceil 4U_L \cdot \frac{f_\lambda(\mathbf{W}^0) - f_\lambda^*}{\epsilon^2} \right\rceil + 1$. *At termination,* $\mathbf{W}^{\mathscr{K}^*+\frac{3}{2}}$ *is an exact wSONC solution to (2.11). Furthermore,* $f_\lambda(\mathbf{W}^{\kappa+\frac{3}{2}}) \leq f_\lambda(\mathbf{W}^0)$ *for all* $\kappa = 1, \ldots, \mathscr{K}^*$.

2.4. Almost-algorithm-independent generalizability at tractably computable solutions

Though it is not the focus of this paper, we can obtain an almost-algorithm-independent generalizability result as a byproduct of Theorem 2.17. In this result we refer to fitting parameters of a trained neural network as a random vector $\widehat{\mathbf{W}}$, because it is a function of both the (random) training data $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, and the additional randomness in the training algorithm (e.g., the SGD).

Theorem 2.20. *Suppose that Assumption 2.7 holds. Consider a neural network* $F_{NN}(\mathbf{x}, \mathbf{W})$ *defined as in (1.1) and let* $\mathbf{W}^{initial}$ *be the initial weights generated by Algorithm 1 with arbitrarily fixed* $\varsigma \geq 0$. *Let* $\widehat{\mathbf{W}} \in \Re^p : \|\widehat{\mathbf{W}}\| \leq R_{\ell_2}$, *for some* $R_{\ell_2} \geq 1$, *be a random vector that satisfies, w.p.1.,*

$$\frac{1}{n}\sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}_i, \widehat{\mathbf{W}})\right) \leq \frac{1}{n}\sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}_i, \mathbf{W}^{initial})\right) + \Gamma, \tag{2.15}$$

for any fixed $\Gamma \in \Re$. *There exists a universal constant* $C_1 > 0$, *such that the expected 0–1 loss is bounded by*

$$\mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}}) < 0\right)\right] \leq 2\varsigma + 2\Gamma + C_1 \cdot \left(\frac{1}{n} + \frac{\ln n}{n} \cdot \sqrt{\frac{d \ln(dKL)}{LK \cdot \upsilon^2}}\right)$$
$$+ C_1 \cdot \left(\frac{d \ln(dK) \cdot (\ln n)^2}{LK \cdot \upsilon^2} + \sqrt{\frac{p \cdot \left[\ln n + L \ln(L \cdot R_{\ell_2})\right]}{n}}\right), \tag{2.16}$$

with probability at least $1 - \exp\left(-p \ln n - \frac{pL}{C_1} \ln(L \cdot R_{\ell_2})\right) - 2\exp\left(-d \ln\left(dKL/C_1\right)\right)$. *Here,* ς *is defined in Algorithm 1 as the sub-optimality gap in solving the convex problem in (2.6).*

Remark 2.21. We have a few remarks on Theorem 2.20:

1. The expected 0–1 loss, as the generalizability metric adopted in this theorem, is a common measure of generalization performance in binary classification. The same metric is used by, e.g., Cao and Gu (2019a, 2019b).
2. Γ in this theorem captures two different possibilities. Firstly, sometimes the output from Algorithm 1 should be converted to start the subsequent training process. Errors as a result of such a conversion is characterized by Γ. Secondly, in more common settings, Γ captures the effect of subsequent optimization; the empirical risk of $\widehat{\mathbf{W}}$, after the subsequent training following the initialization, may be non-trivially lower than the empirical risk of the initializer $\mathbf{W}^{initial}$. In such a case, $\Gamma < 0$. In fact, the generalization error in (2.16) becomes better as the optimization quality of $\widehat{\mathbf{W}}$ becomes better.
3. In view of the foregoing discussions on Γ and ς (as in Remark 2.10), we may as well let $\Gamma = 0$ and $\varsigma = \frac{1}{n}$. These choices of values represent the plausible scenario that the subproblem in Algorithm 1 is solved with an suboptimality gap of ς and then a (descent) training algorithm is started with the output of Algorithm 1 to ensure that $\widehat{\mathbf{W}}$ is always no worse than $\mathbf{W}^{initial}$ in terms of the empirical risk. Since $p = O(1)(d \cdot K + L \cdot K^2)$, we may have a more explicit bound on the generalization error than (2.16):

$$C_2 \cdot \left(\frac{1}{n} + \frac{\ln n}{n} \cdot \sqrt{\frac{d \ln(dK)}{LK \cdot \upsilon^2}} + \frac{d \ln(dK) \cdot (\ln n)^2}{LK \cdot \upsilon^2}\right.$$
$$\left. + \sqrt{\frac{(dK + LK^2) \cdot \left[\ln n + L \ln(LR_{\ell_2})\right]}{n}}\right), \tag{2.17}$$

with probability at least $1 - \exp\left(-p \ln n - \frac{pL}{C_2} \ln(L \cdot R_{\ell_2})\right) - \exp\left(-d \ln\left(dK/C_2\right)\right)$ for some universal constant C_2.
4. We can see that the generalization error increases only polynomially in the number of layers L. This dependence is perhaps more effective than the exponential dependence of the generalization error on L as provided by, e.g., Cao and Gu (2019a, 2019b), when their theories are applied to the same settings under Assumption 2.7. Our results on the choice of hyper-parameters, such as K, is comparatively more flexible than Cao and Gu (2019a) and Cao and Gu (2019b). A more detailed comparison will be presented subsequent to Theorem 2.20, which simplifies Corollary 2.22 .

We may properly choose hyper-parameters, such as K, the width of the DL model, to simplify the generalization error bound into the below.

Corollary 2.22. *Suppose that Assumption 2.7 holds. Consider a neural network* $F_{NN}(\mathbf{x}, \mathbf{W})$ *defined as in (1.1) and let* $\mathbf{W}^{initial}$ *be the initial weights generated by Algorithm 1 with arbitrarily fixed* $\varsigma = \frac{1}{n}$. *Let* $\widehat{\mathbf{W}} \in \Re^p$: $\|\widehat{\mathbf{W}}\| \leq R_{\ell_2}$, *for some* $R_{\ell_2} \geq 1$, *be a random vector that satisfies, w.p.1.,*

$$\frac{1}{n}\sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}_i, \widehat{\mathbf{W}})\right) \leq \frac{1}{n}\sum_{i=1}^{n} \mathscr{F}\left(y_i F_{NN}(\mathbf{x}_i, \mathbf{W}^{initial})\right).$$

There exists a universal constant $C_3 > 0$, *such that, if* $K = \lceil n^{1/4} \rceil$, *then*

$$\mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}}) < 0\right)\right] \leq C_3 \cdot \left(\frac{\ln n}{n} \cdot \sqrt{\frac{d \ln d + \ln n}{L\upsilon^2}}\right)$$
$$+ C_3 \cdot \frac{d \ln d \cdot (\ln n)^3}{Ln^{1/4} \cdot \upsilon^2} + C_3 \cdot \left(\frac{d}{n^{3/8}} + \frac{L}{n^{1/4}}\right) \cdot \left[\ln n + \ln(L \cdot R_{\ell_2})\right]^{1/2}, \tag{2.18}$$

with probability at least $1 - \exp\left(-p \ln n - \frac{pL}{C_3} \ln(L \cdot R_{\ell_2})\right) - C_3 \exp\left(-d \ln\left(\frac{dn}{C_3}\right)\right)$.

Remark 2.23. Ignoring the poly-logarithmic terms, our results can be summarized (with some oversimplification) as $\widetilde{\mathscr{O}}\left(\frac{d \cdot L}{n^{1/4} \cdot \upsilon^2}\right)$. In comparison, for the same data generation process, recent results on SGD indicate an error of $\widetilde{\mathscr{O}}\left(\frac{2^L}{\upsilon \cdot \sqrt{n}}\right)$ (Cao & Gu, 2019a, 2019b). In contrast, our bound is more efficient in a flexible regime where $L - \log_2 L \geq \log_2 d + \frac{1}{4}\log_2 n - \log_2 \upsilon \iff 2^L \geq dLn^{1/4}/\upsilon$. For example, if $d = 256$, $n = 10^5$, $\upsilon = 10^{-5}$, then this regime becomes $L \geq 34$. We thus argue that our results could be more advantageous for deeper models.

Remark 2.24. If we fix all other quantities, the rate on the sample size is $O(n^{-1/4})$, which is less appealing than several existing results that attains $O(n^{-1/2})$. Nonetheless, most results with the sharper bound are also under the assumption that global optimality is achieved, proper regularization is employed, or a specific computing procedure (several of which are shown to entail implicit regularization mechanisms) is followed. In contrast, we are focused on a "non-global" and "non-regularized" solution in the sublevel set determined by a simple initialization scheme. This solution is highly tractably computable.

Empirical results on similarly general solution have repetitively reported worse out-of-sample performance than DLs with even simple regularization schemes and or global optimization schemes. Therefore, we think that our error bound is consistent with empirical findings. We will leave to future research the incorporation of our theories with explicit and implicit regularization mechanisms or approaches (such as in Li and Liang (2018)) to reducing the dependence on n.

3. Numerical experiments

We conducted two sets of numerical experiments. (Sample codes are available at https://github.com/Hyliy/HL_Quantization.) Our first experiment is intended to link our theoretical results to practical performance. To that end, the setup follows the format of Theorem 2.19 with synthetic data. Our second experiment is intended to compare our Quantized ISTA algorithm to QAT on a common testbed. As such, we utilize CIFAR-10 (Krizhevsky, Nair, & Hinton, 2014), MNIST (LeCun, 1998), and SVHN (Netzer et al., 2011) datasets and a variety of classic ResNet architectures and train models to compare performance between the two algorithms. In all cases, our experiments were implemented using PyTorch 1.6 (Paszke et al., 2019) and run on a PC with Intel Core i9 @ 2.30 GHz, 32 GB RAM, and an NVIDIA Quadro RTX 4000 GPU (8 GB RAM).

3.1. Experiment on synthetic data

This subsection presents our results on simulated data. 2000 training and 1000 testing data samples were generated in line with Assumption 2.7, where $d = 20$ and $C_g(\mathbf{u}) := \sin(\sum_{i=1}^{d} u_i/\pi)$ with $\mathbf{u} = (u_i)$. We followed Sections 1.2 and 2.2.2 for the network architecture and initialization process. After initialization, the training of a non-quantized model ($\Delta = 0$) was done by invoking a gradient descent with constant learning rate $lr = 0.15$, and the training of a quantized model was done by invoking Algorithm 2. All quantized models were fully quantized; that is, we let $\mathcal{Q} := \{1, \ldots, p\}$ and thus all the parameters in the network all quantized. We increased the number of layers from $L = 31$ to $L = 101$ (and thus the number of hidden layers were increased from 30 to 100). For quantized models, we also varied the quantization resolution Δ from $\frac{1}{8}$ to $\frac{1}{128}$. The test results are summarized in Table 1, where we see that all the trained models resulted in reasonable out-of-sample performance. Meanwhile, increasing the number of layers or reducing the quantization resolution almost incurred no impact to the performance. In contrast, our theory predicts the generalization error to deteriorate slowly and no faster than a polynomial function of L and Δ. While the numerical results were still consistent with our error bounds (which are supposed to be over-estimators of the generalization errors), the empirical findings seemed to indicate room for further sharpening our theory.

Table 1
Out-of-sample classification errors. "lr" is the constant learning rate. When running Algorithm 2, we set $U_L := \frac{1}{lr}$. "Epochs" is the number of Epochs of the algorithm. The result with $\Delta = \text{inf}$ is for the DL without any quantization.

Δ	L	Classification error	lr	epochs
inf	31	0.896	0.15	200
inf	41	0.894	0.15	200
inf	51	0.895	0.15	200
inf	61	0.895	0.15	200
inf	71	0.895	0.15	200
inf	81	0.896	0.15	200
inf	91	0.894	0.15	200
inf	100	0.896	0.15	200
1/8	31	0.893	0.15	200
1/8	41	0.894	0.15	200
1/8	51	0.895	0.15	200
1/8	61	0.896	0.15	200
1/8	71	0.895	0.15	200
1/8	81	0.895	0.15	200
1/8	91	0.894	0.15	200
1/8	100	0.895	0.15	200
1/16	31	0.893	0.15	200
1/16	41	0.894	0.15	200
1/16	51	0.895	0.15	200
1/16	61	0.896	0.15	200
1/16	71	0.895	0.15	200
1/16	81	0.895	0.15	200
1/16	91	0.894	0.15	200
1/16	101	0.895	0.15	200
1/32	31	0.893	0.15	200
1/32	41	0.894	0.15	200
1/32	51	0.895	0.15	200
1/32	61	0.896	0.15	200
1/32	71	0.895	0.15	200
1/32	81	0.895	0.15	200
1/32	91	0.894	0.15	200
1/32	101	0.895	0.15	200
1/64	31	0.893	0.15	200
1/64	41	0.894	0.15	200
1/64	51	0.895	0.15	200
1/64	61	0.896	0.15	200
1/64	71	0.895	0.15	200
1/64	81	0.895	0.15	200
1/64	91	0.894	0.15	200
1/64	101	0.895	0.15	200
1/128	31	0.893	0.15	200
1/128	41	0.894	0.15	200
1/128	51	0.895	0.15	200
1/128	61	0.896	0.15	200
1/128	71	0.895	0.15	200
1/128	81	0.895	0.15	200
1/128	91	0.894	0.15	200
1/128	101	0.895	0.15	200

3.2. Experiment on CIFAR-10, MNIST, and SVHN datasets

This subsection presents our numerical results on CIFAR-10, MNIST, and SVHN datasets. Our experiments were focused on the ResNet family architectures (He, Zhang, Ren, & Sun, 2016) as those are commonly used for image recognition. We considered two scenarios in our experiments. The first scenario was more demanding; we quantized the entire network, including both the input and output layers, to $\mathbb{S}_\Delta$ with $\Delta = 2^{-B+1}$, i.e., the same set of candidate values representable by B-many bits. Here, B is chosen from $\{3, 4, \ldots, 9\}$. For this scenario, we obtained pretrained networks from Phan, David, Zafar, and Song (2020). The second scenario was less demanding, as we followed Dong, Yao, Gholami, Mahoney, and Keutzer (2019) to quantize only the hidden layers of the networks to $\mathbb{S}_\Delta$ with $\Delta = 2^{-B+1}$. Meanwhile, both the input and output layers were quantized to 8 bits. The corresponding pretrained networks were taken from Idelbayev (2020). The pretrained models were employed as the warm start for the quantization algorithms.

The test results for CIFAR-10, MNIST, and SVHN are shown in Tables 2, 3, and 4, respectively. From these tables, it can be seen that Quantized ISTA was less sensitive to the bit length as compared to the QAT. In particular, the former significantly outperformed the latter at lower bit lengths (when $B = 3$, 4, 5) for CIFAR-10 in the first scenario above. In other less demanding case of Scenario 2, the Quantized ISTA achieved comparable results as the QAT for CIFAR-10. As for MNIST and SVHN, two algorithms perform comparably.

Given the quantization levels used, one may estimate the relative improvement in storage size and forward pass energy usage compared to the original model. For example, quantization with $\Delta = 1/128$ and $B = 8$ (or $1/64$ and $B = 7$) leads to 71.8% (or 75%) of storage size as well as 61.25% (or 97.5%, respectively) energy reduction.

4. Conclusions

Quantized NNs has been recently shown empirically to be promising to effectively increase portability without significantly compromising

Table 2
Test accuracy of ResNet family on CIFAR-10 and the bit length B ranges from 3 to 9. The pretrained models for the two scenarios were taken from Phan et al. (2020) and Idelbayev (2020), respectively. In Scenario 1, all connections weights, including those in the input and output layers, were quantized to B-bits. In Scenario 2, only the hidden layers were quantized to B-bits, while the input and output layers were always quantized to 8-bits.

B	Scenario 1						Scenario 2					
	Resnet18		Resnet34		Resnet50		Resnet20		Resnet32		Resnet56	
	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT
3	0.6731	0.1010	0.6551	0.1010	0.6483	0.1010	0.8652	0.8637	0.8631	0.8487	0.8332	0.8071
4	0.7879	0.1010	0.7534	0.1010	0.7536	0.1010	0.9000	0.8963	0.9098	0.9032	0.9171	0.9148
5	0.8728	0.1010	0.7551	0.1010	0.7881	0.1010	0.9070	0.9069	0.9177	0.9155	0.9276	0.9242
6	0.8905	0.9009	0.8335	0.9060	0.8872	0.8822	0.9135	0.9121	0.9207	0.9225	0.9302	0.9290
7	0.8944	0.9171	0.8993	0.9208	0.9032	0.9210	0.9145	0.9146	0.9221	0.9241	0.9301	0.9316
8	0.9031	0.9200	0.9051	0.9223	0.8988	0.9267	0.9153	0.9138	0.9251	0.9252	0.9305	0.9299
9	0.9105	0.9207	0.9091	0.9221	0.8992	0.9309	0.9156	0.9167	0.9250	0.9251	0.9316	0.9316

Table 3
Test accuracy of ResNet family on MNIST and the bit length B ranges from 3 to 8. The pretrained models are self-generated. All connections weights, including those in the input and output layers, were quantized to B-bits.

B	Resnet18		Resnet34		Resnet50	
	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT
3	0.9825	0.9815	0.9753	0.9660	0.9368	0.9427
4	0.9900	0.9825	0.9897	0.9813	0.9891	0.9768
5	0.9900	0.9839	0.9889	0.9818	0.9883	0.9762
6	0.9892	0.9870	0.9866	0.9819	0.9862	0.9719
7	0.9875	0.9853	0.9818	0.9772	0.9830	0.9651
8	0.9869	0.9848	0.9783	0.9756	0.9819	0.9563

Table 4
Test accuracy of ResNet family on SVHN and the bit length B ranges from 3 to 6. The pretrained models are self-generated. All connections weights, except for those in the input and output layers, were quantized to B-bits.

B	Resnet18		Resnet34		Resnet50	
	Quantized ISTA	QAT	Quantized ISTA	QAT	Quantized ISTA	QAT
3	0.9378	0.9352	0.9430	0.9373	0.9350	0.9296
4	0.9399	0.9376	0.9404	0.9388	0.9385	0.9422
5	0.9411	0.9378	0.9434	0.9400	0.9452	0.9410
6	0.9404	0.9414	0.9452	0.9460	0.9456	0.9453

the model performance. However, limited generalization analysis on quantized NNs is currently available. This paper presents perhaps the first results that can be applied to various types of quantized NNs. We show that, for a generic NN architecture, an NN is provably generalizable if it is trained till global optimum. Furthermore, under some mild conditions on the NN architecture, a tractably computable local solution also ensures the generalizability. Both of these results are algorithm-independent; that is, they provide performance guarantee regardless of the specific designs on how to train an NN. Finally, we provide an effective local optimization algorithm for training a quantized NN and establish its computational complexity.

Our numerical experiments using synthetic data were used to test these theoretical results. Although the performance was within the bounds predicted by our theory, we did not witness the expected degradation as we increased the number of layers or decreased the quantization resolution. This indicates the potential for further tightening of the bounds.

Lastly, our experiments using CIFAR-10, MNIST, and SVHN were intended to show the potential efficacy of our algorithm on problems and network architectures that are already commonly seen. We found that the Quantized ISTA approach performed comparably well to the more common QAT algorithm, thereby demonstrating the practical effectiveness of the algorithm and its potential as an alternative approach for quantized DL problems.

CRediT authorship contribution statement

Charles Hernandez: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft. **Bijan Taslimi:** Conceptualization, Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **Hung Yi Lee:** Methodology, Software, Writing – review & editing. **Hongcheng Liu:** Conceptualization, Methodology, Writing – review & editing, Supervision, Funding acquisition. **Panos M. Pardalos:** Conceptualization, Methodology, Writing – review & editing, Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Appendix

A.1. Proof of Theorem 2.4

Proof. We divide the proof into three steps.

Step 1: Let S^ϵ and $\hat{S}_N^\delta$ be the sets of quantized ϵ-optimal solutions of the true problem and δ-optimal solutions of the SAA problem, respectively. Let ω be the set of all feasible solutions of $\mathbf{W}$. Given our assumed quantization, ω is assumed to be finite meaning that $|\omega| < \infty$. By inequality (5.103) in Shapiro et al. (2014) we know that

$$1 - Pr(\hat{S}_N^\delta \subset S^\epsilon) \le |\omega| e^{-\frac{n(\epsilon-\delta)^2}{2\sigma^2}} \tag{A.19}$$

where $Pr(\hat{S}_N^\delta \subset S^\epsilon)$ is the probability of the event that any δ-optimal solution of the SAA problem is an ϵ-optimal solution of the true problem. To utilize this inequality for function $\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}$, we need to prove it has a sub-Gaussian distribution. Because $\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\} \in (0, \ln 2]$, with probability 1 we have $\|\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}\|_{\psi_2} \le 1$. To see this, observe that by example 2.5.8(c) in Vershynin (2018) it holds that $\|\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}\|_{\psi_2} \le \frac{1}{\ln 2}\|\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}\|_\infty = \frac{1}{\ln 2}\sup_{(\mathbf{x},\mathbf{W})}\left\{\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}\right\} = \frac{1}{\ln 2}\ln 2 = 1$. It implies that the sub-Gaussian distribution assumption is satisfied and we can use inequality (A.19). Now let $\delta = 0$ which means we solve the SAA problem up to optimality. Thus, we can rewrite (A.19) as

$$\begin{aligned}&Pr\left(\left|\mathbf{E}\left[\frac{1}{n}\sum_{i=1}^{n}\min\left\{\ln 2, \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x_i}, \mathbf{W^{SAA}})\right)\right\}\right]\right.\right.\\ &\left.\left. - \min_{\mathbf{W}:\, \|\mathbf{W}\|_\infty \le \frac{\ln n}{2v} R_\Omega} \mathbf{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right\}\right]\right| \le \epsilon\right)\\ &\ge 1 - |\omega| e^{-\frac{n\epsilon^2}{2\sigma^2}},\end{aligned} \tag{A.20}$$

implying $\left|\mathbf{E}\left[\frac{1}{n}\sum_{i=1}^n \min\left\{\ln 2, \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W}^{SAA})\right)\right\}\right] - \min_{\mathbf{W}:\, \|\mathbf{W}\|_\infty \le \frac{\ln n}{2v} R_\Omega} \mathbf{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right\}\right]\right| \le \epsilon$ with probability at least $1 - |\omega| e^{-\frac{n\epsilon^2}{2\sigma^2}}$.

Step 2: We can easily see that function $\mathscr{F}$ is 1-Lipschitz continuous since its first derivative is $\mathscr{F}' = -\frac{1}{1+e^z}$ which implies $|\mathscr{F}'| \le 1$. It follows that $\mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot F_{NN}(\mathbf{x}, \mathbf{W}_1)\right)\right] - \mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot g(\mathbf{x})\right)\right] \le \frac{\ln n}{2v} \cdot \mathbf{E}\left[\left|F_{NN}(\mathbf{x}, \mathbf{W}_1) - g(\mathbf{x})\right|\right]$ for any $\mathbf{W}_1 : \|\mathbf{W}_1\|_\infty \le R_\Omega$. By utilizing this inequality, Assumption 2.3 and the fact that $\inf_u \mathscr{F}(u) = 0$, We obtain the following result

$$\begin{aligned}&\min_{\mathbf{W}\in\omega:\, \|\mathbf{W}\|_\infty \le R_\Omega} \mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right] - \inf_u \mathscr{F}(u)\\ &\le \min_{\mathbf{W}\in\omega:\, \|\mathbf{W}\|_\infty \le R_\Omega} \frac{\ln n}{2v} \cdot \mathbf{E}\left[\left|F_{NN}(\mathbf{x}, \mathbf{W}) - g(\mathbf{x})\right|\right] + \mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot g(\mathbf{x})\right)\right]\\ &\le \frac{\ln n}{2v} \cdot \Omega + \mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot g(\mathbf{x})\right)\right]\end{aligned} \tag{A.21}$$

Since we assume that we have $y{\cdot}g(\mathbf{x}) \ge v$ for all $(\mathbf{x}, y) \in supp(\mathscr{D})$, then it follows that $\mathbf{E}\left[\mathscr{F}\left(y \cdot \frac{\ln n}{2v} \cdot g(\mathbf{x})\right)\right] = \mathbf{E}\left[\ln\left(1 + exp(-y \cdot \frac{\ln n}{2v} \cdot g(\mathbf{x}))\right)\right] \le \ln\left(1 + exp(-0.5 \ln n)\right) \le \frac{1}{\sqrt{n}}$. Thus, inequality (A.21) can be simplified to

$$\min_{\mathbf{W}\in\omega:\, \|\mathbf{W}\|_\infty \le R_\Omega} \mathbf{E}\left[\mathscr{F}\left(y{\cdot}\frac{\ln n}{2v}{\cdot}F_{NN}(\mathbf{x}, \mathbf{W})\right)\right] - \inf_u \mathscr{F}(u) \le \frac{\ln n}{2v}{\cdot}\Omega + \frac{1}{\sqrt{n}}. \tag{A.22}$$

In addition, by Assumption 2.3, $\frac{\ln n}{2v} \cdot F_{NN}(\mathbf{x}, \mathbf{W})$ can be represented by the same NN architecture $F_{NN}(\mathbf{x}, \mathbf{W}') = \frac{\ln n}{2v} \cdot F_{NN}(\mathbf{x}, \mathbf{W})$ for some new fitting parameters $\mathbf{W}' : \|\mathbf{W}'\|_\infty \le \frac{\ln n}{2v} \cdot R_\Omega$. Therefore, we have

$$\min_{\mathbf{W}\in\omega:\, \|\mathbf{W}\|_\infty \le \frac{\ln n}{2v} \cdot R_\Omega} \mathbf{E}\left[\mathscr{F}\left(y{\cdot}F_{NN}(\mathbf{x}, \mathbf{W})\right)\right] - \inf_u \mathscr{F}(u) \le \frac{\ln n}{2v}{\cdot}\Omega + \frac{1}{\sqrt{n}}. \tag{A.23}$$

Step 3: By Assumption 2.2 and since the NN is assumed to be fully quantized, we have $|\omega| = (\frac{R_\Omega \ln n}{\Delta v} + 1)^p$. We let $\epsilon = \sqrt{\frac{2\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}}$. Thus, now we can write $|\omega| e^{-\frac{n\epsilon^2}{2\sigma^2}} = (\frac{R_\Omega \ln n}{\Delta v} + 1)^p\, e^{-\frac{n\left(\sqrt{\frac{2\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}}\right)^2}{2\sigma^2}} = (\frac{R_\Omega \ln n}{\Delta v} + 1)^p e^{\left(\ln\left(n(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1)\right)^{-p}\right)} = (\frac{R_\Omega \ln n}{\Delta v} + 1)^p \left(n(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1)\right)^{-p} = \frac{1}{n^p}$. Therefore, the probability $1 - |\omega| e^{-\frac{n\epsilon^2}{2\sigma^2}}$ can be written as $1 - \frac{1}{n^p}$. Since $\mathbf{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right\}\right] \le \mathbf{E}\left[\mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right]$, by combining (A.20) and (A.23), and since we let $\epsilon = \sqrt{\frac{2\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}}$, we obtain that

$$\begin{aligned}&\mathbf{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W^{SAA}})\right)\right\}\right]\\ &\le \sqrt{\frac{2\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}} + \frac{\ln n}{2v} \cdot \Omega + \frac{1}{\sqrt{n}}\end{aligned} \tag{A.24}$$

with probability at least $1 - \frac{1}{n^p}$.

Furthermore, since $\mathbf{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}^{SAA})\right)\right\}\right] \ge 0.5 \cdot \mathbf{E}\left[\mathbf{1}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}^{SAA}) < 0\right)\right]$, We have

$$\begin{aligned}&\mathbf{E}\left[\mathbf{1}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W^{SAA}}) < 0\right)\right]\\ &\le 2\sqrt{\frac{2\sigma^2 p\left(\ln n + \ln\left(\frac{R_\Omega \cdot \ln n}{\Delta v} + 1\right)\right)}{n}} + \frac{\ln n}{v} \cdot \Omega + \frac{2}{\sqrt{n}}\end{aligned} \tag{A.25}$$

to be satisfied with probability at least $1 - \frac{1}{n^p}$. □

A.2. Proof of Corollary 2.6

Proof. Using the results of Theorem 3.1 in Petersen and Voigtlaender (2018), under the stated condition in this corollary we have $\|F_{NN}(\mathbf{x}, \mathbf{W}) - g(\mathbf{x})\|_{L^2} \le \xi$. It implies that the misspecification error of the NN is at most ξ which is more explicit than what we considered in Assumption 2.2. Therefor, by invoking this result into inequality (2.4), we obtain the generalization error bounded as inequality (2.5) with probability at least $1 - \frac{1}{n^p}$. □

A.3. Semi-closed form to subproblem (2.13) in Step 3 in Algorithm 2

For any given $\mathscr{Q} \subseteq \{1, \dots, p\}$. The semi-closed form solution to (2.13) in Step 3 of Algorithm 2 is given as

$$W_j^{\kappa+1} = \begin{cases} \arg\min\left\{\left(\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})\right)_j \cdot \left(W_j - W_j^{\kappa+\frac{1}{2}}\right) \right. \\ \left. + \frac{L}{2} \cdot \left(W_j - W_j^{\kappa+\frac{1}{2}}\right)^2 + P_\lambda\left(\min_{q_j\in\mathbb{S}_\Delta} |W_j - q_j|\right) \,:\, W_j \in \mathbb{S}_\Delta\right\}, \\ \qquad \text{if } j \in \mathscr{Q}; \\ W_j^{\kappa+\frac{1}{2}} - \frac{1}{L} \cdot \nabla f(\mathbf{W}^{\kappa+\frac{1}{2}}) \\ \qquad \text{if } j \notin \mathscr{Q}. \end{cases} \tag{A.26}$$

where $\left(\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})\right)_j$ is the jth entry of $\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})$. The formulation for $j \in \mathscr{Q}$ is less trivial than the alternative case. Intuitively, the calculation therein is to find the best quantization grid in $\mathbb{S}_\Delta$ to minimize the one-dimensional optimization problem, which can be further reduced into comparing the two quantization grids that are closest to $W_j^{\kappa+\frac{1}{2}}$ −

$\frac{1}{L}\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})$, in terms of the cost function of the one-dimensional problem $\left\{\left(\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})\right)_j \cdot \left(W_j - W_j^{\kappa+\frac{1}{2}}\right) + \frac{L}{2}\cdot\left(W_j - W_j^{\kappa+\frac{1}{2}}\right)^2 + \sum_{j\in\mathscr{Q}} P_\lambda\left(|W_j - W_j^{\kappa+\frac{1}{2}}|\right) \,:\, W_j \in \mathbb{S}_\Delta\right\}$. It is evident that this one-dimensional optimization problem with finite feasible region can be solved in strongly polynomial time.

To derive this semi-closed form, we observe that (2.13) is equivalently decomposed into one-dimensional optimization problems written as below: For each $j = 1,\ldots,p$

$$\begin{cases} \min\left\{\left(\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})\right)_j \cdot \left(W_j - W_j^{\kappa+\frac{1}{2}}\right) \right. \\ \quad \left. + \frac{L}{2}\cdot\left(W_j - W_j^{\kappa+\frac{1}{2}}\right)^2 + P_\lambda\left(\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j|\right) \,:\, W_j \in \Re\right\}, \\ \qquad \text{if } j \in \mathscr{Q}; \\ \min\left\{\left(\nabla f(\mathbf{W}^{\kappa+\frac{1}{2}})\right)_j \cdot \left(W_j - W_j^{\kappa+\frac{1}{2}}\right) + \frac{L}{2}\cdot\left(W_j - W_j^{\kappa+\frac{1}{2}}\right)^2 \,:\, W_j \in \Re\right\}, \\ \qquad \text{if } j \notin \mathscr{Q}. \end{cases} \tag{A.27}$$

In the second case with $j \notin \mathscr{Q}$, the problem is to minimize a quadratic function. Thus, the closed-form to it is evidently consistent with (A.26).

As for the case with $j \in \mathscr{Q}$, we observe that, because $a < L^{-1}$, the (global) optimal solution to this optimization problem must satisfy second-order KKT conditions, which imply that the hessian matrix of the objective function should be positive semidefinite. From the diagonals of this matrix, we have $L + P_\lambda''(|W_j - W_j^{\kappa+1/2}) \geq 0$, if $P_\lambda''\left(\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j|\right)$ exists. Observe that $P_\lambda''\left(\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j|\right) = -1/a$ must exist if $|W_j - q_j| \in (0, \lambda\cdot\varrho) = (0,\Delta)$ by the definition of P_λ. Therefore, the satisfaction of second-order KKT conditions implies that $L - \varrho^{-1} \geq 0$, if $\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j| \in (0,\Delta)$. This contradicts with our specification of hyper-parameter that $\varrho < L^{-1}$. The contradiction means that $\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j| \notin (0,\Delta)$. Furthermore, $\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j| \neq \Delta$; this is because, otherwise, if $W_j - q_j^* = \Delta$ (where q_j^* is the optimal q_j) then $W_j - (q_j^* + \Delta) = 0$, implying that $q_j^* + \Delta$ is actually the optimal quantization point, because by our assumption, $q_j^* + \Delta \in \mathbb{S}_\Delta$. The same argument applies to the case where $W_j - q_j^* = -\Delta$. Therefore, it must be the case that $\min_{q_j\in\mathbb{S}_\Delta}|W_j - q_j| = 0$. In the other words, solving first problem in (A.27) (for $j \in \mathscr{Q}$) is equivalent to solving the first problem in (A.26).

A.4. Proof of Theorem 2.15

Proof. We will divide the proof into two steps. Step 1 shows that wSONC solutions are quantized by using the properties of wSONC. Step 2 then shows that those quantized solutions are generalizable by invoking Theorem 2.20.

Step 1. By the definition of wSONC as in Definition 2.11, it holds that $\left[\frac{\partial^2\left[n^{-1}\sum_{i=1}^n \mathscr{F}(y_i\cdot F_{NN}(\mathbf{x}_i,\mathbf{W}))\right]}{\partial W_j^2}\right]_{\mathbf{W}=\widehat{\mathbf{W}}} + \frac{\partial^2 R(\widehat{\mathbf{W}},\mathbf{x})}{\partial \widehat{W}_j^2} \geq 0$ for all $j \in \mathscr{Q}_2 : \widehat{W}_j \notin \mathbb{S}_\Delta$, w.p.1. By the definition of U_L and the fact that $P_\lambda''(|t|) = -\frac{1}{a}$ for any $t \notin \mathbb{S}_\Delta$, the wSONC condition implies that $U_L \geq \frac{1}{a}$, if there exists $j : \widehat{W}_j \notin \mathbb{S}_\Delta$, which contradicts with the assumption that $U_L < \frac{1}{a}$. Therefore, $\mathbb{P}[\widehat{W}_j \notin \mathbb{S}_\Delta, \forall j \in \mathscr{Q}_1] = 1$.

Step 2. By assumption, $\frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}};\mathscr{Q}_2) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\mathbf{W}^{initial})\right) + R(\mathbf{W}^{initial};\mathscr{Q}_2)$, w.p.1. Further notice that $R(\mathbf{W}^{initial};\mathscr{Q}_2) = 0$ because $\widehat{W}_j = 0$ for all $j \in \mathscr{Q}_2$. Thus, $\frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}};\mathscr{Q}_2) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\mathbf{W}^{initial})\right)$. We may then invoke Theorem 2.20 with $\varsigma = 1/n$, $\Gamma = 0$, and $K = \lceil n^{1/4}\rceil$ to obtain the desired result. □

A.5. Proof of Theorem 2.17

Proof. The proof is closely similar to that for Theorem 2.15. We will divide the proof into two steps. Step 1 shows that wSONC solutions are quantized by using the properties of wSONC. Step 2 then shows that those quantized solutions are generalizable by invoking Theorem 2.20.

Step 1. By the definition of wSONC as in Definition 2.11, it holds that $\left[\frac{\partial^2\left[n^{-1}\sum_{i=1}^n \mathscr{F}(y_i\cdot F_{NN}(\mathbf{x}_i,\mathbf{W}))\right]}{\partial W_j^2}\right]_{\mathbf{W}=\widehat{\mathbf{W}}} + \frac{\partial^2 R(\widehat{\mathbf{W}},\mathbf{x})}{\partial \widehat{W}_j^2} \geq 0$ for all $j \in \mathscr{Q}_2 : \widehat{W}_j \notin \mathbb{S}_\Delta$, w.p.1. By the definition of U_L and the fact that $P_\lambda''(|t|) = -\frac{1}{a}$ for any $t \notin \mathbb{S}_\Delta$, the wSONC condition implies that $U_L \geq \frac{1}{a}$, if there exists $j : \widehat{W}_j \notin \mathbb{S}_\Delta$, which contradicts with the assumption that $U_L < \frac{1}{a}$. Therefore, $\mathbb{P}[\widehat{W}_j \notin \mathbb{S}_\Delta, \forall j \in \mathscr{Q}_2] = 1$.

Step 2. By assumption, $\frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}};\mathscr{Q}_2) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\mathbf{W}^{initial})\right) + R(\mathbf{W}^{initial};\mathscr{Q}_2)$, w.p.1. Further notice that $R(\mathbf{W}^{initial};\mathscr{Q}_2) = 0$ because $\widehat{W}_j = 0$ for all $j \in \mathscr{Q}_1$. Thus, $\frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\widetilde{\mathbf{W}})\right) + R(\widetilde{\mathbf{W}};\mathscr{Q}_2) \leq \frac{1}{n}\sum_{i=1}^n \mathscr{F}\left(y_i F_{NN}(\mathbf{x},\mathbf{W}^{initial})\right) + R(\mathbf{W}^{initial};\mathscr{Q}_2)$. Notice that $P_\lambda(|\widehat{W}_j|) \leq \frac{\varrho\lambda^2}{2}$, for all $j \in \mathscr{Q}_2\backslash\mathscr{Q}_1$, and that $W_j^{initial} = 0$ for all $j \in \mathscr{Q}_1$, where $\mathscr{Q}_1$ is defined as in Theorem 2.15. Since $|\mathscr{Q}_2\backslash\mathscr{Q}_2| \leq (d+1)\cdot K\cdot L$, we have $R(\widehat{\mathbf{W}};\mathscr{Q}_2) \leq \frac{\varrho\lambda^2}{2}\cdot(d+1)\cdot K\cdot L$. Since $\varrho\lambda = \Delta$, we have $\lambda = \frac{\Delta}{\varrho}$. Combining the above, $R(\widehat{\mathbf{W}};\mathscr{Q}_2)\frac{\varrho\lambda^2}{2} = \sum_{j=1}^p P_\lambda(|\widehat{W}_j|) \leq |\mathscr{Q}_2\backslash\mathscr{Q}_2|\cdot\frac{\varrho\lambda^2}{2} = (d+1)\cdot K\cdot L\cdot\frac{\Delta^2}{\varrho}\cdot L\cdot K$. We may then invoke Theorem 2.20 with $\varsigma = 1/n$, $\Gamma = 0$, and $K = \lceil n^{1/4}\rceil$ to obtain the desired result. □

A.6. Proof of Theorem 2.19

Proof. Hereafter, we let $q_j^{\kappa+1} \in \arg\min\{|q - W_j^{\kappa+1}| \,:\, q \in \mathbb{S}_\Delta\}$ and $q_j^\kappa \in \arg\min\{|q - W_j^\kappa| \,:\, q \in \mathbb{S}_\Delta\}$. Notice that a well-known inequality under the $\mathscr{U}_\mathscr{F}$-Lipschitz continuity of the gradient ∇f (c.f., $U_L \geq \mathscr{U}_\mathscr{F}$) yields that

$$f(\mathbf{W}) - f(\mathbf{W}^\kappa) \leq \langle\nabla f(\mathbf{W}^\kappa), \mathbf{W} - \mathbf{W}^\kappa\rangle + \frac{U_L}{2}\|\mathbf{W} - \mathbf{W}^\kappa\|^2 \tag{A.28}$$

for all $\mathbf{W} \in \Re^p$. By the optimality condition to (2.12), we have that

$$\nabla f(\mathbf{W}^\kappa) - \mathbf{g}^k = \nabla f(\mathbf{W}^\kappa) + U_L\cdot\left(\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\right) + \boldsymbol{\gamma}^{\kappa+1} \tag{A.29}$$

where $\boldsymbol{\gamma}^{\kappa+1} := \left(P_\lambda'\left(\left|W_j^\kappa - q_j^\kappa\right|\right)\cdot\partial\left|W_j^{\kappa+1} - q_j^\kappa\right| \,:\, j = 1,\ldots,p\right)$ and $\partial|\cdot|$ is the partial differential of $|\cdot|$.

Combining (A.28) and (A.29), we then have

$$\begin{aligned} f(\mathbf{W}^{\kappa+1}) - f(\mathbf{W}^\kappa) &\leq -U_L\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 - \langle\boldsymbol{\gamma}^{\kappa+1}, \mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\rangle \\ &\quad + \frac{U_L}{2}\cdot\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 + \langle\nabla f(\mathbf{W}^\kappa) - \mathbf{g}^k, \mathbf{W}^{k+\frac{1}{2}} - \mathbf{W}^k\rangle \\ &= -\frac{U_L}{2}\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 - \langle\boldsymbol{\gamma}^{\kappa+1}, \mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\rangle \\ &\quad + \langle\nabla f(\mathbf{W}^\kappa) - \mathbf{g}^k, \mathbf{W}^{k+\frac{1}{2}} - \mathbf{W}^k\rangle \\ &\leq -\frac{U_L}{2}\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 - \langle\boldsymbol{\gamma}^{\kappa+1}, \mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\rangle + \vartheta\cdot\|\mathbf{W}^{k+\frac{1}{2}} - \mathbf{W}^k\|. \end{aligned}$$

Because $P_\lambda(t)$ is concave, we know that $\sum_{j=1}^p\left[P_\lambda'(|W_j^\kappa - q_j^\kappa|)\cdot\left(|W_j^{\kappa+1} - q_j^\kappa| - |W_j^\kappa - q_j^\kappa|\right)\right] \geq P_\lambda(|W_j^{\kappa+1} - q_j^\kappa|) - P_\lambda(|W_j^\kappa - q_j^\kappa|) \geq P_\lambda(|W_j^{\kappa+1} - q_j^{\kappa+1}|) - P_\lambda(|W_j^\kappa - q_j^\kappa|)$. The last inequality is due to the definition of $q_j^{\kappa+1}$. Therefore,

$$\begin{aligned} &f(\mathbf{W}^{\kappa+1}) + \sum_{j=1}^p P_\lambda(|W_j^{\kappa+1} - q_j^{\kappa+1}|) - f(\mathbf{W}^\kappa) - \sum_{j=1}^p P_\lambda(|W_j^\kappa - q_j^\kappa|) \\ &\quad \leq -\frac{U_L}{2}\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|^2 + \vartheta\cdot\|\mathbf{W}^{\kappa+1} - \mathbf{W}^\kappa\|. \end{aligned} \tag{A.30}$$

Considering, again, the well-known inequality under the $\mathscr{U}_{\mathscr{F}}$-Lipschitz continuity of the gradient ∇f (c.f., $U_L \geq \mathscr{U}_{\mathscr{F}}$), we have

$$f(\mathbf{W}^{\kappa+\frac{3}{2}}) - f(\mathbf{W}^{\kappa+1}) \leq \langle \nabla f(\mathbf{W}^{\kappa+1}), \mathbf{W}^{\kappa+\frac{3}{2}} - \mathbf{W}^{\kappa+1}\rangle + \frac{U_L}{2}\|\mathbf{W}^{\kappa+\frac{3}{2}} - \mathbf{W}^{\kappa+1}\|^2. \tag{A.31}$$

This, combined with (2.13), yields that

$$f(\mathbf{W}^{\kappa+\frac{3}{2}}) + \sum_{j=1}^{p} P_\lambda\left(|W_j^{\kappa+\frac{3}{2}} - q_j^{\kappa+\frac{3}{2}}|\right) - f(\mathbf{W}^{\kappa+1}) \tag{A.32}$$

$$\leq \langle \nabla f(\mathbf{W}^{\kappa+1}), \mathbf{W}^{\kappa+\frac{3}{2}} - \mathbf{W}^{\kappa+1}\rangle + \frac{U_L}{2}\|\mathbf{W}^{\kappa+\frac{3}{2}} - \mathbf{W}^{\kappa+1}\|^2$$

$$+ \sum_{j=1}^{p} P_\lambda\left(|W_j^{\kappa+1} - q_j^{\kappa+1}|\right) \tag{A.33}$$

where the last inequality is due to the fact that $\mathbf{W}^{\kappa+\frac{3}{2}}$ is an optimal solution to the per-iteration problem (2.13). As a result,

$$f(\mathbf{W}^{\kappa+\frac{3}{2}}) + \sum_{j=1}^{p} P_\lambda\left(|W_j^{\kappa+\frac{3}{2}} - q_j^{\kappa+\frac{3}{2}}|\right)$$

$$\leq f(\mathbf{W}^{\kappa+1}) + \sum_{j=1}^{p} P_\lambda\left(|W_j^{\kappa+1} - q_j^{\kappa+1}|\right). \tag{A.34}$$

Because the algorithm terminates when $\frac{U_L}{2}\|\mathbf{W}^{\kappa+1} - \mathbf{W}^{\kappa}\|^2 < \frac{\epsilon^2}{2U_L}$, the above inequality (A.30) implies that, if the algorithm does not terminate at the κth iteration, then $f(\mathbf{W}^{\kappa+1}) + \sum_{j=1}^{p} P_\lambda(|W_j^{\kappa+1} - q_j^{\kappa+1}|) - f(\mathbf{W}^{\kappa}) - \sum_{j=1}^{p} P_\lambda(|W_j^{\kappa} - q_j^{\kappa}|) \leq -\frac{\epsilon^2}{4U_L}$. Since the algorithm terminates at iteration $\mathscr{K}$, the above inequality holds for all $\kappa = 1, \ldots, \mathscr{K} - 1$. Combining these inequalities, we have

$$f(\mathbf{W}^{\mathscr{K}+1}) + \sum_{j=1}^{p} P_\lambda(|W_j^{\mathscr{K}+1} - q_j^{\kappa+1}|) - f(\mathbf{W}^0) - \sum_{j=1}^{p} P_\lambda(|W_j^0 - q_j^0|)$$

$$\leq -\frac{\epsilon^2}{4U_L} \cdot (\mathscr{K} - 1).$$

Apparently, $\mathscr{K} \leq \left\lceil \frac{4U_L \cdot \left(f_\lambda(\mathbf{W}^0) - f_\lambda^*\right)}{\epsilon^2} \right\rceil + 1$, because, otherwise, $f_\lambda(\mathbf{W}^{\mathscr{K}+1}) < f_\lambda^*$. This is impossible due to the definition of f_λ^*.

Because, if $\mathbf{W}_j^{\mathscr{K}+\frac{3}{2}} \notin \mathbb{S}_\Delta$, then the second-order necessary conditions to (2.13) (at the $(\mathscr{K} - 1)$th iteration) yield that $U_L - 1/a \geq 0$, which contradicts with the assumption that $a < U_L^{-1}$. Therefore, it holds that $\mathbf{W}^{\mathscr{K}+\frac{3}{2}} \in \mathbb{S}_\Delta^p$ and verifiably the wSONC is satisfied, as claimed.

Finally, combining (A.30), we have that $f(\mathbf{W}^{\kappa+1}) + R(\mathbf{W}^{\kappa+1}; \mathscr{Q}) \leq f(\mathbf{W}^{\kappa}) + R(\mathbf{W}^{\kappa}; \mathscr{Q})$, for all $\kappa \geq 0$. Applying this inequality recursively, we know that $f(\mathbf{W}^{\kappa}) + R(\mathbf{W}^{\kappa}; \mathscr{Q}) \leq f(\mathbf{W}^0) + R(\mathbf{W}^0; \mathscr{Q})$, for all $\kappa = 0, \ldots, \mathscr{K}^*$. Further invoking (A.34), we have the desired result that $f_\lambda(\mathbf{W}^{\kappa+\frac{3}{2}}) \leq f_\lambda(\mathbf{W}^0)$ for all $\kappa = 1, \ldots, \mathscr{K}^*$. □

A.7. Proof of Theorem 2.20

Proof. We divide the proof into four steps. The first three steps are necessary components to be combined in Step 4, which completes the proof. We denote by $c_1, c_2, \ldots$ potentially different universal constants.

Step 1. Step 1 is focused on verifying that $C_g(\xi) \cdot \max\{\xi^\top \mathbf{x}, 0\}$, for a fixed $\mathbf{x} \in \mathscr{X}$ and a standard Gaussian random vector with i.i.d. entries $\xi \in \mathfrak{R}^d$, is a sub-exponential random variable.

To see this, observe that $\xi^\top \mathbf{x}$ follows a standard normal distribution, because $\|\mathbf{x}\| = 1$ and ξ have i.i.d. standard normal entries. Thus, $\|\xi^\top \mathbf{x}\|_{\psi_2} \leq c_1$, where $\|\cdot\|_{\psi_2}$ is the subgaussian norm. Consequently, $\mathbb{P}[|\xi^\top \mathbf{x}| \geq t] \leq 2\exp(-c \cdot t^2/c_1)$, for any $t \geq 0$. Observe that $\mathbb{P}\left[\left|\max\left\{0, \xi^\top \mathbf{x}\right\}\right| \geq t\right] = \mathbb{P}\left[\left|\max\left\{0, \xi^\top \mathbf{x}\right\}\right| \geq t \;\middle|\; \xi^\top \mathbf{x} \geq 0\right] \cdot \mathbb{P}[\xi^\top \mathbf{x} \geq 0] = \mathbb{P}\left[\xi^\top \mathbf{x} \geq t\right] \leq \mathbb{P}\left[|\xi^\top \mathbf{x}| \geq t\right] \leq 2\exp\left(-c \cdot t^2/c_1\right)$, for any $t \geq 0$. Thus, by the definition of a subgaussian random variable, $\left\|\max\{0, \xi^\top \mathbf{x}\}\right\|_{\psi_2} \leq c_2$.

Let $\|\cdot\|_{\psi_1}$ be the sub-exponential norm. Then, by Lemma 2.7.7 of Vershynin (2018), $\|C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}\|_{\psi_1} \leq \|C_g(\xi)\|_{\psi_2} \cdot \|\max\{0, \xi^\top \mathbf{x}\}\|_{\psi_2} \leq c_3$, and thus $\|C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\} - \mathbb{E}_\xi[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}]\|_{\psi_1} \leq c_4$, which means that both $C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}$ and $C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\} - \mathbb{E}_\xi[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}]$ are sub-exponential, as desired in Step 1.

Step 2. This step employs the epsilon-net argument to derive an upper bound on

$$\sup_{\mathbf{x}:\, \|\mathbf{x}\|=1} \left| \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \cdot \max\{0, \xi_k^\top \mathbf{x}\} - \mathbb{E}\left[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}\right] \right|.$$

This bound will later be useful to show the efficacy of the initialization algorithm in Algorithm 1. For the purpose of epsilon-analysis, we need two components:

- The first component is a concentration inequality implied by the proven subexponentiality in Step 1. Let ξ_k, $k = 1, \ldots, T$, (for any integer $T \geq 1$) be i.i.d. samples of ξ. Invoking the Bernstein's inequality, we have, for any $\mathbf{x} \in \mathscr{X}$,

$$\mathbb{P}\left[\left| \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \cdot \max\{0, \xi_k^\top \mathbf{x}\} - \mathbb{E}_\xi\left[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}\}\right] \right| \right.$$

$$\left. \leq c_5 \cdot \left(\frac{1}{T} + \sqrt{\frac{1}{T}}\right)\right] \geq 1 - 2\exp(-t) \tag{A.35}$$

- The second component consists of the Lipschitz constants of both

$$C_g(\xi_k) \cdot \max\left\{0, \xi_k^\top \mathbf{x}\right\}$$

and $\mathbb{E}_\xi\left[C_g(\xi) \cdot \max\left\{0, \xi^\top \mathbf{x}\right\}\right]$ w.r.t. $\mathbf{x}$. Observe that $\left\|\frac{1}{\sqrt{T}}\sum_{k=1}^{T} \xi_k\right\|^2$ is an χ^2-distribution, whose degree of freedom is d. Therefore, by a well-known tail bound for the χ^2-distribution, $\mathbb{P}\left[\left\|\frac{1}{\sqrt{T}}\sum_{k=1}^{T} \xi_k\right\|^2 \leq d \cdot \left(1 + 2\sqrt{t} + 2t\right)\right] \geq 1 - \exp(-dt)$, which implies that $\mathbb{P}\left[\left\|\frac{1}{\sqrt{T}}\sum_{k=1}^{T} \xi_k\right\|^2 \leq 5dT\right] = \mathbb{P}\left[\left\|\frac{1}{T}\sum_{k=1}^{T} \xi_k\right\|^2 \leq 5d\right] \geq 1 - \exp(-d \cdot T)$. Also by properties of χ^2-distribution, $\left(\mathbb{E}_\xi\left[\left\|\frac{1}{\sqrt{T}}\sum_{k=1}^{T} \xi_k\right\|\right]\right)^2 \leq \mathbb{E}_\xi\left[\left\|\frac{1}{\sqrt{T}}\sum_{k=1}^{T} \xi_k\right\|^2\right] = d$. These have ensured that both the sample average and the expected functions follow Lipschitz conditions. More specifically, observe that

$$\left| \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \cdot \max\{0, \xi_k^\top \mathbf{x}_1\} - \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \cdot \max\{0, \xi_k^\top \mathbf{x}_2\} \right|$$

$$\leq \sup_{\xi \in \mathfrak{R}^d} |C_g(\xi)| \cdot \|\frac{1}{T}\sum_{k=1}^{T} \xi_k\| \cdot \|\mathbf{x}_1 - \mathbf{x}_2\|$$

for all (ξ_k), $\mathbf{x}_1$, and $\mathbf{x}_2$. As a result,

$$\sup_{\mathbf{x}_1, \mathbf{x}_2 \in \mathscr{X}} \left\{ \left| \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \max\{0, \xi_k^\top \mathbf{x}_1\} \right.\right.$$

$$\left.\left. - \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \max\{0, \xi_k^\top \mathbf{x}_2\} \right| - \sqrt{5d} \cdot \|\mathbf{x}_1 - \mathbf{x}_2\| \right\} \leq 0,$$

with probability at least $1 - \exp(-d \cdot T)$, and $\left|\mathbb{E}_\xi\left[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}_1\}\right] - \mathbb{E}_\xi\left[C_g(\xi) \cdot \max\{0, \xi^\top \mathbf{x}_2\}\right]\right| \leq \sup_{\xi \in \mathfrak{R}^d} |C_g(\xi)| \cdot \sqrt{d} \cdot \|\mathbf{x}_1 - \mathbf{x}_2\| \leq \sqrt{d} \cdot \|\mathbf{x}_1 - \mathbf{x}_2\|$.

We are now ready for the epsilon-net analysis. Construct a net of grids $\mathscr{B}_{\tilde{\delta}}$ such that, for any $\mathbf{x} : \|\mathbf{x}\| = 1$, there exists $\mathbf{z} \in \mathscr{B}_{\tilde{\delta}}$ such that $\|\mathbf{x} - \mathbf{z}\| \leq \frac{\tilde{\delta}}{(\sqrt{5}+1) \cdot \sqrt{d}}$. It suffices to have as many as $|\mathscr{G}_{\tilde{\delta}}| := \left\lceil \frac{2 \cdot (\sqrt{5}+1)d}{\tilde{\delta}} \right\rceil^d$ grids. Therefore,

$$\mathbb{P}\left[\max_{\mathbf{x} \in \mathscr{B}_{\tilde{\delta}}} \left| \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k) \cdot \max\{\xi_k^\top \mathbf{x}, 0\} - \mathbb{E}_\xi\left[C_g(\xi) \cdot \max\{\xi^\top \mathbf{x}, 0\}\right] \right| \right.$$

$$\geq c_5 \cdot \left(\frac{t}{T} + \sqrt{\frac{t}{T}}\right)\Bigg]$$

$$\leq 2\left[\frac{2\cdot(\sqrt{5}+1)d}{\tilde{\delta}}\right]^d \exp(-t). \tag{A.36}$$

Notice that, for any $\mathbf{z} \in \mathscr{B}_{\tilde{\delta}}$, given the events that

$$\mathscr{E}^1 := \left\{\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top \mathbf{x}, 0\} - \mathbb{E}_\xi[C_g(\xi)\cdot\max\{\xi^\top\mathbf{x}, 0\}]\right| \right.$$
$$\left. \leq c_5 \cdot \left(\frac{t}{T}+\sqrt{\frac{t}{T}}\right)\right\}$$

and that

$$\mathscr{E}^2 := \left\{\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\max\{0,\, \xi_k^\top\mathbf{x}_1\} - \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\max\{0,\, \xi_k^\top\mathbf{x}_2\}\right|\right.$$
$$\left. \leq \sqrt{5d}\cdot\|\mathbf{x}_1-\mathbf{x}_2\|,\ \forall\,\mathbf{x}_1,\,\mathbf{x}_2\,:\ \|\mathbf{x}_1\|=1,\ \|\mathbf{x}_2\|=1\right\},$$

it holds that, for any $\mathbf{x}\in\Re^d$ and $\mathbf{z}\in\mathscr{B}_{\tilde{\delta}}\,:\ \|\mathbf{x}-\mathbf{z}\| \leq \frac{\tilde{\delta}}{(\sqrt{5+1})d}$,

$$\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{x}, 0\} - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\{\xi^\top\mathbf{x}, 0\}\right]\right|$$
$$\leq\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{x}, 0\} - \frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{z}, 0\}\right|$$
$$+\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{z}, 0\} - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\{\xi^\top\mathbf{z}, 0\}\right]\right|$$
$$+\left|\mathbb{E}_\xi\left[C_g(\xi)\cdot\max\{\xi^\top\mathbf{x}, 0\}\right] - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\{\xi^\top\mathbf{z}, 0\}\right]\right|$$
$$\leq\sqrt{5d}\|\mathbf{z}-\mathbf{x}\| + \left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\left\{0,\,\xi_k^\top\mathbf{z}\right\} - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\{\xi^\top\mathbf{z},0\}\right]\right|$$
$$+\sqrt{d}\|\mathbf{z}-\mathbf{x}\|$$
$$\leq c_5\cdot\left(\frac{t}{T}+\sqrt{\frac{t}{T}}\right)+(\sqrt{5}+1)\sqrt{d}\|\mathbf{z}-\mathbf{x}\| \leq c_5\cdot\left(\frac{t}{T}+\sqrt{\frac{t}{T}}\right)+\tilde{\delta}.$$

Notice that $\mathbb{P}[\mathscr{E}^1\cap\mathscr{E}^2] \geq 1-2\left[\frac{2(\sqrt{5}+1)d}{\tilde{\delta}}\right]^d\cdot\exp(-t)-\exp(-d\cdot T)$. Combining the above, we have that, for any $\tilde{\delta}>0$ and any $t\geq 0$,

$$\mathbb{P}\left[\sup_{\mathbf{x}:\,\|\mathbf{x}\|=1}\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{x}, 0\} - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\left\{\xi^\top\mathbf{x}, 0\right\}\right]\right|\right.$$
$$\left.\leq c_5\cdot\left(\frac{t}{T}+\sqrt{\frac{t}{T}}\right)+\tilde{\delta}\right]$$
$$\geq 1-2\left[\frac{2\cdot(\sqrt{5}+1)d}{\tilde{\delta}}\right]^d\cdot\exp(-t)-\exp(-d\cdot T)$$
$$=1-2\exp\left(-t+d\ln\left[\frac{2\cdot(\sqrt{5}+1)d}{\tilde{\delta}}\right]\right)-\exp(-d\cdot T). \tag{A.37}$$

We may as well let $\tilde{\delta}=1/T$ and $t=2d\ln\left[\frac{2\cdot(\sqrt{5}+1)d}{\tilde{\delta}}\right] = 2d\ln(2(\sqrt{5}+1)dT)$. Consequently, (A.37) is reduced to

$$\mathbb{P}\left[\sup_{\mathbf{x}:\,\|\mathbf{x}\|=1}\left|\frac{1}{T}\sum_{k=1}^{T} C_g(\xi_k)\cdot\max\{\xi_k^\top\mathbf{x}, 0\} - \mathbb{E}_\xi\left[C_g(\xi)\cdot\max\left\{\xi^\top\mathbf{x}, 0\right\}\right]\right|\right.$$
$$\left.\leq c_7\cdot\sqrt{\frac{d\ln(dT)}{T}}\right]$$
$$\geq 1-2\exp\left(-d\ln\left(2\cdot(\sqrt{5}+1)dT\right)\right)-\exp(-d\cdot T). \tag{A.38}$$

This completes Step 2.

Step 3. In this step, we evaluate the initial sub-optimality gap generated by Algorithm 1, making use of results from Steps 1 and 2. Notice that the initialization essentially yields a neural network as a subnetwork to the original model. This subnetwork can be captured by

$$F_{NN}(\mathbf{x},\,\mathbf{W}^{initial}) := \sum_{\ell=1}^{L-1}(\mathbf{w}_{\ell,L}^{initial})^\top\sigma\left((\mathbf{W}_{0,\ell}^{initial})^\top\mathbf{x}\right). \tag{A.39}$$

Let $\mathbf{w}_{0,\ell,k}^{initial}$ be the kth column of $\mathbf{W}_{0,\ell}^{initial}$; that is, the weights for the connections that joins the kth neuron in the ℓth layer from the input layer. Observe that, via Algorithm 1, the sequence of $\mathbf{w}_{0,\ell,k}^{initial}$, for all $k=1,\ldots,K$ and $\ell=1,\ldots,L-1$, are $(L-1)\cdot K$-many i.i.d. samples of ξ in Step 1. Invoking (A.38), with $T := K\cdot(L-1)$, we have (c.f., $y\in\{-1,1\}$), with probability $1-2\exp\left(-d\ln\left(2\cdot(\sqrt{5}+1)dK\cdot(L-1)\right)\right)-\exp(-d\cdot K\cdot(L-1))$,

$$\sup_{\mathbf{x}:\,\|\mathbf{x}\|=1}\left|\frac{y\ln n}{K\cdot(L-1)\cdot v}\sum_{k=1}^{K}\sum_{\ell=1}^{L-1}C_g(\mathbf{w}_{0,\ell,k}^{initial})\cdot\max\left\{0,\,\left(\mathbf{w}_{0,\ell,k}^{initial}\right)^\top\mathbf{x}\right\}\right.$$
$$\left.-\frac{y\cdot\ln n}{v}\mathbb{E}_\xi\left[C_g(\xi)\cdot\max\left\{0,\,\xi^\top\mathbf{x}\right\}\right]\right|$$
$$\leq c_6\cdot\ln n\cdot\sqrt{\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}}. \tag{A.40}$$

Observe that $\mathscr{F}'(z) = -\frac{\exp(-z)}{1+\exp(-z)}$ and $\mathscr{F}''(z)=\frac{\exp(z)}{(1+\exp(z))^2}$. Thus $\mathscr{F}'$ is 0.5-Lipschitz continuous. Consequently,

$$\mathscr{F}(x_1)-\mathscr{F}(x_2)\leq\mathscr{F}'(x_2)\cdot(x_1-x_2)+0.5/2\cdot(x_1-x_2)^2.$$

Also observe that $y_i\cdot E_\xi\left[C_g(\xi,\mathbf{x}_i)\cdot\max\{0,\,\xi^\top\mathbf{x}_i\}\right]\geq v \iff \frac{\ln n}{v}\cdot y_i\cdot E_\xi\left[C_g(\xi,\mathbf{x}_i)\cdot\max\{0,\,\xi^\top\mathbf{x}_i\}\right]\geq\ln n$ for all i by Assumption 2.7 and $|F'(z)| = \left|-\frac{1/n}{1+1/n}\right|\leq\frac{1}{n}$ for all $z\leq-\ln n$. We thus have that

$$\left|n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(\frac{y_i\ln n}{Tv}\sum_{k=1}^{T}C_g(\mathbf{w}_{0,l,k}^{initial})\cdot\max\left\{0,\,\left(\mathbf{w}_{0,l,k}^{initial}\right)^\top\mathbf{x}_i\right\}\right)\right.$$
$$\left.-n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(\frac{y_i\cdot\ln n}{v}\mathbb{E}_\xi\left[C_g(\xi,\mathbf{x}_i)\cdot\max\{0,\,\xi^\top\mathbf{x}_i\}\right]\right)\right|$$
$$\leq c_7\cdot\frac{1}{n}\cdot\ln n\cdot\sqrt{\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}}$$
$$+c_7\cdot(\ln n)^2\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}. \tag{A.41}$$

with probability $1-2\exp\left(-d\ln\left(2\cdot(\sqrt{5}+1)dK\cdot(L-1)\right)\right)-\exp(-d\cdot K\cdot(L-1))$. By combining (A.41) with the assumption of (2.15) and the definition of $\mathbf{W}^{initial}$ in (2.6), we have

$$n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(y_i\cdot F_{NN}(\mathbf{x}_i,\widehat{\mathbf{W}})\right)$$
$$\leq n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(y_i\cdot\frac{\ln n}{v\cdot T}\sum_{\ell=1}^{L-1}(\mathbf{w}_{\ell,L}^{initial})^\top\sigma\left((\mathbf{W}_{0,\ell}^{initial})^\top\mathbf{x}\right)\right)+\varGamma$$
$$\leq n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(\frac{y_i\cdot\ln n}{v}\mathbb{E}\left[C_g(\xi)\cdot\max\{0,\,\xi^\top\mathbf{x}_i\}\right]\right)+\varsigma$$
$$+c_7\cdot\frac{1}{n}\cdot\ln n\cdot\sqrt{\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}}$$
$$+c_7\cdot(\ln n)^2\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}+\varGamma$$
$$\leq n^{-1}\sum_{i=1}^{n}\mathscr{F}\left(\frac{\ln n}{v}\cdot v\right)+c_7\cdot\frac{1}{n}\cdot\ln n\cdot\sqrt{\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}}$$
$$+c_7\cdot(\ln n)^2\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}+\varGamma+\varsigma \tag{A.42}$$
$$=\ln(1+\exp(-\ln n))$$
$$+c_7\cdot\frac{1}{n}\cdot\ln n\cdot\sqrt{\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}}$$
$$+c_7\cdot(\ln n)^2\frac{d\ln(d\cdot K\cdot(L-1))}{K\cdot(L-1)\cdot v^2}+\varGamma+\varsigma$$

$$\leq \frac{1}{n} + c_7 \cdot \frac{1}{n} \cdot \ln n \cdot \sqrt{\frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2}} + c_7 \cdot (\ln n)^2 \frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2} + \Gamma + \varsigma. \tag{A.43}$$

with probability at least $1 - \exp\left(-d \ln\left(2 \cdot (\sqrt{5}+1) d K \cdot (L-1)\right)\right) - \exp(-d \cdot K \cdot (L-1))$. In the above, (A.42) is due to Assumption 2.7. In Step 3, we have now shown the global suboptimality of $\widehat{\mathbf{W}}$ is no more than $\frac{1}{n} + c_7 \cdot \frac{1}{n} \cdot \ln n \cdot \sqrt{\frac{d \ln(d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2}} + c_7 \cdot (\ln n)^2 \frac{d \ln(d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2} + \Gamma + \varsigma$, by further observing that $\mathscr{F}(x) \geq 0$ for all x in the domain.

Step 4. This step invokes another round of epsilon-net analysis to show that, $\widehat{\mathbf{W}}$, whose suboptimality gap can be controlled as per Step 3 above, yields the bounded generalization error as desired in the statement of this theorem. Below are the details.

Noting that $\mathscr{F}(z) := \ln(1 + \exp(-z)) \geq \min\{\ln(1 + \exp(-z)), \ln 2\} \geq \frac{1}{2} \cdot \mathbb{1}(z < 0)$, we may continue from (A.43) to obtain that, with probability at least $1 - \delta - \exp\left(-d \ln\left(2 \cdot (\sqrt{5}+1) d K \cdot (L-1)\right)\right) - \exp(-d \cdot K \cdot (L-1))$,

$$n^{-1} \sum_{i=1}^{n} \min\left\{\ln 2, \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \widehat{\mathbf{W}})\right)\right\}$$
$$\leq \frac{1}{n} + c_7 \cdot \frac{1}{n} \cdot \ln n \cdot \sqrt{\frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2}}$$
$$+ c_7 \cdot (\ln n)^2 \frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2} + \Gamma + \varsigma. \tag{A.44}$$

Notice that $\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}})\right)\right\} \in [0, \ln 2]$; that is, it has a bounded support. Therefore, the Hoeffding's inequality yields that

$$\mathbb{P}\left[\left|\frac{1}{n} \sum_{i=1}^{n} \min\left\{\ln 2, \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right\} - \mathbb{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right\}\right]\right| \geq t\right] \leq 2 \exp\left(\frac{-2nt^2}{(\ln 2)^2}\right),$$

for all $t \geq 0$. The epsilon-net argument again starts below.

Notice that, for any $\mathbf{x} \in \mathscr{X}$, the Lipschitz constant of $F_{NN}(\mathbf{x}, \cdot)$ is $L(K \cdot R_{\ell_2})^{c_7 \cdot L}$. To see this, by observation, for any $\mathbf{W}_1, \mathbf{W}_2 \in \mathfrak{R}^p : \|\mathbf{W}_1 - \mathbf{W}_2\| \leq \tau$ and any $\tau > 0$, it holds that $\left|F_{NN}(\mathbf{x}, \mathbf{W}_1) - F_{NN}(\mathbf{x}, \mathbf{W}_2)\right| \leq |(\mathbf{w}_{0,L})_1^\top \mathbf{x} + \sum_{\ell=1}^{L-1} (\mathbf{w}_{\ell,L})_1^\top (\mathbf{z}_\ell(\mathbf{x}))_1 - (\mathbf{w}_{0,L})_2^\top \mathbf{x} - \sum_{\ell=1}^{L-1} (\mathbf{w}_{\ell,L})_2^\top (\mathbf{z}_\ell(\mathbf{x}))_2| \leq \|(\mathbf{w}_{0,L+1})_1 - (\mathbf{w}_{0,L+1})_2\| \cdot \|\mathbf{x}\| + \sum_{\ell=1}^{L-1} \max\left\{\|(\mathbf{w}_{\ell,L})_1\|, \|(\mathbf{w}_{\ell,L})_2\|\right\} \cdot \|(\mathbf{z}_\ell(\mathbf{x}))_1 - (\mathbf{z}_\ell(\mathbf{x}))_2\| \leq \tau + \sum_{\ell=2}^{L-1} R_{\ell_2} \cdot \|(\mathbf{z}_\ell(\mathbf{x}))_1 - (\mathbf{z}_\ell(\mathbf{x}))_2\|$, where $\|(\mathbf{z}_\ell(\mathbf{x}))_1 - (\mathbf{z}_\ell(\mathbf{x}))_2\| \leq K \cdot R_{\ell_2} \cdot \|(\mathbf{z}_{\ell-1}(\mathbf{x}))_1 - (\mathbf{z}_{\ell-1}(\mathbf{x}))_2\| + \sum_{k=1}^{K} \|(\mathbf{w}_{0,\ell,k})_1 - (\mathbf{w}_{0,\ell,k})_2\| \cdot \|\mathbf{x}\| + \|(\mathbf{b}_\ell)_1 - (\mathbf{b}_\ell)_2\| \leq K \cdot R_{\ell_2} \cdot \|(\mathbf{z}_{\ell-1}(\mathbf{x}))_1 - (\mathbf{z}_{\ell-1}(\mathbf{x}))_2\| + K \cdot \tau + \tau \leq c_7 \cdot (K \cdot R_{\ell_2})^L \cdot \tau$. Combining the above, we thus have

$$\left|F_{NN}(\mathbf{x}, \mathbf{W}_2) - F_{NN}(\mathbf{x}, \mathbf{W}_2)\right| \leq L(K \cdot R_{\ell_2})^{c_7 \cdot L} \cdot \|\mathbf{W}_1 - \mathbf{W}_2\|. \tag{A.45}$$

Therefore, $L(K \cdot R_{\ell_2})^{c_7 \cdot L}$ is an upper bound on the Lipschitz constant of $F_{NN}(\mathbf{x}, \cdot)$ for any $\mathbf{x}$. Thus, $\min\{\ln 2, \mathscr{F}(y \cdot F_{NN}(\mathbf{x}, \mathbf{W}))\}$ is Lipschitz continuous w.r.t. $\mathbf{W}$ with constant $C_{\mathscr{L}} := L(K \cdot R_{\ell_2})^{c_8 \cdot L}$. Then, a standard epsilon-net argument leads to that, for any $\tilde{\epsilon} > 0$ and $t \geq 0$,

$$\mathbb{P}\left[\sup_{\mathbf{W}: \|\mathbf{W}\| \leq R_{\ell_2}} \left|\frac{1}{n} \sum_{i=1}^{n} \min\left\{\ln 2, \mathscr{F}\left(y_i \cdot F_{NN}(\mathbf{x}_i, \mathbf{W})\right)\right\} - \mathbb{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \mathbf{W})\right)\right\}\right]\right| \leq c_8 \cdot \left(\frac{t}{n} + \sqrt{\frac{t}{n}}\right) + \tilde{\epsilon}\right] \geq 1 - \left[\frac{R_{\ell_2} \cdot C_{\mathscr{L}}}{\tilde{\epsilon}}\right]^p \cdot \exp(-t)$$
$$= 1 - \exp\left(-t + p \ln\left[\frac{R_{\ell_2} \cdot C_{\mathscr{L}}}{\tilde{\epsilon}}\right]\right). \tag{A.46}$$

Combining (A.44) and (A.46) (where we let $t := 2n^{-1} p \ln(R_{\ell_2} \cdot C_{\mathscr{L}} n)$ and $\tilde{\epsilon} = 1/n$), we then have that

$$\mathbb{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}})\right)\right\}\right]$$
$$\leq \frac{1}{n} + c_9 \cdot \frac{1}{n} \cdot \ln n \cdot \sqrt{\frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2}}$$
$$+ c_9 \cdot (\ln n)^2 \frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2} + \Gamma + \varsigma + c_9 \sqrt{\frac{p \ln(R_{\ell_2} \cdot C_{\mathscr{L}} n)}{n}} \tag{A.47}$$

to be satisfied with probability at least $1 - \exp\left(-p \ln\left(R_{\ell_2} \cdot C_{\mathscr{L}} n\right)\right) - 2\exp\left(-d \ln\left(2 \cdot (\sqrt{5}+1) d \cdot K \cdot (L-1)\right)\right) - \exp(-d \cdot K \cdot (L-1))$.

Furthermore, because $\mathbb{E}\left[\min\left\{\ln 2, \mathscr{F}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}})\right)\right\}\right] \geq 0.5 \cdot \mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}}) < 0\right)\right]$, we have

$$\mathbb{E}\left[\mathbb{1}\left(y \cdot F_{NN}(\mathbf{x}, \widehat{\mathbf{W}}) < 0\right)\right] \leq \frac{2}{n} + c_{10} \cdot \frac{1}{n} \cdot \ln n \cdot \sqrt{\frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2}}$$
$$+ c_{10} \cdot (\ln n)^2 \frac{d \ln (d \cdot K \cdot (L-1))}{K \cdot (L-1) \cdot v^2} + 2\Gamma + 2\varsigma + c_{10} \sqrt{\frac{p \ln(R_{\ell_2} \cdot C_{\mathscr{L}} n)}{n}} \tag{A.48}$$

to be satisfied with probability at least $1 - \exp\left(-p \ln\left(R_{\ell_2} \cdot C_{\mathscr{L}} n\right)\right) - 2\exp\left(-d \ln\left(2 \cdot (\sqrt{5}+1) d \cdot K \cdot (L-1)\right)\right) - \exp(-d \cdot K \cdot (L-1))$. Further invoking the fact that $\ln(C_{\mathscr{L}}) := \ln(c_8 \cdot L \cdot (K \cdot R_{\ell_2})^{c_8 L}) = \ln(c_8 \cdot L) + c_8 \cdot L \cdot \ln(L \cdot R_{\ell_2})$, we immediately have the desired results of this theorem after simplification. □

A.8. Proof of Corollary 2.22

Proof. Immediately from simplifying Theorem 2.20 by letting $\Gamma = 0$, $\varsigma = 1/n$, and $K = O(1) \cdot n^{1/4}$. □

References

Affonso, C., Rossi, A. L. D., Vieira, F. H. A., de Leon Ferreira, A. C. P., et al. (2017). Deep learning for biological image classification. *Expert Systems with Applications*, *85*, 114–122.

Ahn, B. S., Cho, S., & Kim, C. (2000). The integrated methodology of rough set theory and artificial neural network for business failure prediction. *Expert Systems with Applications*, *18*(2), 65–74.

Baskin, C., Liss, N., Chai, Y., Zheltonozhskii, E., Schwartz, E., Giryes, R., et al. (2018). Nice: Noise injection and clamping estimation for neural network quantization. arXiv preprint arXiv:1810.00162.

Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, *2*(1), 183–202.

Bengio, Y., Léonard, N., & Courville, A. (2013). Estimating or propagating gradients through stochastic neurons for conditional computation. arXiv preprint arXiv:1308.3432.

Berner, J., Elbrächter, D., Grohs, P., & Jentzen, A. (2019). Towards a regularity theory for ReLU networks–chain rule and global error estimates. arXiv preprint arXiv:1905.04992.

Bertsimas, D., Gupta, V., & Kallus, N. (2018). Robust sample average approximation. *Mathematical Programming*, *171*(1–2), 217–282.

Brutzkus, A., Globerson, A., Malach, E., & Shalev-Shwartz, S. (2017). Sgd learns over-parameterized networks that provably generalize on linearly separable data. arXiv preprint arXiv:1710.10174.

Bu, Y., Gao, W., Zou, S., & Veeravalli, V. V. (2019). Information-theoretic understanding of population risk improvement with model compression. arXiv preprint arXiv:1901.09421.

Cao, Y., & Gu, Q. (2019a). Generalization bounds of stochastic gradient descent for wide and deep neural networks. In *Advances in neural information processing systems* (pp. 10835–10845).

Cao, Y., & Gu, Q. (2019b). A generalization theory of gradient descent for learning over-parameterized deep relu networks. arXiv preprint arXiv:1902.01384.

Courbariaux, M., & Bengio, Y. (2016). Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1. CoRR arXiv:1602.02830, URL http://arxiv.org/abs/1602.02830.

Daniely, A. (2017). SGD learns the conjugate kernel class of the network. In *Advances in neural information processing systems* (pp. 2422–2430).

Ding, Y., Liu, J., Xiong, J., & Shi, Y. (2018). On the universal approximability and complexity bounds of quantized relu neural networks. arXiv preprint arXiv:1802.03646.

Dong, Z., Yao, Z., Gholami, A., Mahoney, M. W., & Keutzer, K. (2019). Hawq: Hessian aware quantization of neural networks with mixed-precision. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 293–302).

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al.

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding back-translation at scale. arXiv preprint arXiv:1808.09381.

Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., & Bengio, S. (2010). Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, *11*(Feb), 625–660.

Foret, P., Kleiner, A., Mobahi, H., & Neyshabur, B. (2020). Sharpness-aware minimization for efficiently improving generalization. arXiv preprint arXiv:2010.01412.

Fragoso, V., Gauglitz, S., Zamora, S., Kleban, J., & Turk, M. (2011). Translatar: A mobile augmented reality translator. In *2011 IEEE workshop on applications of computer vision* (pp. 497–502). IEEE.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256).

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315–323).

Goncharenko, A., Denisov, A., Alyamkin, S., & Terentev, E. (2018). Fast adjustable threshold for uniform neural network quantization (winning solution of LPIRC-II). arXiv preprint arXiv:1812.07872.

Goncharenko, A., Denisov, A., Alyamkin, S., & Terentev, E. (2019). Fast adjustable threshold for uniform neural network quantization. *International Journal of Computer and Information Engineering*, *13*(9), 499–503.

Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*(8), 10389–10397.

Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507.

Hinton, G., Srivastava, N., & Swersky, K. (2012). Neural networks for machine learning. *Coursera, Video Lectures*, *264*, 1.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y. (2017). Quantized neural networks: Training neural networks with low precision weights and activations. *Journal of Machine Learning Research*, *18*(1), 6869–6898.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). Squeezenet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv:1602.07360.

Idelbayev, Y. (2020). Proper ResNet implementation for CIFAR10/CIFAR100 in PyTorch. Accessed: 20xx-xx-xx.

Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., et al. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2704–2713).

Kabir, H., Abdar, M., Jalali, S. M. J., Khosravi, A., Atiya, A. F., Nahavandi, S., et al. (2020). Spinalnet: Deep neural network with gradual input. arXiv preprint arXiv:2007.03347.

Kim, S., Pasupathy, R., & Henderson, S. G. (2015). A guide to sample average approximation. In *Handbook of simulation optimization* (pp. 207–243). Springer.

Krizhevsky, A., Nair, V., & Hinton, G. (2014). Cifar-10 (canadian institute for advanced research). http://www.cs.toronto.edu/kriz/cifar.html.

Laine, M., & Nevalainen, O. S. (2006). A standalone OCR system for mobile cameraphones. In *2006 IEEE 17th international symposium on personal, indoor and mobile radio communications* (pp. 1–5). IEEE.

LeCun, Y. (1998). The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.

LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). Efficient backprop. In *Neural networks: tricks of the trade* (pp. 9–48). Springer.

Li, Y., & Liang, Y. (2018). Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in neural information processing systems* (pp. 8157–8166).

Li, F., Zhang, B., & Liu, B. (2016). Ternary weight networks. arXiv preprint arXiv:1605.04711.

Liu, X., Duh, K., Liu, L., & Gao, J. (2020). Very deep transformers for neural machine translation. arXiv preprint arXiv:2008.07772.

Meller, E., Finkelstein, A., Almog, U., & Grobman, M. (2019). Same, same but different-recovering neural network quantization error through weight factorization. arXiv preprint arXiv:1902.01917.

Mishkin, D., & Matas, J. (2015). All you need is a good init. arXiv preprint arXiv:1511.06422.

Mishra, A., & Marr, D. (2017). Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. arXiv preprint arXiv:1711.05852.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., & Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems* (pp. 8026–8037).

Petersen, P., & Voigtlaender, F. (2018). Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, *108*, 296–330.

Phan, H., David, W., Zafar, & Song, H. (2020). PyTorch CIFAR10 github repository. URL https://github.com/huyvnphan/PyTorch_CIFAR10.

Polino, A., Pascanu, R., & Alistarh, D. (2018). Model compression via distillation and quantization. arXiv preprint arXiv:1802.05668.

Saxe, A. M., McClelland, J. L., & Ganguli, S. (2013). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv preprint arXiv:1312.6120.

Shapiro, A., Dentcheva, D., & Ruszczyński, A. (2014). *Lectures on stochastic programming: modeling and theory*. SIAM.

Tann, H., Hashemi, S., Bahar, R. I., & Reda, S. (2017). Hardware-software codesign of accurate, multiplier-free deep neural networks. In *2017 54th ACM/EDAC/IEEE design automation conference* (pp. 1–6). IEEE.

Tao, A., Sapra, K., & Catanzaro, B. (2020). Hierarchical multi-scale attention for semantic segmentation. arXiv preprint arXiv:2005.10821.

Tsai, C.-F., & Wu, J.-W. (2008). Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, *34*(4), 2639–2649.

Vershynin, R. (2018). *vol. 47, High-dimensional probability: an introduction with applications in data science*. Cambridge University Press.

Wang, G., Giannakis, G. B., & Chen, J. (2019). Learning ReLU networks on linearly separable data: Algorithm, optimality, and generalization. *IEEE Transactions on Signal Processing*, *67*(9), 2357–2370.

Wu, J., Leng, C., Wang, Y., Hu, Q., & Cheng, J. (2016). Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4820–4828).

Xiao, L., Bahri, Y., Sohl-Dickstein, J., Schoenholz, S. S., & Pennington, J. (2018). Dynamical isometry and a mean field theory of cnns: How to train 10,000-layer vanilla convolutional neural networks. arXiv preprint arXiv:1806.05393.

Yin, P., Lyu, J., Zhang, S., Osher, S., Qi, Y., & Xin, J. (2019). Understanding straight-through estimator in training activation quantized neural nets. arXiv preprint arXiv:1903.05662.

Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Zhang, Z., Lin, H., et al. (2020). Resnest: Split-attention networks. arXiv preprint arXiv:2004.08955.

Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6848–6856).

Zhao, L., & Tsai, R. (2015). Locking and unlocking a mobile device using facial recognition. US Patent 8, 994, 499.

Zhou, S., Wu, Y., Ni, Z., Zhou, X., Wen, H., & Zou, Y. (2016). Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. arXiv preprint arXiv:1606.06160.