Journal of Machine Learning Research 26 (2025) 1-30 Submitted 2/24; Revised 9/24; Published 2/25

Local Linear Recovery Guarantee of Deep Neural Networks at Overparameterization

Yaoyu Zhang[a,b*] ZHYY.SJTU@SJTU.EDU.CN
Leyang Zhang[c] LEYANGZ_HAWK@OUTLOOK.COM
Zhongwang Zhang[a] 0123ZZW666@SJTU.EDU.CN
Zhiwei Bai[a] BAI299@SJTU.EDU.CN
[a] *School of Mathematical Sciences, Institute of Natural Sciences and MOE-LSC, Shanghai Jiao Tong University, Shanghai, 200240, China*
[b] *School of Artificial Intelligence, Shanghai Jiao Tong University, Shanghai, 200240, China*
[c] *School of Mathematics, Georgia Institute of Technology, Atlanta, GA, 30332, United States*

Editor: Joan Bruna

Abstract

Determining whether deep neural network (DNN) models can reliably recover target functions at overparameterization is a critical yet complex issue in the theory of deep learning. To advance understanding in this area, we introduce a concept we term "local linear recovery" (LLR), a weaker form of target function recovery that renders the problem more amenable to theoretical analysis. In the sense of LLR, we prove that functions expressible by narrower DNNs are guaranteed to be recoverable from fewer samples than model parameters. Specifically, we establish upper limits on the optimistic sample sizes, defined as the smallest sample size necessary to guarantee LLR, for functions in the space of a given DNN. Furthermore, we prove that these upper bounds are achieved in the case of two-layer tanh neural networks. Our research lays a solid groundwork for future investigations into the recovery capabilities of DNNs in overparameterized scenarios.

Keywords: deep learning theory, recovery at overparameterization, local linear recovery guarantee, optimistic sample size

1. Introduction

Determining the requisite number of data points for a model to recover a target function is a fundamental problem in regression. This issue is particularly pertinent for deep neural network (DNN) models, where the question arises: Can these models recover target functions using fewer data points than their parameter count, thereby operating effectively in an overparameterized regime? Traditional wisdom suggests that a linear model with M parameters typically needs M data points to reconstruct a linear function. Particularly, in band-limited signal recovery, the Nyquist-Shannon sampling theorem posits that a periodic signal with a maximum frequency of f (represented by $M = 2f$ coefficients) can be perfectly reconstructed from $n \geqslant 2f$ uniformly sampled points (Shannon, 1984). These precedents imply that linear models lack recovery guarantee when overparameterized. In contrast, nonlinear DNN models have been empirically shown to generalize effectively even when

*. Corresponding author.

overparameterized (Zhang et al., 2017). Subsequent experiments reinforce this observation, indicating that DNNs can achieve zero generalization error in recovering target functions under overparameterization (Zhang et al., 2023b). This study delves into the theoretical underpinnings of target function recovery for DNNs, with the goal of establishing a recovery guarantee in overparameterized scenarios.

Within the spectrum of recovery guarantees, the global recovery guarantee—ensuring target recovery through comprehensive training—is the most robust and practically relevant, especially in overparameterized contexts. However, the intricate and highly nonlinear training dynamics of DNNs render this global guarantee theoretically elusive. To navigate this complexity, we propose a more attainable goal: establishing a weaker form of recovery guarantee for DNNs in overparameterized conditions. Specifically, we incorporates two simplifications to achieve a theoretically tractable weaker form of recovery, namely the local linear recovery (LLR) guarantee: (i) *localization*: focusing on recovery in the vicinity of an optimal point within the parameter space; and (ii) *linearization*: considering the recovery of the model linearized at the optimal point. While local linear fitting of data is not practically feasible in general, the LLR-guarantee sheds light on the optimistic (i.e., best-possible) performance for global recovery of DNNs, suggesting that without an LLR-guarantee at overparameterization, stronger forms of recovery are unlikely.

Our main contributions are threefold: (i) We introduce the LLR-guarantee and formulate its theoretical framework for both differentiable and analytic models, demonstrating that linear models do not possess an LLR-guarantee when overparameterized. (ii) Employing the Embedding Principle (Zhang et al., 2021b, 2022), we derive upper bounds for the optimistic sample sizes—defined as the smallest sample sizes that ensure an LLR guarantee—for general DNNs. These bounds affirm the LLR-guarantee at overparameterization for all functions expressible by narrower DNNs. (iii) We pinpoint the exact optimistic sample sizes for two-layer fully-connected and convolutional tanh neural networks, which meet their respective upper bounds, thereby illustrating the exactitude of our upper bounds.

2. Related Works

In our research, we introduce the concept of *optimistic sample size*, which quantifies the minimum number of training samples necessary to recover a target function under the best possible conditions. This approach differs significantly from traditional sample complexity analyses, which typically estimate the sample requirements to achieve a specified performance level in worst-case scenarios (Shalev-Shwartz and Ben-David, 2014). For example, Zhong et al. (2017) estimate the sample complexity for recovering a two-layer neural network based on the condition of local strong convexity around the ground truth, which does not hold under overparameterization. Recognizing that deep neural networks (DNNs) often perform substantially better in practice than theoretical worst-case predictions suggest (Zhang et al., 2017, 2021a), our LLR analysis pioneers a framework for estimating sample sizes under the best possible conditions. Furthermore, our empirical findings reveal that the practical performance of a finely-tuned DNN can approach this optimistic threshold, even in scenarios of substantial overparameterization. Moreover, as demonstrated in Zhang and Xu (2024), techniques such as dropout can further enhance the network's performance to recover target functions.

Our theoretical LLR framework, which is predicated on the linearization of DNNs, stands in stark contrast to linear analyses based on the NTK/lazy training/linear regime (Jacot et al., 2018; Arora et al., 2019; Chizat et al., 2019; Luo et al., 2021). The main differences are as follows: (i) NTK analysis linearizes around a random initial point, whereas LLR analysis considers linearization around a target point, i.e., a global minimizer with zero generalization error. (ii) NTK pertains to the linear training behavior of DNNs with large initial weights, whereas the optimistic sample sizes from LLR analysis are empirically linked to the performance of nonlinear training dynamics exhibited at small initializations. According to the phase diagrams in Refs. (Luo et al., 2021; Zhou et al., 2022a), NTK analysis corresponds to the linear regimes, while our LLR results associate to the condensed regimes, where nonlinear condensation dynamics are instrumental in achieving near-optimism performance.

The discovery of the embedding principle (Zhang et al., 2021b, 2022; Fukumizu et al., 2019; Simsek et al., 2021; Bai et al., 2024) has shed light on the analysis of critical points within the loss landscapes of DNNs and has forged connections between the loss landscapes of networks of varying widths. Notably, Zhang et al. (2022) introduces a comprehensive suite of critical embedding operators that map the parameter space of a narrower DNN to that of a wider one while preserving both the output function and the criticality of the network. This powerful analytical tool further illuminates the hierarchical model rank structure inherent in DNNs. Leveraging these critical embeddings, which inherently maintain the model rank, we derive in this work an upper bound for the optimistic sample size applicable to general DNNs.

Remark that the notion of rank, which is used to capture the intrinsic complexity of a mapping, plays a fundamental role in linear algebra, differential topology and other areas. In the study of deep learning, Jacot (2023) introduces notions of rank for nonlinear functions, i.e., Jacobian rank and bottleneck rank, to study the implicit bias of large depth networks. These notions characterize the complexity of model outputs varies over the input space, whereas our model rank characterizes the complexity of model function varies over the parameter space. Whether these apparently different notions of rank are related to each other for DNNs remains an interesting problem for the future study.

3. Theory of Local Linear Recovery

3.1 Assumptions and Definitions

Assumption.
(i) We consider differentiable (w.r.t. both inputs and parameters) models with 1*-d output* $f_{\boldsymbol{\theta}}(\cdot) = f(\cdot;\boldsymbol{\theta}) : \mathbb{R}^d \times \mathbb{R}^M \to \mathbb{R}$.
(ii) We consider target functions f^* *expressible by the model* $f_{\boldsymbol{\theta}}$*, i.e.,* $f^* \in \mathcal{F} := \{f_{\boldsymbol{\theta}}(\cdot)|\boldsymbol{\theta} \in \mathbb{R}^M\}$*. And the training data* $S = \{(\boldsymbol{x}_i \in \mathbb{R}^d, y_i = f^*(\boldsymbol{x}_i) \in \mathbb{R})\}_{i=1}^n$ *is sampled from* f^*.
(iii) The loss function $\ell(\cdot,\cdot)$ *is a continuously differentiable distance function and the empirical loss* $L_S(\boldsymbol{\theta}) = \mathbb{E}_S \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = \frac{1}{n}\sum_{i=1}^n \ell(f(\boldsymbol{x}_i;\boldsymbol{\theta}), y_i)$.

Under the above general assumptions, this work focuses on the recovery of target function f^* for the following regression problem when $n \leqslant M$:

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^M} \frac{1}{n}\sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), f^*(\boldsymbol{x}_i)). \tag{1}$$

We say f^* is successfully recovered when the solution $\boldsymbol{\theta}^*$ obtained by an algorithm satisfies $f_{\boldsymbol{\theta}^*} = f^*$. For convenience, we define the target set $\mathcal{M}_{f^*}$ as follows, by which f^* is recovered when $\boldsymbol{\theta}^* \in \mathcal{M}_{f^*}$.

Definition 1 (target set)**.** *Suppose we have a model $f_{\boldsymbol{\theta}}(\cdot) = f(\cdot;\boldsymbol{\theta}) : \mathbb{R}^d \times \mathbb{R}^M \to \mathbb{R}$ with model function space $\mathcal{F} = \{f_{\boldsymbol{\theta}}(\cdot)|\boldsymbol{\theta} \in \mathbb{R}^M\}$. For any target function $f' \in \mathcal{F}$, the target set $\mathcal{M}_{f'} := \{\boldsymbol{\theta}|f(\cdot;\boldsymbol{\theta}) = f'\}$.*

We note that analyzing the recovery of f^* at overparameterization through global training presents significant difficulties due to two primary challenges: (i) the optimization challenge, where it remains uncertain whether the training process can avoid local minima and saddle points to converge to a global minimum (Sun et al., 2020); (ii) the challenge of infinite solutions, where the target set $\mathcal{M}_{f^*}$ is embedded within an approximately $(M-n)$-dimensional manifold of global minima with complex geometry (Cooper, 2021; Zhang et al., 2023a), making it difficult to ascertain if global training can precisely reach $\mathcal{M}_{f^*}$. To bypass these obstacles, we introduce a more theoretically tractable variant of recovery as follows.

Definition 2 (local linear recovery (LLR) guarantee)**.** *Suppose we have a differentiable model $f_{\boldsymbol{\theta}}(\cdot)$ with M parameters, a target function $f^* \in \mathcal{F} := \{f_{\boldsymbol{\theta}}(\cdot)\}$ and training data $S = \{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i))\}_{i=1}^n$.*
(a) ***LLR-guarantee:*** *We say f^* has local linear recovery (LLR) guarantee (by model $f_{\boldsymbol{\theta}}$ from S) if the following condition holds: There exists $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$ such that*

$$\{f^*\} = \operatorname{argmin}_{g\in\widetilde{\mathcal{T}}_{\boldsymbol{\theta}'}} \frac{1}{n}\sum_{i=1}^{n} \ell(g(\boldsymbol{x}_i), f^*(\boldsymbol{x}_i)), \tag{2}$$

where $\widetilde{\mathcal{T}}_{\boldsymbol{\theta}'} = \{f(\cdot;\boldsymbol{\theta}') + \boldsymbol{a}^T\nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}')|\boldsymbol{a} \in \mathbb{R}^M\}$ is the tangent function hyperplane at $\boldsymbol{\theta}'$. If the above condition holds, we say f^* has LLR-guarantee at $\boldsymbol{\theta}'$ (by model $f_{\boldsymbol{\theta}}$ from S).
(b) ***n-sample LLR-guarantee:*** *We say a function $f^* \in \mathcal{F}$ has n-sample LLR-guarantee if there exists a n-sample data set $S = \{(\boldsymbol{x}_i \in \mathbb{R}^d, f^*(\boldsymbol{x}_i) \in \mathbb{R})\}_{i=1}^n$ such that f^* has LLR-guarantee. Furthermore, if $n < M$, then we say f^* has LLR-guarantee at overparameterization.*
(c) ***n-sample LLR-guarantee a.e.:*** *We say a function $f^* \in \mathcal{F}$ has n-sample LLR-guarantee a.e. (almost everywhere) if the following condition holds: For inputs $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$ a.e. with respect to $\mathcal{L}^{d\times n}$ Lebesgue measure, f^* has LLR-guarantee from $S = \{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i))\}_{i=1}^n$.*

Remark 1 (LLR-guarantee vs. LLR-guarantee a.e.)**.** *(i) If a certain result holds for inputs X almost everywhere (a.e.) with respect to the Lebesgue measure $\mathcal{L}^{d\times n}$, then it holds with probability 1 for any probability measure that is absolutely continuous with respect to $\mathcal{L}^{d\times n}$,*

e.g., a non-degenerate gaussian probability measure (See Remark 3 for more detailed discussion). (ii) Although n-sample LLR-guarantee seems to be significantly weaker than n-sample LLR-guarantee a.e., we will show later that any n-sample LLR-guarantee automatically upgrades to n-sample LLR-guarantee a.e. for analytic models, e.g., neural networks with tanh, sigmoid or GELU activation. For the pursuit of generality, we mainly focus on the n-sample LLR-guarantee in this work. (iii) In general, n-sample LLR-guarantee informs the potential of recovering the target from n samples in practice, whereas no n-sample LLR-guarantee is a strong indication that n samples are not sufficient to recover the target.

Proposition 1. *Suppose we have a differentiable model $f_{\boldsymbol{\theta}}(\cdot)$ with M parameters. For any target function $f^* \in \mathcal{F}$, if it has n-sample LLR-guarantee, then it has n'-sample LLR-guarantee for any $n' \geqslant n$.*

Proof When f^* has n-sample LLR-guarantee, there exists $S = \{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i))\}_{i=1}^n$ and $\boldsymbol{\theta}^* \in \mathcal{M}_{f^*}$ such that the solution set

$$\phi_S^* := \operatorname{argmin}_{f \in \widetilde{\mathcal{T}}_{\boldsymbol{\theta}^*}} \mathbb{E}_{(\boldsymbol{x},y)\in S} \ell(f(\boldsymbol{x}), y) = \{f^*\},$$

where $\mathbb{E}_{(\boldsymbol{x},y)\in S} \ell(f(\boldsymbol{x}), y) = \frac{1}{n}\sum_{i=1}^n \ell(f(\boldsymbol{x}_i), f^*(\boldsymbol{x}_i))$. For $S' = S \cup \{(\boldsymbol{x}_i', f^*(\boldsymbol{x}_i'))\}_{i=n+1}^{n'}$ with any $\boldsymbol{x}_i' \in \mathbb{R}^M$ for $i \in [n+1 : n']$, let

$$\phi_{S'}^* := \operatorname{argmin}_{f \in \widetilde{\mathcal{T}}_{\boldsymbol{\theta}^*}} \mathbb{E}_{(\boldsymbol{x},y)\in S'} \ell(f(\boldsymbol{x}), y).$$

Obviously, $f^* \in \phi_{S'}^*$, by which the empirical loss attains 0 on S'. Then, any global minimizer $f \in \phi_{S'}^*$ also attains 0 empirical loss on S', thus 0 empirical loss on S. Therefore, $\phi_{S'}^* \subset \phi_S^* = \{f^*\}$, which yields

$$\phi_{S'}^* = \phi_S^* = \{f^*\}.$$

By Definition 2, f^* has n'-sample LLR-guarantee. ■

Proposition. 1 signifies the importance of understanding the minimum sample size with LLR-guarantee for a target function, which is rigorously defined below as an optimistic sample size.

Definition 3 (optimistic sample size)**.** *Suppose we have a differentiable model $f_{\boldsymbol{\theta}}(\cdot)$. For any function $f^* \in \mathcal{F}$, if f^* has LLR-guarantee from n samples but not $n-1$ samples, then its optimistic sample size*

$$O_{f_{\boldsymbol{\theta}}}(f^*) = n.$$

3.2 General LLR Theory

In this section, we present the theoretical results of LLR for the regression problem of general differentiable models. In particular, we establish the quantitative relation between the LLR-guarantee and the model rank, by which estimating the optimistic sample size of a target converts to an estimation of its model rank.

Definition 4 (model rank). *Given any differentiable (in parameters) model $f_{\boldsymbol{\theta}}$, the model rank for any $\boldsymbol{\theta}^* \in \mathbb{R}^M$ is defined as*

$$\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*) := \dim\left(\operatorname{span}\{\partial_{\theta_i} f(\cdot;\boldsymbol{\theta}^*)\}_{i=1}^M\right), \tag{3}$$

where $\operatorname{span}\{\phi_i(\cdot)\}_{i=1}^M = \{\sum_{i=1}^M a_i\phi_i(\cdot)|a_1,\cdots,a_M \in \mathbb{R}\}$ *and* $\dim(\cdot)$ *returns the dimension of a linear function space.*

Remark 2 (significance of model rank in neural networks). *Quantifying the extent of condensation in neural networks—a phenomenon where neurons tend to cluster together (Luo et al., 2021; Zhou et al., 2022b) (also known as weight quantization (Maennel et al., 2018) and weight clustering (Brutzkus and Globerson, 2019))—is crucial for understanding nonlinear training dynamics and the implicit bias of neural networks. We demonstrate through a simple example that the model rank effectively quantifies condensation: lower model rank indicates stronger condensation. Consider a simple two-neuron neural network defined as $f_{\boldsymbol{\theta}}(x) = a_1\tanh(w_1x) + a_2\tanh(w_2x)$. The model rank is given by*

$$\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}) = \dim\left(\operatorname{span}\left\{\tanh(w_1x),\ a_1\tanh'(w_1x)x,\ \tanh(w_2x),\ a_2\tanh'(w_2x)x\right\}\right).$$

Intuitively, condensation occurs when $w_2 = \pm w_1$, allowing the two neurons to be effectively combined into one. Under this condition, the model rank satisfies $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \leqslant 2$. Conversely, when $w_2 \neq \pm w_1 \neq 0$ and $a_1, a_2 \neq 0$, condensation does not occur since the two neurons cannot be combined into one, and the model rank is $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}) = 4$. Therefore, the phenomenon of condensation implies that neural networks tend to learn functions with lower model rank via nonlinear training.

Definition 5 (empirical tangent matrix and empirical model rank). *Given any differentiable model $f_{\boldsymbol{\theta}}$ and training data $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, at any parameter point $\boldsymbol{\theta}^*$, $\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*) = [\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_1;\boldsymbol{\theta}^*),\cdots,\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_n;\boldsymbol{\theta}^*)]$ is referred to as the empirical tangent matrix. Then the empirical model rank is defined as follows*

$$\operatorname{rank}_S(\boldsymbol{\theta}^*) = \operatorname{rank}(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)).$$

Lemma 1 (LLR condition). *f^* has local linear recovery guarantee at $\boldsymbol{\theta}^*$ by model $f_{\boldsymbol{\theta}}$ from S if and only if $\operatorname{rank}_S(\boldsymbol{\theta}^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*)$.*

Proof Define

$$\tilde{R}_S(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^n \ell(f(\boldsymbol{x}_i,\boldsymbol{\theta}^*) + \boldsymbol{a}^{\mathrm{T}}\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_i;\boldsymbol{\theta}^*), f^*(\boldsymbol{x}_i)).$$

First assume that f^* has LLR guarantee at $\boldsymbol{\theta}^*$ by model $f_{\boldsymbol{\theta}}$ from S. Then $f(\cdot;\boldsymbol{\theta}^*) = f^*$ by Definition 2, and this function is unique. This uniqueness implies that for any $\boldsymbol{a} \in \ker\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)$ we must have

$$f(\cdot;\boldsymbol{\theta}^*) + \boldsymbol{a}^{\mathrm{T}}\nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}^*) = f^*.$$

Equivalently, $\boldsymbol{a} \in \ker\nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}^*)$. This shows $\ker\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*) \subseteq \ker\nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}^*)$. But clearly $\ker\nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}^*) \subseteq \ker\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)$, so the two kernels are equal. It follows that

$$\operatorname{rank}_S(\boldsymbol{\theta}^*) = \operatorname{rank}\left(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)\right) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*)$$

Conversely, assume that $\text{rank}_S(\boldsymbol{\theta}^*) = \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*)$. Then, similar as above, we have $\ker \nabla_{\boldsymbol{\theta}} f(\cdot;\boldsymbol{\theta}^*) = \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)$. Because ℓ is a distance function, for any $\boldsymbol{a} \in \mathbb{R}^M$ with $\tilde{R}_S(\boldsymbol{a}) = R_S(\boldsymbol{\theta}^*) = 0$, we must have

$$\boldsymbol{a}^{\mathrm{T}} \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_i;\boldsymbol{\theta}^*) = 0, \quad \forall\, 1 \leqslant i \leqslant n.$$

Equivalently, $\boldsymbol{a} \in \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)$ and thus $\boldsymbol{a}^{\mathrm{T}} f(\cdot;\boldsymbol{\theta}^*) = 0$. This shows

$$\{f^*\} = \{f(\cdot;\boldsymbol{\theta}^*)\} = \text{argmin}_{g \in \widetilde{\mathcal{T}}_{\boldsymbol{\theta}'}} \frac{1}{n} \sum_{i=1}^{n} \ell(g(\boldsymbol{x}_i), f^*(\boldsymbol{x}_i))$$

is well-defined, i.e., f^* has LLR guarantee at $\boldsymbol{\theta}^*$.

■

Corollary 1 (phase transition of LLR-guarantee at a target point)**.** *For any $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$, if training data size $n < \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$, f^* has no local linear recovery guarantee at $\boldsymbol{\theta}'$. Otherwise, if $n \geqslant \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$, f^* has n-sample LLR-guarantee, i.e., there exists an n-sample data set $S' = \{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i))\}_{i=1}^n$ such that f^* has local linear recovery guarantee at $\boldsymbol{\theta}'$.*

Proof For $n < \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$, $\text{rank}_S(\boldsymbol{\theta}') \leqslant n < \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$. By Lemma 1, f^* has no local linear recovery guarantee at $\boldsymbol{\theta}'$. On the other hand, for $n \geqslant \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$, we claim that there exist $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)$ with $\text{rank}\,(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)) = \text{rank}_{f_{\boldsymbol{\theta}}}(\cdot;\boldsymbol{\theta}')$. Suppose this is not true. Let

$$M' = \max_{\boldsymbol{X} \in \mathbb{R}^{d \times n}} \text{rank}\left(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}')\right)$$

and let $\boldsymbol{x}_1, ..., \boldsymbol{x}_{M'}$ be such that

$$\text{rank}\left[\nabla_{\boldsymbol{\theta}} f((\boldsymbol{x}_1, ..., \boldsymbol{x}_{M'});\boldsymbol{\theta}')\right] = M'.$$

So in particular $M' < \text{rank}_{f_{\boldsymbol{\theta}}}(\cdot;\boldsymbol{\theta}') \leqslant M$. Given $a \in \mathbb{R}^M$ such that $\nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_i;\boldsymbol{\theta}) = 0$ for all $1 \leq i \leq M'$, we must have $a \in \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}';\cdot)$. Therefore,

$$\ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}';\cdot) \subseteq \ker \nabla_{\boldsymbol{\theta}} f((\boldsymbol{x}_1, ..., \boldsymbol{x}_{M'});\boldsymbol{\theta}') \subseteq \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}';\cdot).$$

This means

$$\dim \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}';\cdot) = \dim \ker \nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}') = M - M'$$

and thus $\text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') = M'$, a contradiction. So there must exist $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)$ with $\text{rank}\,(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{X};\boldsymbol{\theta}^*)) = \text{rank}_{f_{\boldsymbol{\theta}}}(\cdot;\boldsymbol{\theta}')$. By Lemma 1, f^* has n-sample guarantee.

■

We address above the LLR-guarantee at a target point. In the following, we further address the LLR-guarantee for recovering a target function.

Definition 6 (model rank for function)**.** *The model rank for any function $f^* \in \mathcal{F}$ is defined as*

$$\text{rank}_{f_{\boldsymbol{\theta}}}(f^*) := \min_{\boldsymbol{\theta}' \in \mathcal{M}_{f^*}} \text{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}'). \tag{4}$$

Theorem 1 (optimistic sample size estimate)**.** *Suppose we have a differentiable model $f_{\boldsymbol{\theta}}$. For any function $f^* \in \mathcal{F}$, we have*

$$O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*).$$

Proof When the sample size $n < \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*)$, for any $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$, $n < \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$. By Corollary 1, f^* has no LLR-guarantee at $\boldsymbol{\theta}'$. Therefore, f^* has no n-sample LLR-guarantee. When the sample size $n \geqslant \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*)$, there exists $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$, such that $n \geqslant \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$. By Corollary 1, f^* has n-sample LLR-guarantee. By Definition 3, we have

$$O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*).$$

■

The subsequent two corollaries provide a rigorous formulation of two intuitive assertions about the best-possible cases of regression problem (1): (i) M samples are sufficient for recovery; (ii) linear models generally cannot be recovered at overparameterization.

Corollary 2 (generic upper bound for optimistic sample size)**.** *Suppose we have a differentiable model $f_{\boldsymbol{\theta}}$. For any function $f^* \in \mathcal{F}$, we have*

$$O_{f_{\boldsymbol{\theta}}}(f^*) \leqslant M.$$

Proof Because $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \leqslant M$ for any $\boldsymbol{\theta} \in \mathbb{R}^M$, $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) \leqslant M$ for any $f^* \in \mathcal{F}$. Therefore, by Theorem 1, $O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) \leqslant M$. ■

Corollary 3 (no LLR-guarantee for linear models at overparameterization)**.** *For a linear model $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{i=1}^{M} \theta_i \phi_i(\boldsymbol{x})$ with $\boldsymbol{\theta} = [\theta_1, \cdots, \theta_M]^T$, if its basis functions are linearly independent, i.e.,* $\dim(\operatorname{span}\{\phi_i(\cdot)\}_{i=1}^{M}) = M$*, then*

$$O_{f_{\boldsymbol{\theta}}}(f^*) \equiv M,$$

i.e., none of the functions in the model function space $\mathcal{F}$ has LLR-guarantee at overparameterization.

Proof For any $\boldsymbol{\theta}^* \in \mathbb{R}^M$, we have

$$\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*) = \dim\left(\operatorname{span}\left\{\partial_{\theta_i} f(\cdot;\boldsymbol{\theta}^*)\right\}_{i=1}^{M}\right) = \dim\left(\operatorname{span}\left\{\phi_i(\boldsymbol{x})\right\}_{i=1}^{M}\right) = M.$$

Therefore, $O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) = M$ for any $f^* \in \mathcal{F}$. ■

3.3 LLR-Guarantee for Analytic Models

In this subsection, we exploit the characteristics of analytic functions to enhance the LLR guarantee, extending it to an almost everywhere (a.e.) guarantee for analytic models.

Lemma 2. *Given m linearly independent analytic functions $\phi_1(\boldsymbol{x}), \cdots, \phi_m(\boldsymbol{x})$ with $\phi_i : \mathbb{R}^d \to \mathbb{R}$ for all $i \in [m]$, $\operatorname{rank}(\boldsymbol{\Phi}(\boldsymbol{X})) = m$ almost everywhere (a.e.) with respect to $\mathcal{L}^{d\times m}$ Lebesgue measure, where*

$$\boldsymbol{\Phi}(\boldsymbol{X}) := \begin{bmatrix} \phi_1(\boldsymbol{x}_1) & \dots & \phi_m(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_1(\boldsymbol{x}_m) & \dots & \phi_m(\boldsymbol{x}_m) \end{bmatrix}.$$

Proof Clearly, $\det(\boldsymbol{\Phi}(\cdot)) : \mathbb{R}^{d\times m} \to \mathbb{R}$ is an analytic function over $\mathbb{R}^{d\times m}$. In addition, because $\{\phi_i\}_{i=1}^m$ are linearly independent, there exists $\boldsymbol{X} \in \mathbb{R}^{d\times m}$ such that $\det(\boldsymbol{\Phi}(\cdot)) \neq 0$, i.e., $\det(\boldsymbol{\Phi}(\cdot))$ is not constant zero. By the property of real analytic function (Mityagin, 2020), $\operatorname{rank}(\boldsymbol{\Phi}(\boldsymbol{X})) = m$ a.e. with respect to $\mathcal{L}^{d\times m}$ Lebesgue measure. ■

Remark 3 (zero set of a real analytic function)**.** *In the above proof, we utilize the fact that the zero set of a nontrivial real analytic function on $\mathbb{R}^d$ has Lebesgue measure zero (Mityagin, 2020). Specifically, a set $C \subset \mathbb{R}^d$ has Lebesgue measure zero if, for any $\epsilon > 0$, there exists a countable collection of balls whose total volume is less than ϵ that covers C. This property is significant because it implies that C occupies no volume in the space, despite potentially being infinite or uncountable. Furthermore, this result has important implications in probability theory. If a set has Lebesgue measure zero, then any probability measure that is absolutely continuous with respect to the Lebesgue measure will assign a probability of zero to that set. For example, if we consider a non-degenerate gaussian measure, the zero set of a real analytic function will have probability zero.*

Corollary 4. *Given m analytic functions $\phi_1(\boldsymbol{x}), \cdots, \phi_m(\boldsymbol{x})$ with $\phi_i : \mathbb{R}^d \to \mathbb{R}$ for all $i \in [m]$ and $\dim(\operatorname{span}(\{\phi_i(\cdot)\}_{i=1}^m)) = r$, $\operatorname{rank}(\boldsymbol{\Phi}(\boldsymbol{X})) = \min\{n, r\}$ a.e. with respect to $\mathcal{L}^{d\times n}$ Lebesgue measure.*

Proof It is obvious that $\operatorname{rank}(\boldsymbol{\Phi}(\boldsymbol{X})) \leqslant \min\{n, r\}$. For $n \leqslant r$, we can always pick n independent functions from $\{\phi_i(\cdot)\}_{i=1}^m$. By Lemma 2, $\boldsymbol{\Phi}(\boldsymbol{X})$ has a rank-n submatrix of $\boldsymbol{\Phi}(\boldsymbol{X})$ a.e. with respect to Lebesgue measure. For $n > r$, we have that the submatrix of the first r rows of $\boldsymbol{\Phi}(\boldsymbol{X})$ has rank r a.e. by Lemma 2. Therefore, $\operatorname{rank}(\boldsymbol{\Phi}(\boldsymbol{X})) = \min\{n, r\}$ a.e. with respect to $\mathcal{L}^{d\times n}$ Lebesgue measure. ■

Proposition 2 (LLR-guarantee a.e. for analytic models)**.** *Given any analytic model $f_{\boldsymbol{\theta}}$ (w.r.t. both inputs and parameters), if target function $f^* \in \mathcal{F}$ has n-sample LLR-guarantee, then f^* has n-sample LLR-guarantee a.e.*

Proof If target function $f^* \in \mathcal{F}$ has n-sample LLR-guarantee, by Definition 3, $n \geqslant O_{f_{\boldsymbol{\theta}}}(f^*)$. In addition, there exists $\boldsymbol{\theta}^* \in \mathcal{M}_{f^*}$ such that $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*) = O_{f_{\boldsymbol{\theta}}}(f^*)$. Then by Corollary 4, for $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n]$ a.e. with respect to $\mathcal{L}^{d\times n}$ Lebesgue measure with data set $S = \{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i))\}_{i=1}^n$, $\operatorname{rank}_S(\boldsymbol{\theta}^*) = \min\{\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*), n\} = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*)$. By the LLR condition Lemma 1, f^* has n-sample LLR-guarantee a.e. ■

4. Upper Bounds of Optimistic Sample Sizes for DNNs

Building on the LLR theory, the task of estimating optimistic sample sizes for target functions necessitates the elucidation of the model rank across the space of functions. This task is notably challenging, particularly for DNNs with three or more layers, due to the complexity inherent in disentangling the linear dependencies among the compositional tangent functions. Fortunately, we observe that the critical embedding operators introduced in Refs. (Zhang et al., 2021b, 2022) can preserve the tangent function space and the model rank when transitioning from a parameter point in a narrower DNN to one in a wider DNN. This observation allows us to constrain the model rank of a target function within a wide DNN by referencing the smallest DNN capable of representing it regardless of the depth. The formal mathematical exposition of these results is provided below.

4.1 DNNs and the Embedding Principle

In this subsection, we briefly recapitulate the key elements of Embedding Principle in Refs. (Zhang et al., 2021b, 2022). Consider L-layer ($L \geq 2$) fully-connected DNNs with a general differentiable activation function. We regard the input as the 0-th layer and the output as the L-th layer. Let m_l be the number of neurons in the l-th layer. In particular, $m_0 = d$ and $m_L = d'$. For any $i, k \in \mathbb{N}$ and $i < k$, we denote $[i : k] = \{i, i+1, \ldots, k\}$. In particular, we denote $[k] := \{1, 2, \ldots, k\}$. Given weights $W^{[l]} \in \mathbb{R}^{m_l \times m_{l-1}}$ and bias $b^{[l]} \in \mathbb{R}^{m_l}$ for $l \in [L]$, we define the collection of parameters $\boldsymbol{\theta}$ as a $2L$-tuple (an ordered list of $2L$ elements) whose elements are matrices or vectors

$$\boldsymbol{\theta} = \Big(\boldsymbol{\theta}|_1, \cdots, \boldsymbol{\theta}|_L\Big) = \Big(\boldsymbol{W}^{[1]}, \boldsymbol{b}^{[1]}, \ldots, \boldsymbol{W}^{[L]}, \boldsymbol{b}^{[L]}\Big). \tag{5}$$

where the l-th layer parameters of $\boldsymbol{\theta}$ is the ordered pair $\boldsymbol{\theta}|_l = \Big(\boldsymbol{W}^{[l]}, \boldsymbol{b}^{[l]}\Big), \quad l \in [L]$. We may misuse the notation and identify $\boldsymbol{\theta}$ with its vectorization $\mathrm{vec}(\boldsymbol{\theta}) \in \mathbb{R}^M$ with $M = \sum_{l=0}^{L-1}(m_l + 1)m_{l+1}$.

Given $\boldsymbol{\theta} \in \mathbb{R}^M$, the neural network function $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$ is defined recursively. First, we write $\boldsymbol{f}_{\boldsymbol{\theta}}^{[0]}(\boldsymbol{x}) = \boldsymbol{x}$ for all $\boldsymbol{x} \in \mathbb{R}^d$. Then for $l \in [L-1]$, $\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]}$ is defined recursively as $\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]}(\boldsymbol{x}) = \sigma(\boldsymbol{W}^{[l]}\boldsymbol{f}_{\boldsymbol{\theta}}^{[l-1]}(\boldsymbol{x}) + \boldsymbol{b}^{[l]})$. Finally, we denote

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{f}_{\boldsymbol{\theta}}^{[L]}(\boldsymbol{x}) = \boldsymbol{W}^{[L]}\boldsymbol{f}_{\boldsymbol{\theta}}^{[L-1]}(\boldsymbol{x}) + \boldsymbol{b}^{[L]}. \tag{6}$$

For notational simplicity, we may drop the subscript $\boldsymbol{\theta}$ in $\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]}$, $l \in [0 : L]$.

We formally define the notion of wider/narrower as follows.

Definition 7 (wider/narrower DNN)**.** *We write* $\mathrm{NN}(\{m_l\}_{l=0}^L)$ *for a fully-connected neural network with width* $(m_0, \ldots, m_L)$*. Given two* L*-layer* ($L \geq 2$) *fully-connected neural networks* $\mathrm{NN}(\{m_l\}_{l=0}^L)$ *and* $\mathrm{NN}'(\{m_l'\}_{l=0}^L)$*, if* $m_0' = m_0$*,* $m_L' = m_L$*, and for any* $l \in [L-1]$*,* $m_l' \geq m_l$ *and* $K = \sum_{l=1}^{L-1}(m_l' - m_l) > 0$*, then we say that* $\mathrm{NN}'(\{m_l'\}_{l=0}^L)$ *is wider or* K*-neuron wider than* $\mathrm{NN}(\{m_l\}_{l=0}^L)$ *and* $\mathrm{NN}(\{m_l\}_{l=0}^L)$ *is narrower or* K*-neuron narrower than* $\mathrm{NN}'(\{m_l'\}_{l=0}^L)$*.*

Theorem 2 (Embedding Principle, Theorem 4.2 in Ref. (Zhang et al., 2022))**.** *Given any NN and any* K*-neuron wider NN, there exists a* K*-step composition embedding* $\mathcal{P}$ *satisfying*

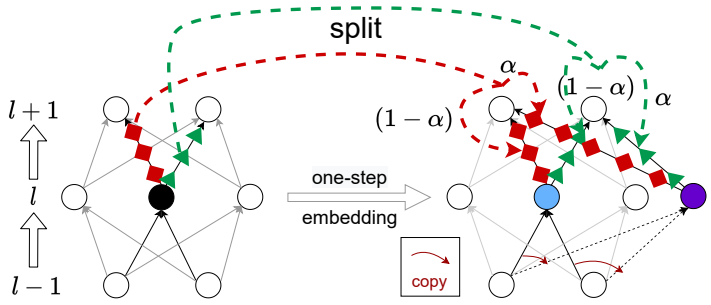


Figure 1: (Figure 2 in Zhang et al. (2021b)) Illustration of one-step splitting embedding. The black neuron in the left network is split into the blue and purple neurons in the right network. The red (green) output weight of the black neuron in the left net is splitted into two red (green) weights in the right net with ratio α and $(1-\alpha)$, respectively.

that: For any given data S, loss function $\ell(\cdot,\cdot)$, activation function $\sigma(\cdot)$, given any critical point $\boldsymbol{\theta}^{\rm c}_{\rm narr}$ of the narrower NN, $\boldsymbol{\theta}^{\rm c}_{\rm wide} := \mathcal{P}(\boldsymbol{\theta}^{\rm c}_{\rm narr})$ is still a critical point of the K-neuron wider NN with the same output function, i.e., $\boldsymbol{f}_{\boldsymbol{\theta}^{\rm c}_{\rm narr}} = \boldsymbol{f}_{\boldsymbol{\theta}^{\rm c}_{\rm wide}}$.

The cornerstone of the Embedding Principle's proof lies in constructing a series of critical embeddings that map any given DNN to a wider one. Figure 1 illustrates a typical type of critical embedding, specifically the 'splitting' embedding. This process involves dividing a single neuron into two, while simultaneously preserving both the neural network's output function and its criticality. It is readily apparent that this splitting embedding can be seamlessly adapted to Convolutional Neural Networks (CNNs) and Residual Networks (ResNets), suggesting that the Embedding Principle is applicable to these architectures as well. Consequently, it is reasonable to infer that all subsequent results that depend on the presence of critical embeddings for fully-connected neural networks can be extended to CNNs and ResNets. Additionally, Zhang et al. (2022) describes other varieties of critical embeddings, such as the null embedding and the general compatible embedding.

4.2 Upper Bounding Optimistic Sample Size via Critical Mappings

In this subsection, we first provide a general definition of critical mappings, by which the previously proposed critical embeddings (see Zhang et al. (2022) Definition 4.2 for details) are special cases. Then, we prove Lemma 3, showing that uncovering critical mappings is an important means for obtaining an upper bound estimate of the optimistic sample size. This general result combined with the embedding principle of DNNs directly provides an upper bound estimate of optimistic sample sizes for general DNNs in Theorem 3. We also illustrate the upper bounds for a general depth-L DNN without bias terms in Table 1.

Definition 8 (critical mapping)**.** *Given differentiable model A $f_{\boldsymbol{\theta}_A} = f(\cdot;\boldsymbol{\theta}_A)$ with $\boldsymbol{\theta}_A \in \mathbb{R}^{M_A}$ and differentiable model B $g_{\boldsymbol{\theta}_B} = g(\cdot;\boldsymbol{\theta}_B)$ with $\boldsymbol{\theta}_B \in \mathbb{R}^{M_B}$, $\mathcal{P}: \mathbb{R}^{M_A} \to \mathbb{R}^{M_B}$ is a critical mapping from model A to B if given any $\boldsymbol{\theta} \in \mathbb{R}^{M_A}$, we have*
(i) output preserving: $f_{\boldsymbol{\theta}} = g_{\mathcal{P}(\boldsymbol{\theta})}$;
(ii) criticality preserving: for any data $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with mean-squared error (MSE)

empirical risk function $R_S(h) = \frac{1}{2}\sum_{i=1}^n (h(\boldsymbol{x}_i) - y_i)^2$, *if* $\nabla_{\boldsymbol{\theta}} R_S(f_{\boldsymbol{\theta}}) = \mathbf{0}$, *then* $\nabla_{\boldsymbol{\theta}} R_S(g_{\mathcal{P}(\boldsymbol{\theta})}) = \mathbf{0}$.

Lemma 3 (upper bound of optimistic sample size)**.** *Given two models* $f_{\boldsymbol{\theta}_A} = f(\cdot;\boldsymbol{\theta}_A)$ *with* $\boldsymbol{\theta}_A \in \mathbb{R}^{M_A}$ *and* $g_{\boldsymbol{\theta}_B} = g(\cdot;\boldsymbol{\theta}_B)$ *with* $\boldsymbol{\theta}_B \in \mathbb{R}^{M_B}$*, if there exists a critical mapping* $\mathcal{P}$ *from model A to B, then the optimistic sample size* $O_g(f^*) \leqslant O_f(f^*) \leqslant M_A$ *for any* $f^* \in \mathcal{F}_A$.

Remark 4. *If* $M_B \gg M_A$*, this upper bound estimate is highly informative, indicating target recovery capability at heavy overparameterization for model B. Importantly, this lemma establishes the relation between our rank stratification and previous studies about the critical embedding for the DNN loss landscape analysis. As a result, the critical embedding intrinsic to the DNN architecture not only benefits optimization as studied in previous works, but also profoundly benefits the recovery/generalization performance.*

Proof By Definition 6, for any $f^* \in \mathcal{F}_f$, there exists $\boldsymbol{\theta}^* \in \mathbb{R}^{M_A}$ such that $\mathrm{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*) = \mathrm{rank}_{f_{\boldsymbol{\theta}}}(f^*)$. Then, $g_{\mathcal{P}(\boldsymbol{\theta}^*)} = f_{\boldsymbol{\theta}^*} = f^*$. Because $R_S(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^n (f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - y_i)^2$, we have

$$\nabla_{\boldsymbol{\theta}} R_S(f_{\boldsymbol{\theta}^*}) = \sum_{i=1}^n (y_i - f^*(\boldsymbol{x}_i)) \nabla_{\boldsymbol{\theta}^*} f_{\boldsymbol{\theta}^*}(\boldsymbol{x}_i)$$

and $\nabla_{\mathcal{P}(\boldsymbol{\theta})} R_S(g_{\mathcal{P}(\boldsymbol{\theta}^*)}) = \sum_{i=1}^n (y_i - f^*(\boldsymbol{x}_i)) \nabla_{\mathcal{P}(\boldsymbol{\theta})} g_{\mathcal{P}(\boldsymbol{\theta}^*)}(\boldsymbol{x}_i)$. Because $\mathcal{P}$ is criticality preserving for arbitrary data S, we have $\ker(\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}^*}(\boldsymbol{X})) \subseteq \ker(\nabla_{\mathcal{P}(\boldsymbol{\theta})} g_{\mathcal{P}(\boldsymbol{\theta}^*)}(\boldsymbol{X}))$ for any $\boldsymbol{X} := [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n]$. Here, $\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}^*}(\boldsymbol{X}) = [\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}^*}(\boldsymbol{x}_1), \cdots, \nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}^*}(\boldsymbol{x}_n)]$. Because $\mathrm{rank}_S(\mathcal{P}(\boldsymbol{\theta}^*)) + \dim(\ker(\nabla_{\mathcal{P}(\boldsymbol{\theta})} g_{\mathcal{P}(\boldsymbol{\theta}^*)}(\boldsymbol{X}))) = \mathrm{rank}_S(\boldsymbol{\theta}^*) + \dim(\ker(\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}^*})) = n$, we have $\mathrm{rank}_S(\mathcal{P}(\boldsymbol{\theta}^*)) \leqslant \mathrm{rank}_S(\boldsymbol{\theta}^*)$ for any data S (See Definition 5 for $\mathrm{rank}_S(\cdot)$). Taking the infinite data limit, we obtain $\mathrm{rank}_g(\mathcal{P}(\boldsymbol{\theta}^*)) \leqslant \mathrm{rank}_f(\boldsymbol{\theta}^*) \leqslant M_A$. Therefore, $\mathrm{rank}_g(f^*) \leqslant \mathrm{rank}_f(f^*) \leqslant M_A$ for any $f^* \in \mathcal{F}_f$. By Theorem 1, $O_g(f^*) \leqslant O_f(f^*) \leqslant M_A$. ■

As a direct consequence of Lemma 3 and Theorem 2, we obtain the following theorem.

Theorem 3 (upper bound of optimistic sample size for DNNs)**.** *Given any NN with* M_{wide} *parameters, for any function in the function space of a narrower NN with* M_{narr} *parameters and for any* $f^* \in \mathcal{F}_{\mathrm{narr}}$*, we have* $O_{f_{\boldsymbol{\theta}_{\mathrm{wide}}}}(f^*) \leqslant O_{f_{\boldsymbol{\theta}_{\mathrm{narr}}}}(f^*) \leqslant M_{\mathrm{narr}}$.

Proof By Theorem 2, there exists a critical mapping $\mathcal{P} : \mathbb{R}^{M_{\mathrm{narr}}} \to \mathbb{R}^{M_{\mathrm{wide}}}$ from the narrower NN to the given NN. Then, by Lemma 3, $O_{f_{\boldsymbol{\theta}_{\mathrm{wide}}}}(f^*) \leqslant O_{f_{\boldsymbol{\theta}_{\mathrm{narr}}}}(f^*) \leqslant M_{\mathrm{narr}}$. ■

It should be noted that while Theorem 3 is proven here for fully-connected DNNs, the results are readily extendable to CNNs and ResNets by employing their respective critical embeddings, such as the splitting embedding. Theorem 3 provides a significantly more precise upper bound for optimistic sample sizes than the generic upper bound presented in Corollary 2, as demonstrated for an L-layer DNN in Table 1.

In Table 1, it's noteworthy that all function sets, except for $\mathcal{F}_{\{m_i\}_{i=1}^{L-1}}$ in the last row, where Theorem 3's bound significantly improves upon Corollary 2, have measure zero in $\mathcal{F}_{\{m_i\}_{i=1}^{L-1}}$. Nonetheless, estimating their optimistic sample sizes remains crucial as these values indicate the likelihood of a large DNN overfitting simple target functions. The generic

bound suggests that for a simple target function, the risk of overfitting increases with model size. However, this fails to account for the widely observed good generalization performance of DNNs under overparameterization (Zhang et al., 2017). In contrast, Theorem 3's bound indicates that one can employ very large DNNs to fit simple targets without significant concern for overfitting, as the bound on the optimistic sample size remains independent of the parameter size M. The qualitative difference between these suggestions demonstrates the significance of Theorem 3 for understanding DNN generalization.

model: $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{W}^{[L]}\sigma(\cdots\sigma(\boldsymbol{W}^{[1]}\boldsymbol{x})\cdots)$, $\boldsymbol{W}^{[l]} = \mathbb{R}^{m_l \times m_{l-1}}$,$m_L = 1$, $m_0 = d$		
f^*	generic bound (Cor. 2)	our bound (Thm. 3)
$\mathcal{F}_{\{1,1,\cdots,1\}}$	M	$d + L - 1$
$\vdots$	$\vdots$	$\vdots$
$\mathcal{F}_{\{m'_i\}_{i=1}^{L-1}}, 1 \leq m'_i \leq m_i$	M	$dm'_1 + m'_1m'_2 + \cdots + m'_{L-2}m'_{L-1} + m'_{L-1}$
$\vdots$	$\vdots$	$\vdots$
$\mathcal{F}_{\{m_i\}_{i=1}^{L-1}}$	M	M

Table 1: Upper bound of optimistic sample size for a general L-layer fully-connected DNN with width-$\{m_i\}_{i=1}^{L-1}$. For the simplicity of presentation, we consider the DNN without bias terms. Its total number of parameters $M = dm_1 + m_1m_2 + \cdots + m_{L-2}m_{L-1} + m_{L-1}$. $\mathcal{F}_{\{m_i\}_{i=1}^{L-1}}$ denotes the function space of the L-layer DNN with width-$\{m_i\}_{i=1}^{L-1}$ for hidden layers.

Corollary 5 (LLR-guarantee at overparameterization for DNNs)**.** *For a DNN model, all functions expressible by any narrower DNNs have LLR-guarantee at overparameterization.*

Proof We denote $f_{\boldsymbol{\theta}_{\text{wide}}}$ as the DNN model and M_{wide} as its total parameter size. For any narrower DNN $f_{\boldsymbol{\theta}_{\text{narr}}}$ (see Definition 7), its parameter size $M_{\text{narr}} < M_{\text{wide}}$. For any $f^* \in \mathcal{F}_{\text{narr}}$, by Theorem 3, the optimistic sample size $O_{f_{\boldsymbol{\theta}_{\text{wide}}}}(f^*) \leqslant O_{f_{\boldsymbol{\theta}_{\text{narr}}}}(f^*) \leqslant M_{\text{narr}} < M_{\text{wide}}$. By the Definition 3 of the optimistic sample size and Definition 2 of the LLR-guarantee, f^* has LLR-guarantee at overparameterization. ■

As far as we know, this is the first recovery guarantee at overparameterization for general DNNs. Though LLR-guarantee is a relatively weak form of recovery guarantee, it sets a solid ground for further improving the guarantee to stronger types of recovery, e.g., local or even global recovery. In particular, we reasonably conjecture that having LLR-guarantee at overparameterization is a necessary condition for having any stronger type of recovery guarantee.

Corollary 6 (free expressiveness in width)**.** *The optimistic sample size of a target function expressible by any DNN never increases as the DNN gets wider.*

Proof For any DNN $f_{\boldsymbol{\theta}_{\text{narr}}}$, any wider DNN $f_{\boldsymbol{\theta}_{\text{wide}}}$ and a target function $f^* \in \mathcal{F}_{\text{narr}}$, by Theorem 3, the optimistic sample size $O_{f_{\boldsymbol{\theta}_{\text{wide}}}}(f^*) \leqslant O_{f_{\boldsymbol{\theta}_{\text{narr}}}}(f^*)$. ■

Corollary 6 ensures that the potential for recovery in the sense of LLR is not compromised when employing wider neural networks. This provides theoretical justification for the application of large DNNs in modeling even simple target functions.

5. Optimistic Sample Sizes for Two-Layer Tanh NNs

Theorem 3 prompts the investigation of two pertinent questions: (i) the tightness of the established upper bound, and (ii) the methodology for determining the precise value of the optimistic sample size. In the subsequent analysis, we specifically address these issues for two-layer NNs with tanh activation functions. We provide exact estimates of the optimistic sample sizes that reach the upper limits as delineated by Theorem 3, with the results detailed in Table 2. Furthermore, we extend our estimation to the optimistic sample sizes for two-layer tanh-CNNs, both with and without the implementation of weight-sharing. A comparative analysis of the optimistic sample sizes for fully-connected NNs versus CNNs reveals that superfluous connections among neurons lead to an increase in the optimistic sample size, which in turn adversely affects the fitting performance in terms of LLR.

model: $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{i=1}^{m} a_i \tanh(\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^d, \boldsymbol{\theta} = (a_i, \boldsymbol{w}_i)_{i=1}^{m}$			
f^*	generic bound (Cor. 2)	upper bound (Thm. 3)	$O_{f_{\boldsymbol{\theta}}}(f^*)$ (Thm. 4)
$\{0(\cdot)\}$	0	0	0
$\mathcal{F}_1^{\text{NN}} \backslash \{0(\cdot)\}$	$m(d+1)$	$d+1$	$d+1$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{F}_k^{\text{NN}} \backslash \mathcal{F}_{k-1}^{\text{NN}}$	$m(d+1)$	$k(d+1)$	$k(d+1)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{F}_m^{\text{NN}} \backslash \mathcal{F}_{m-1}^{\text{NN}}$	$m(d+1)$	$m(d+1)$	$m(d+1)$

Table 2: Results of the optimistic sample sizes for two-layer width-m tanh-NN. Here $\mathcal{F}_k^{\text{NN}} := \{\sum_{i=1}^{k} a_i^* \sigma(\boldsymbol{w}_i^{*\mathrm{T}} \boldsymbol{x}) | a_i \in \mathbb{R}, \boldsymbol{w}_i \in \mathbb{R}^d\}$ denotes the function space of the width-k tanh-NN.

5.1 Theoretical Preparation

The exact estimation of model rank hinges on unraveling the linear dependencies among tangent functions. The subsequent results concerning linear independence form the bedrock for estimating the model rank in two-layer neural networks.

Proposition 3 (linear independence of neurons)**.** *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be any analytic function such that $\sigma^{(n_j)}(0) \neq 0$ for an infinite sequence of distinct indices $\{n_j\}_{j=1}^{\infty}$. Given $d \in \mathbb{N}$ and m distinct weights $\boldsymbol{w}_1, ..., \boldsymbol{w}_m \in \mathbb{R}^d \backslash \{\mathbf{0}\}$, such that $\boldsymbol{w}_k \neq \pm \boldsymbol{w}_j$ for all $1 \leqslant k < j \leqslant m$. Then $\{\sigma(\boldsymbol{w}_i^T \boldsymbol{x}), \sigma'(\boldsymbol{w}_i^T \boldsymbol{x})x_1, ..., \sigma'(\boldsymbol{w}_i^T \boldsymbol{x})x_d\}_{i=1}^m$ is a linearly independent function set.*

Proof For x sufficiently close to $0 \in \mathbb{R}$, we can write $\sigma(x) = \sum_{j=0}^{\infty} c_j x^j$, where $c_j = \sigma^{(j)}(0)/(j!)$. Then, $\sigma'(x) = \sum_{j=1}^{\infty} j c_j x^{j-1}$. Suppose that the set is not linearly independent. Choose not-all-zero constants $\{\alpha_i\}_{i=1}^m$ and $\{\beta_{i1}, ..., \beta_{id}\}_{i=1}^m$ such that

$$\boldsymbol{x} \mapsto \sum_{i=1}^{m} \left(\alpha_i \sigma(\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{x}) + \sum_{t=1}^{d} \beta_{it} \sigma'(\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{x}) x_t \right)$$

is a zero map on $\mathbb{R}^d$, where x_t denotes the t-th component of input. For $k, j, i \in [d]$, define the sets

$$\begin{aligned} A_{k,j} &:= \{\boldsymbol{x} \in \mathbb{R}^d | \, \langle \boldsymbol{x}, \boldsymbol{w}_k \pm \boldsymbol{w}_j \rangle = 0\} \\ B_i &:= \{\boldsymbol{x} \in \mathbb{R}^d | \, \langle \boldsymbol{x}, \boldsymbol{w}_i \rangle = 0\}. \end{aligned}$$

Clearly, each $A_{k,j}$ is the union of two linear subspaces of dimension $(d-1)$, while each B_i is a possibly empty affine subspace of dimension $(d-1)$. Thus,

$$E := (\cup_{1 \leqslant k,j \leqslant m} A_{k,j}) \cup \left(\cup_{i=1}^{d} B_i \right)$$

has $\mathcal{L}^d$ Lebesgue measure zero. Let $\boldsymbol{e} \in \mathbb{R}^d \backslash E$. Denote $p_i := \langle \boldsymbol{w}_i, \boldsymbol{e} \rangle$ for each $i \in [m]$. Since $p_i \neq p_j$ and $p_i + p_j \neq 0$ whenever $i \neq j$, we can, without loss of generality, assume that $|p_1| > |p_2| > ... > |p_m| > 0$. For any sufficiently small ε and any i, t we have

$$\begin{aligned} \sigma(\boldsymbol{w}_i^{\mathrm{T}}(\varepsilon \boldsymbol{e})) &= \sum_{j=0}^{\infty} (c_j p_i^j) \varepsilon^j, \\ \sigma'(\boldsymbol{w}_i^{\mathrm{T}}(\varepsilon \boldsymbol{e}))(\varepsilon \boldsymbol{e})_t &= e_t \sum_{j=1}^{\infty} (j c_j p_i^{j-1}) \varepsilon^j. \end{aligned}$$

Thus, for sufficiently small ε,

$$\begin{aligned} \sum_{i=1}^{m} (\alpha_i \sigma(\boldsymbol{w}_i^{\mathrm{T}}(\varepsilon \boldsymbol{e})) + \sum_{t=1}^{d} \beta_{it} \sigma'(\boldsymbol{w}_i^{\mathrm{T}}(\varepsilon \boldsymbol{e}))(\varepsilon \boldsymbol{e})_t) &= \sum_{i=1}^{m} \alpha_i c_0 + \sum_{j=1}^{\infty} c_j \sum_{i=1}^{m} (\alpha_i + \frac{1}{p_i} \sum_{t=1}^{d} j \beta_{it} e_t) p_i^j \varepsilon^j \\ &= 0. \end{aligned} \tag{7}$$

We have $c_j \sum_{i=1}^m \left(\alpha_i + \frac{1}{p_i}\sum_{t=1}^d j\beta_{it}e_t\right) p_i^j = 0$ for all $j \in \mathbb{N}$. In particular, for any $j \geq 2$, since $n_j \geq 1$ and $c_{n_j} \neq 0$, we have $\sum_{i=1}^m \left(\alpha_i + \frac{1}{p_i}\sum_{t=1}^d n_j\beta_{it}e_t\right) p_i^{n_j} = 0$, which yields

$$\alpha_1 + \frac{1}{p_1}\sum_{t=1}^d n_j\beta_{1t}e_t = -\sum_{i=2}^m \left(\alpha_i + \frac{1}{p_i}\sum_{t=1}^d n_j\beta_{it}e_t\right)\frac{p_i^{n_j}}{p_1^{n_j}}.$$

If $m = 1$, by taking limits $j \to \infty$, we have $\alpha_1 = \sum_{t=1}^d \beta_{1t}e_t = 0$.

Otherwise, since $|p_1| > |p_i|$ for any $2 \leqslant i \leqslant m$, it follows that, by taking limits $j \to \infty$,

$$\lim_{j\to\infty}\left(\alpha_1 + \frac{1}{p_1}\sum_{t=1}^d n_j\beta_{1t}e_t\right) = \lim_{j\to\infty} -\sum_{i=2}^m \left(\alpha_i + \frac{1}{p_i}\sum_{t=1}^d n_j\beta_{it}e_t\right)\frac{p_i^{n_j}}{p_1^{n_j}} = 0.$$

Thus, we also have $\alpha_1 = \sum_{t=1}^d \beta_{1t}e_t = 0$. For $m > 2$, we may rewrite Equation (7) as

$$\alpha_2 + \frac{1}{p_2}\sum_{t=1}^d n_j\beta_{2t}e_t = -\sum_{i=3}^m \left(\alpha_i + \frac{1}{p_i}\sum_{t=1}^d n_j\beta_{it}e_t\right)\frac{p_i^{n_j}}{p_2^{n_j}}$$

for each $j \geq 2$, and take limits as we do above to deduce that $\alpha_2 + \frac{1}{p_2}\sum_{t=1}^d n_j\beta_{2t}e_t = 0$. By repeating this procedure for at most m times, we conclude that $\alpha_i + \frac{1}{p_i}\sum_{t=1}^d n_j\beta_{it}e_t = 0$ for all $i \in [m]$. Then, $\alpha_i = \sum_{t=1}^d \beta_{it}e_t = 0$ for any $i \in [m]$. For each i, $\sum_{t=1}^d \beta_{it}e_t'$ is a linear function of $\boldsymbol{e}'$ on the open set $\mathbb{R}^d \backslash E$ which vanishes on a neighborhood of $\boldsymbol{e}$, we must have $\alpha_i = \beta_{it} = 0$ for any $i \in [m], t \in [d]$. Therefore, $\{\sigma(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x}), \sigma'(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x})x_1, ..., \sigma'(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x})x_d\}_{i=1}^m$ must be a linearly independent set. ■

Corollary 7 (model rank estimate for two-layer tanh-NNs)**.** *Let $\sigma = \tanh$. Given $d \in \mathbb{N}$, weights $\boldsymbol{w}_1, ..., \boldsymbol{w}_m \in \mathbb{R}^d$, $a_1, ..., a_m \in \mathbb{R}$, we have*

$$\dim(\mathrm{span}\{\sigma(\boldsymbol{w}_i^T\boldsymbol{x}), a_i\sigma'(\boldsymbol{w}_i^T\boldsymbol{x})x_1, ..., a_i\sigma'(\boldsymbol{w}_i^T\boldsymbol{x})x_d\}_{i=1}^m) = m_{\boldsymbol{w}} + m_a d,$$

where $m_{\boldsymbol{w}} = \frac{1}{2}|\{\boldsymbol{w}_i, -\boldsymbol{w}_i | \boldsymbol{w}_i \neq \boldsymbol{0}, i \in [m]\}|$ indicating the number of independent neurons, $m_a = \frac{1}{2}|\{\boldsymbol{w}_i, -\boldsymbol{w}_i | \boldsymbol{w}_i \neq \boldsymbol{0}, a_i \neq 0, i \in [m]\}| + |\{\boldsymbol{w}_i | \boldsymbol{w}_i = \boldsymbol{0}, a_i \neq 0, i \in [m]\}|$ indicating the number of independent effective neurons. Here, $|\cdot|$ is the cardinality of a set, i.e., number of different elements in a set.

Proof Note that $\sigma = \tanh$ is analytic and $\sigma^{(2n+1)}(0) \neq 0$ for all n. Because tanh is an odd function, we have $\tanh(x) = -\tanh(-x)$ and $\tanh(0) = 0$. Therefore, given $\boldsymbol{w}_i, \boldsymbol{w}_j \neq \boldsymbol{0}$ with $\boldsymbol{w}_i = \pm\boldsymbol{w}_j$, $\mathrm{span}\{\sigma(\boldsymbol{w}_i^{\mathrm{T}}x)\} = \mathrm{span}\{\sigma(\boldsymbol{w}_j^{\mathrm{T}}x)\}$ and $\mathrm{span}\{\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1, ..., \sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d\} = \mathrm{span}\{\sigma'(\boldsymbol{w}_j^{\mathrm{T}}x)x_1, ..., \sigma'(\boldsymbol{w}_j^{\mathrm{T}}x)x_d\}$. Since there are $m_{\boldsymbol{w}}$ different non-zero weights, by Proposition 3 we have

$$\dim\left(\mathrm{span}\{\sigma(\boldsymbol{w}_i^{\mathrm{T}}x)\}_{i=1}^m\right) = m_{\boldsymbol{w}}.$$

Furthermore, note that

$$\mathrm{span}\{a_i\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1, ..., a_i\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d\}_{i=1}^m = \mathrm{span}\{\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1, ..., \sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d : a_i \neq 0, i \in [m]\}.$$

Thus, by Proposition 3,

$$\begin{aligned}&\dim\left(\mathrm{span}\{\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1,...,\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d : \boldsymbol{w}_i\neq\boldsymbol{0}, a_i\neq 0, i\in[m]\}\right)\\ =&\frac{1}{2}|\{\boldsymbol{w}_i,-\boldsymbol{w}_i|\boldsymbol{w}_i\neq\boldsymbol{0},a_i\neq 0,i\in[m]\}|\cdot d.\end{aligned}$$

Now suppose that $\boldsymbol{w}_j=\boldsymbol{0}\in\mathbb{R}^d$ for some $j\in[m]$. Since $\sigma'(\boldsymbol{w}^{\mathrm{T}}x)=\sigma'(0)\neq 0$ for all $x\in\mathbb{R}^d$,

$$\mathrm{span}\{\sigma'(\boldsymbol{w}_j^{\mathrm{T}}x)x_1,...,\sigma'(\boldsymbol{w}_j^{\mathrm{T}}x)x_d\}=\mathrm{span}\{x_1,...,x_d\}$$

which consists only of linear functions. By the nonlinearity of tanh, we conclude that

$$\dim\left(\mathrm{span}\{\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1,...,\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d\}\right)=m_a d$$

and thus $\dim\left(\mathrm{span}\{\sigma(\boldsymbol{w}_i^{\mathrm{T}}x),\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_1,...,\sigma'(\boldsymbol{w}_i^{\mathrm{T}}x)x_d\}\right)=m_{\boldsymbol{w}}+m_a d$ as desired. ■

Next we consider the estimate of the model rank for CNNs which are widely used in practice. Here we consider the case where the input has two-dimensional indices, which is the most general case for the image input. The following two propositions can be directly generalized to the model rank estimate of CNNs with an input of one index dimension.

Definition 9 (ineffective neurons/kernels)**.** *For two-layer tanh NNs, we say a neuron/kernel is output-ineffective if its output weight array is zero but input weight array is nonzero. We say a neuron/kernel is input-ineffective if its input weight array is zero but output weight array is nonzero. We say a neuron/kernel is null if both input and out weight arrays are zero. All these neurons/kernels are ineffective. Neurons/kernels that are not ineffective are effective.*

Remark 5. *Estimating the model rank at parameter points with input-ineffective neurons is complicated for CNNs. Luckily, we notice that model rank of a target function $f^*\in\mathcal{F}$ can be obtained by considering only $\boldsymbol{\theta}'\in\mathcal{M}_{f^*}$ with no input-ineffective neurons. This is because any $\boldsymbol{\theta}'\in\mathcal{M}_{f^*}$ with input-ineffective neurons associates to a $\boldsymbol{\theta}''\in\mathcal{M}_{f^*}$ such that (i) input-ineffective neurons at $\boldsymbol{\theta}'$ are replaced by null neurons (ii) $\mathrm{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}'')\leqslant\mathrm{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$. Therefore, we only estimate the model rank for CNNs at the parameter points with no input-ineffective neurons in the following propositions.*

Proposition 4 (model rank estimate for CNNs (with weight sharing))**.** *Given $m\in\mathbb{N}$, $d\in\mathbb{N}$ and $s\in[d]$. For any $l\in[m]$, let $\boldsymbol{K}_l$ be a $(s\times s)$ matrix. Consider CNNs with stride = 1. For a tanh-CNN $f_{\boldsymbol{\theta}}$ with weight sharing,*

$$f_{\boldsymbol{\theta}}(\boldsymbol{I})=\sum_{l=1}^{m}\sum_{i,j=1}^{d+1-s}a_{lij}\tanh\left(\sum_{\alpha,\beta}I_{i+s-\alpha,j+s-\beta}K_{l;\alpha,\beta}\right),\boldsymbol{I}\in\mathbb{R}^{d\times d},$$

its model rank at $\boldsymbol{\theta}=(a_{lij},\boldsymbol{K}_l)_{l,i,j}$ with no input-ineffective neurons is given by

$$\mathrm{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta})=m_a s^2+m_K(d+1-s)^2,$$

where $m_K = \frac{1}{2}|\{\boldsymbol{K}_l, -\boldsymbol{K}_l | l \in [m], \boldsymbol{K}_l \neq \boldsymbol{0}\}|$ *indicating the number of independent kernels,* $m_a = \frac{1}{2}\sum_{\boldsymbol{K}\in\mathcal{K}} \dim(\text{span}\{a_{l,:,:}\}_{l\in h(\boldsymbol{K})})$ *indicating the number of independent effective neurons. Here* $\mathcal{K} = \{\boldsymbol{K}_l, -\boldsymbol{K}_l | l \in [m], \boldsymbol{K}_l \neq \boldsymbol{0}\}$*,* h *is a function over* $\mathcal{K}$ *s.t. for each* $\boldsymbol{K} \in \mathcal{K}, h(\boldsymbol{K}) = \{l | l \in [m], \boldsymbol{K}_l = \pm\boldsymbol{K}\}$*.* $|\cdot|$ *is the cardinality of a set, i.e., number of different elements in a set, and* $a_{l,:,:}$ *denotes the* $(d+1-s)\times(d+1-s)$ *matrix whose entries are* a_{lij}*'s.*

Proof Let $\sigma = \tanh$. We first consider the case in which there is no ineffective neuron (i.e., $a_{lij} \neq 0$ for all l, i, j) and $\boldsymbol{K}_l \pm \boldsymbol{K}_{l'} \neq \boldsymbol{0}$ for any distinct $l, l' \in [m]$. In this case the model rank is the dimension of the following function space (with respect to variable $\boldsymbol{I} \in \mathbb{R}^{d\times d}$)

$$\begin{aligned}
&\text{span}\left\{\frac{\partial f_{\boldsymbol{\theta}}}{\partial a_{lij}}, \frac{\partial f_{\boldsymbol{\theta}}}{\partial K_{l;\alpha,\beta}}\right\} \\
&= \text{span}\left\{\sigma\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right),\right. \\
&\left.\sum_{i',j'=1}^{d+1-s} a_{li'j'}\sigma'\left(\sum_{\alpha',\beta'} I_{i'+s-\alpha',j'+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i'+s-\alpha,j'+s-\beta}\right\}_{l,i,j,\alpha,\beta},
\end{aligned}$$

where $l \in [m]$ and $\alpha, \beta \in [s]$. Next, we prove by contradiction that the set of functions

$$\left\{\sum_{i,j=1}^{d+1-s} a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta}\right\}_{l\in[m],\alpha,\beta\in[s]}$$

are linearly independent. If they are not linearly independent, there exist not all zero constants $\zeta_{l11}, ..., \zeta_{lss}$ for $l \in [m]$, such that

$$\sum_{l=1}^{m}\sum_{\alpha,\beta=1}^{s} \zeta_{l\alpha\beta}\sum_{i,j=1}^{d+1-s} a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0,$$

which implies that the set of functions

$$\left\{a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta}\right\}_{l,i,j,\alpha,\beta}$$

are linearly dependent, contradicting Proposition 3.

In presence of null neurons and output-ineffective neurons, if $a_{lij} = 0$ for certain $l \in [m]$ and all $i, j \in \{1, ..., d+1-s\}$,

$$\sum_{i,j=1}^{d+1-s} a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0$$

for all $\alpha, \beta \in [s]$. If $\boldsymbol{K}_l = \boldsymbol{0}$ for certain $l \in [m]$, then by Definition 9 $a_{lij} = 0$ must hold for all i, j. Thus, we have

$$\sigma\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0,$$

$$\sum_{i,j=1}^{d+1-s} a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{l;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0.$$

Moreover, notice that two kernels with $\boldsymbol{K}_l = \pm\boldsymbol{K}_{l'}$ can be reduced to one while maintaining model rank if and only if the corresponding output weights $a_{l,:,:}$ and $a_{l',:,:}$ are linearly dependent. Then, similar to Proposition 3, we conclude that the model rank is $m_a s^2 + m_K(d+1-s)^2$. ■

Proposition 5 (model rank estimate for CNN-NS)**.** *Given $m \in \mathbb{N}$, $d \in \mathbb{N}$ and $s \in [d]$. For any $l \in [m]$ and $i, j \in [d+1-s]$, let $\boldsymbol{K}_{lij}$ be a $(s \times s)$ matrices. Consider CNNs with stride = 1. For a* tanh *CNN-NS $f_{\boldsymbol{\theta}}$,*

$$f_{\boldsymbol{\theta}}(\boldsymbol{I}) = \sum_{l=1}^{m}\sum_{i,j=1}^{d+1-s} a_{lij}\tanh\left(\sum_{\alpha,\beta} I_{i+s-\alpha,j+s-\beta} K_{lij;\alpha,\beta}\right), \boldsymbol{I} \in \mathbb{R}^{d\times d},$$

its model rank at $\boldsymbol{\theta} = (a_{lij}, \boldsymbol{K}_{lij})_{l,i,j}$ with no input-ineffective neuron is given by

$$\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}) = m_a s^2 + m_K,$$

where $m_K = \frac{1}{2}|\{p(\boldsymbol{K}_{lij}), -p(\boldsymbol{K}_{lij}) | l \in [m], i, j \in [d+1-s], \boldsymbol{K}_{lij} \neq \boldsymbol{0}\}|$ indicating the number of independent kernels, $m_a = \frac{1}{2}|\{p(\boldsymbol{K}_{lij}), -p(\boldsymbol{K}_{lij}) | l \in [m], i, j \in [d+1-s], a_{lij} \neq 0\}|$ indicating the number of independent effective neurons. Here p is the padding function over kernels, i.e., for each $(s \times s)$ kernel $\boldsymbol{K}_{lij}$, $p(\boldsymbol{K}_{lij}) \in \mathbb{R}^{d\times d}$ s.t. $p(\boldsymbol{K}_{lij})[i : i+s-1, j : j+s-1] = \boldsymbol{K}_{lij}$ and the other elements of $p(\boldsymbol{K}_{lij})$ are zero. $|\cdot|$ is the cardinality of a set, i.e., number of different elements in a set.

Proof Let $\sigma = \tanh$. The model rank is the dimension of the following function space

$$\begin{aligned}
&\operatorname{span}\left\{\frac{\partial f_{\boldsymbol{\theta}}}{\partial a_{lij}}, \frac{\partial f_{\boldsymbol{\theta}}}{\partial K_{lij;\alpha,\beta}}\right\}_{l,i,j,\alpha,\beta} \\
&= \operatorname{span}\left\{\sigma\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{lij;\alpha',\beta'}\right),\right. \\
&\qquad \left. a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{lij;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta}\right\}_{l,i,j,\alpha,\beta},
\end{aligned}$$

where $l \in [m]$, $1 \leqslant i, j \leqslant d+1-s$, and $\alpha, \beta \in [s]$. Also note that if $a_{lij} = 0$ for some $l \in [m]$ and $i, j \in \{1, ..., d+1-s\}$, then

$$a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{lij;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0$$

for all $\alpha, \beta \in [s]$. If $\boldsymbol{K}_{lij} = \mathbf{0}$ for some $l \in [m]$ and $i, j \in \{1, ..., d+1-s\}$, because there is no input-ineffective neurons, we must have $a_{lij} = 0$. Then

$$\sigma\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{lij;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0,$$

$$a_{lij}\sigma'\left(\sum_{\alpha',\beta'} I_{i+s-\alpha',j+s-\beta'} K_{lij;\alpha',\beta'}\right) I_{i+s-\alpha,j+s-\beta} = 0.$$

It follows from Proposition 3 that this space has dimension $m_a s^2 + m_K$. ■

5.2 Optimistic Sample Size Estimates

Theorem 4 (optimistic sample sizes for two-layer tanh-NN)**.** *Given a two-layer NN* $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sum_{i=1}^m a_i \tanh(\boldsymbol{w}_i^T \boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^d, \boldsymbol{\theta} = (a_i, \boldsymbol{w}_i)_{i=1}^m$*, for any target function* $f^* \in \mathcal{F}_k^{\mathrm{NN}} \backslash \mathcal{F}_{k-1}^{\mathrm{NN}}$ *with* $0 \leqslant k \leqslant m$*, the optimistic sample size*

$$O_{f_{\boldsymbol{\theta}}}(f^*) = k(d+1).$$

Here $\mathcal{F}_k^{\mathrm{NN}} := \{\sum_{i=1}^k a_i^* \sigma(\boldsymbol{w}_i^{*T} \boldsymbol{x}) | a_i \in \mathbb{R}, \boldsymbol{w}_i \in \mathbb{R}^d\}$ *for* $k \in \mathbb{N}^+$, $\mathcal{F}_0^{\mathrm{NN}} := \{0(\cdot)\}$ *and* $\mathcal{F}_{-1}^{\mathrm{NN}} := \emptyset$.

Proof Given any target function $f^* \in \mathcal{F}_k^{\mathrm{NN}} \backslash \mathcal{F}_{k-1}^{\mathrm{NN}}$, for $k = 0$, $f^* = 0(\cdot)$ and $O_{f_{\boldsymbol{\theta}}}(f^*) = \mathrm{rank}_{f_{\boldsymbol{\theta}}}(f^*) = 0 = k(d+1)$. For $0 < k \leqslant m$, we have

$$\mathcal{F}_k^{\mathrm{NN}} \backslash \mathcal{F}_{k-1}^{\mathrm{NN}} = \{\sum_{i=1}^k a_i \tanh(\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{x}), a_i \neq 0, \boldsymbol{w}_i \neq \mathbf{0}, \boldsymbol{w}_i \neq \pm \boldsymbol{w}_j\}.$$

Therefore, there exists $\boldsymbol{\theta}^* = (a_i^*, \boldsymbol{w}_i^*)_{i=1}^k$ with $a_i^* \neq 0$, $\boldsymbol{w}_i^* \neq \mathbf{0}$, and $\boldsymbol{w}_i^* \neq \pm \boldsymbol{w}_j^*$ for $i \neq j$, such that $f^* = f_{\boldsymbol{\theta}^*} := \sum_{i=1}^k a_i^* \tanh(\boldsymbol{w}_i^{*\mathrm{T}} \boldsymbol{x})$. By the upper bound estimate Theorem 3, $O_{f_{\boldsymbol{\theta}}}(f^*) \leqslant k(d+1)$.

By definition, the model rank of f^* is the minimal model rank among all parameters recovering f^* in the target set $\mathcal{M}_{f^*}$. For any $\boldsymbol{\theta}' = (a_i', \boldsymbol{w}_i')_{i=1}^m \in \mathcal{M}_{f^*}$, by Corollary 7, $\mathrm{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') = m'_{\boldsymbol{w}} + m'_a d$, where $m'_{\boldsymbol{w}} = \frac{1}{2}|\{\boldsymbol{w}_i', -\boldsymbol{w}_i' | \boldsymbol{w}_i' \neq \mathbf{0}, i \in [m]\}|$, $m_a = \frac{1}{2}|\{\boldsymbol{w}_i', -\boldsymbol{w}_i' | \boldsymbol{w}_i' \neq \mathbf{0}, a_i' \neq 0, i \in [m]\}| + |\{\boldsymbol{w}_i' | \boldsymbol{w}_i' = \mathbf{0}, a_i' \neq 0, i \in [m]\}|$. Because

$$\sum_{i=1}^m a_i' \tanh(\boldsymbol{w}_i'^{\mathrm{T}} \boldsymbol{x}) = \sum_{i=1}^k a_i^* \tanh(\boldsymbol{w}_i^{*\mathrm{T}} \boldsymbol{x}) = f^*(\boldsymbol{x}),$$

by the linear independence of neurons Proposition 3, $m'_{\boldsymbol{w}} \geqslant k$ and $m'_a \geqslant k$. Then $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') \geqslant k(d+1)$, which yields $O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) \geqslant k(d+1)$. Therefore $O_{f_{\boldsymbol{\theta}}}(f^*) = k(d+1)$.

■

Theorem 5 (optimistic sample sizes for two-layer tanh-CNN)**.** *Given a m-kernel two-layer CNN with weight sharing with 2-d input $I \in \mathbb{R}^{d\times d}$, $s \times s$ kernel and stride 1*

$$f_{\boldsymbol{\theta}}(\boldsymbol{I}) = \sum_{l=1}^{m}\sum_{i,j=1}^{d+1-s} a_{lij} \tanh\left(\sum_{\alpha,\beta} I_{i+s-\alpha,j+s-\beta} K_{l;\alpha,\beta}\right), \quad \boldsymbol{I} \in \mathbb{R}^{d\times d},$$

for any target function $f^* \in \mathcal{F}_k^{\mathrm{CNN}} \backslash \mathcal{F}_{k-1}^{\mathrm{CNN}}$ with $0 \leqslant k \leqslant m$, the optimistic sample size

$$O_{f_{\boldsymbol{\theta}}}(f^*) = k(s^2 + (d+1-s)^2).$$

Here $\mathcal{F}_k^{\mathrm{CNN}}$ indicates the function space of k-kernel CNN for $k \in \mathbb{N}^+$, $\mathcal{F}_0^{\mathrm{CNN}} := \{0(\cdot)\}$ and $\mathcal{F}_{-1}^{\mathrm{CNN}} := \emptyset$.

Proof Let $\sigma = \tanh$. The above theorem obviously holds for $k = 0$. For any target function $f^* \in \mathcal{F}_k^{\mathrm{CNN}} \backslash \mathcal{F}_{k-1}^{\mathrm{CNN}}$ with $0 < k \leqslant m$, there exists $\boldsymbol{\theta}^* = (a^*_{lij}, \boldsymbol{K}^*_l)_{l\in[k],i,j\in[d+1-s]}$ satisfying (i) $\boldsymbol{K}^*_l \neq \pm\boldsymbol{K}^*_{l'}$ for any $l \neq l'$ and (ii) $\forall l \in [k], \exists a^*_{lij} \neq 0$, such that

$$f^*(\boldsymbol{I}) = f_{\boldsymbol{\theta}^*}(\boldsymbol{I}) = \sum_{l=1}^{k}\sum_{i,j=1}^{d+1-s} a^*_{lij}\sigma\left(\sum_{\alpha,\beta} I_{i+s-\alpha,j+s-\beta} K^*_{l;\alpha,\beta}\right).$$

By the upper bound estimate Theorem 3, $O_{f_{\boldsymbol{\theta}}}(f^*) \leqslant k(s^2 + (d+1-s)^2)$ *.

Next we prove that $k(s^2 + (d+1-s)^2)$ is also a lower bound of $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}')$ for $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$. Note that, we only need to consider the parameter points with no input-ineffective neuron. For any $\boldsymbol{\theta}' = (a'_{lij}, \boldsymbol{K}'_l)_{l\in[m],i,j\in[d+1-s]} \in \mathcal{M}_{f^*}$ with no input-ineffective neuron, by Proposition 5,

$$\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') = m'_a s^2 + m'_K (d+1-s)^2,$$

where $m'_K = \frac{1}{2}|\mathcal{K}|$, $m'_a = \frac{1}{2}\sum_{\boldsymbol{K}\in\mathcal{K}} \dim(\operatorname{span}\{a_{l,:,:}\}_{l\in h(\boldsymbol{K})})$ with $\mathcal{K} = \{\boldsymbol{K}'_l, -\boldsymbol{K}'_l | l \in [m], \boldsymbol{K}'_l \neq \boldsymbol{0}\}$. Here h is a function over $\mathcal{K}$ such that for each $\boldsymbol{K} \in \mathcal{K}, h(\boldsymbol{K}) = \{l | l \in [m], \boldsymbol{K}'_l = \pm\boldsymbol{K}\}$. Because

$$\sum_{l=1}^{k}\sum_{i,j=1}^{d+1-s} a^*_{lij}\sigma\left(\sum_{\alpha,\beta} I_{i+s-\alpha,j+s-\beta} K^*_{l;\alpha,\beta}\right) = \sum_{l=1}^{m}\sum_{i,j=1}^{d+1-s} a'_{lij}\sigma\left(\sum_{\alpha,\beta} I_{i+s-\alpha,j+s-\beta} K'_{l;\alpha,\beta}\right),$$

by the linear independence of neurons Proposition 3, $m_K \geqslant k$ and $m_a \geqslant k$. Therefore $\operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') \geqslant k(s^2 + (d+1-s)^2)$ for $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$, which yields $O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) \geqslant$

∗. Although Theorem 3 surely holds for CNNs, we do not prove it in our work because it requires proving the Embedding Principle for CNNs out of the focus of the current work. For the rigor of our proof, Theorem 3 can be walk around as follows. Considering $\boldsymbol{\theta}' = (a'_{lij}, \boldsymbol{K}'_l)_{l\in[m],i,j\in[d+1-s]}$ with $a'_{lij} = a^*_{lij}, \boldsymbol{K}'_l = \boldsymbol{K}^*_l$ for $l \in [k]$ and $a'_{lij} = 0, \boldsymbol{K}'_l = \boldsymbol{0}$ for $l > k$, then $O_{f_{\boldsymbol{\theta}}}(f^*) = \operatorname{rank}_{f_{\boldsymbol{\theta}}}(f^*) \leqslant \operatorname{rank}_{f_{\boldsymbol{\theta}}}(\boldsymbol{\theta}') = k(s^2 + (d+1-s)^2)$.

$k(s^2+(d+1-s)^2)$. Then we obtain $O_{f_{\boldsymbol{\theta}}}(f^*)=k(s^2+(d+1-s)^2)$. ■

To compare CNNs with and without weight sharing, we can estimate the optimistic sample sizes of functions in $\mathcal{F}_k^{\mathrm{CNN}}\backslash\mathcal{F}_{k-1}^{\mathrm{CNN}}$ for both models using $m\geqslant k$ kernels. However, we observe that the optimistic sample size varies within this function set for CNN-NS, complicating our comparison. To address this issue, we consider a smaller function set $\mathcal{G}_k^*\subset\mathcal{F}_k^{\mathrm{CNN}}\backslash\mathcal{F}_{k-1}^{\mathrm{CNN}}$ defined as follows:

Definition 10 (all-effective k-kernel CNN functions)**.** *Let $\mathcal{F}_0^{\mathrm{CNN}}=\{0\}$. For $k\in\mathbb{Z}^+$, the all-effective k-kernel CNN function set $\mathcal{G}_k^*$ is defined as the set of all functions in $\mathcal{F}_k^{\mathrm{CNN}}\backslash\mathcal{F}_{k-1}^{\mathrm{CNN}}$ that can be represented by a k-kernel CNN where each output weight value is nonzero (i.e., $a_{lij}\neq 0$ for all l,i,j).*

Theorem 6 (optimistic sample sizes of CNN functions in two-layer CNN-NS (no-sharing CNN, CNN-NS))**.** *We consider a m-kernel two-layer no-sharing CNN (CNN-NS) with 2-d input $I\in\mathbb{R}^{d\times d}$, $s\times s$ kernel and stride 1*

$$f_{\boldsymbol{\theta}}(\boldsymbol{I})=\sum_{l=1}^{m}\sum_{i,j=1}^{d+1-s}a_{lij}\tanh\left(\sum_{\alpha,\beta}I_{i+s-\alpha,j+s-\beta}K_{lij;\alpha,\beta}\right),\quad \boldsymbol{I}\in\mathbb{R}^{d\times d}.$$

For any target function $f^*\in\mathcal{G}_k^*$ with $0\leqslant k\leqslant m$, then the optimistic sample size

$$O_{f_{\boldsymbol{\theta}}}(f^*)=k(s^2+1)(d+1-s)^2.$$

Proof Let $\sigma=\tanh$. The above theorem obviously holds for $k=0$. For any target function $f^*\in\mathcal{G}_k^*$ with $0<k\leqslant m$, there is a point $\boldsymbol{\theta}^*$ of CNN (with sharing) such that

$$f^*(\boldsymbol{I})=f_{\boldsymbol{\theta}^*}(\boldsymbol{I})=\sum_{l=1}^{k}\sum_{i,j=1}^{d+1-s}a_{lij}^*\sigma\left(\sum_{\alpha,\beta}I_{i+s-\alpha,j+s-\beta}H_{l;\alpha,\beta}^*\right),$$

where $\boldsymbol{H}_l^*\neq\pm\boldsymbol{H}_{l'}^*$ for any $l\neq l'$ and $a_{lij}^*\neq 0$ for any l,i,j. Then the target can also be represented by the following CNN-NS

$$f^*(\boldsymbol{I})=f_{\boldsymbol{\theta}_{\mathrm{NS}}^*}(\boldsymbol{I})=\sum_{l=1}^{k}\sum_{i,j=1}^{d+1-s}a_{lij}^*\sigma\left(\sum_{\alpha,\beta}I_{i+s-\alpha,j+s-\beta}K_{lij;\alpha,\beta}^*\right),$$

where $\boldsymbol{K}_{lij}^*=\boldsymbol{H}_l^*$ for all l,i,j, $\boldsymbol{\theta}_{\mathrm{NS}}^*=(a_{lij}^*,\boldsymbol{K}_{lij}^*)_{l\in[k],i,j\in[d+1-s]}$. By Proposition 4, the model rank at $\boldsymbol{\theta}_{\mathrm{NS}}^*$ for the k-kernel CNN-NS becomes

$$\mathrm{rank}_{\mathrm{CNN}_{\mathrm{k}}^{\mathrm{NS}}}(\boldsymbol{\theta}_{\mathrm{NS}}^*)=m_a^*s^2+m_K^*,$$

where

$$m_K^*=\frac{1}{2}|\{p(\boldsymbol{K}_{lij}^*),-p(\boldsymbol{K}_{lij}^*)|\boldsymbol{K}_{lij}^*\neq\boldsymbol{0},l\in[k],i,j\in[d+1-s]\}|=k(d+1-s)^2,$$

$$m_a^*=\frac{1}{2}|\{p(\boldsymbol{K}_{lij}^*),-p(\boldsymbol{K}_{lij}^*)|l\in[k],i,j\in[d+1-s],a_{lij}\neq 0\}|=k(d+1-s)^2,$$

and p is the padding function over kernels, i.e., for each $(s \times s)$ kernel $\boldsymbol{K}_{lij}$, $p(\boldsymbol{K}_{lij}) \in \mathbb{R}^{d\times d}$ s.t. $p(\boldsymbol{K}_{lij})[i:i+s-1, j:j+s-1] = \boldsymbol{K}_{lij}$ and the other elements of $p(\boldsymbol{K}_{lij})$ are zero. Similar to the proof of Theorem 5, by the upper bound estimate Theorem 3,

$$O_{\mathrm{CNN}_{\mathrm{m}}^{\mathrm{NS}}}(f^*) \leqslant O_{\mathrm{CNN}_{\mathrm{k}}^{\mathrm{NS}}}(f^*) \leqslant \mathrm{rank}_{\mathrm{CNN}_{\mathrm{k}}^{\mathrm{NS}}}(\boldsymbol{\theta}^*_{\mathrm{NS}}) = k(s^2+1)(d+1-s)^2.$$

Also similar to the proof of Theorem 5, for any $\boldsymbol{\theta}'_{\mathrm{NS}} = (a'_{lij}, \boldsymbol{K}'_{lij})_{l\in[m],i,j\in[d+1-s]} \in \mathcal{M}_{f^*}$ with no input-ineffective neuron, by the linear independence of neurons Proposition 3, we have

$$m'_K = \frac{1}{2}|\{p(\boldsymbol{K}'_{lij}), -p(\boldsymbol{K}'_{lij}) | l \in [m], i,j \in [d+1-s], \boldsymbol{K}'_{lij} \neq \boldsymbol{0}\}| \geqslant k(d+1-s)^2,$$

$$m'_a = \frac{1}{2}|\{p(\boldsymbol{K}'_{lij}), -p(\boldsymbol{K}'_{lij}) | l \in [m], i,j \in [d+1-s], a'_{lij} \neq 0\}| \geqslant k(d+1-s)^2.$$

Therefore $\mathrm{rank}_{\mathrm{CNN}_{\mathrm{m}}^{\mathrm{NS}}}(\boldsymbol{\theta}') \geqslant k(s^2+1)(d+1-s)^2$ for $\boldsymbol{\theta}' \in \mathcal{M}_{f^*}$, which yields $O_{\mathrm{CNN}_{\mathrm{m}}^{\mathrm{NS}}}(f^*) = \mathrm{rank}_{\mathrm{CNN}_{\mathrm{m}}^{\mathrm{NS}}}(f^*) \geqslant k(s^2+1)(d+1-s)^2$. Then we obtain $O_{\mathrm{CNN}_{\mathrm{m}}^{\mathrm{NS}}}(f^*) = k(s^2+1)(d+1-s)^2$. ■

Costly expressiveness in connection for two-layer NNs. Drawing from the aforementioned findings, we present a comparative analysis of optimistic sample sizes across various architectures, including CNNs with and without weight sharing, as well as fully-connected NNs, as depicted in Figure 2. It is important to note that, to ensure an equitable comparison, the total number of hidden neurons is held constant $m(d+1-s)^2$ across the different architectures. As demonstrated in Table 3, for a typical image data with dimension $d = 28$, if the target function is recoverable by a CNN with at least $k \leqslant m$ kernels, then the model rank for different NN architectures exhibits significant variation, ranging from $685k$ for a CNN with weight sharing, to $6760k$ for a CNN-NS, and up to $530660k$ for a fully-connected NN. This stark contrast underscores the vast disparities in their target recovery performance especially when the training data is limited.

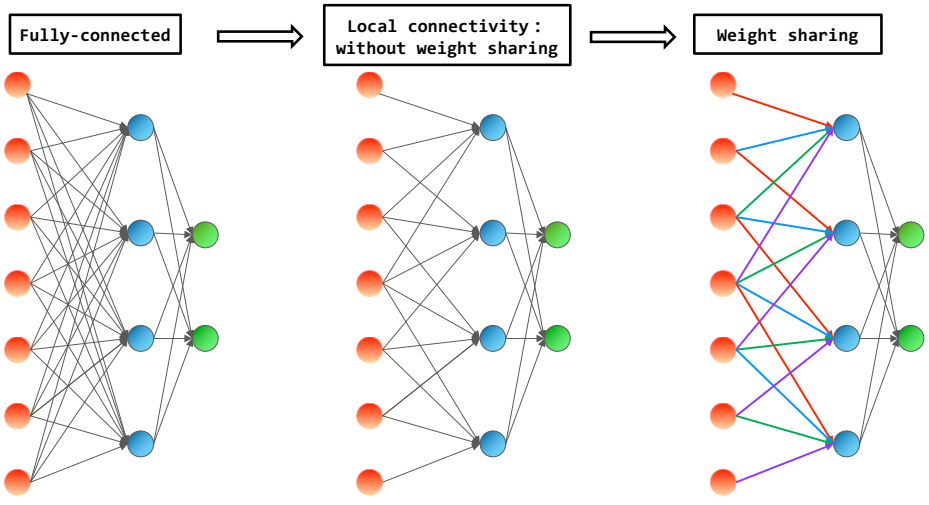


Figure 2: Illustration of architectures from fully-connected NN to CNN for comparison.

f^*	CNN	CNN (no sharing)	Fully-connected NN
$\{0\}$	0	0	0
$\mathcal{G}_1^*$	$s^2+(d+1-s)^2$	$(s^2+1)(d+1-s)^2$	$(d^2+1)(d+1-s)^2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{G}_k^*$	$k(s^2+(d+1-s)^2)$	$k(s^2+1)(d+1-s)^2$	$k(d^2+1)(d+1-s)^2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{G}_m^*$	$m(s^2+(d+1-s)^2)$	$m(s^2+1)(d+1-s)^2$	$m(d^2+1)(d+1-s)^2$

Table 3: The optimistic sample size for two-layer tanh-CNN with m-kernels of size $s \times s$ and stride 1. The input $\boldsymbol{x} \in \mathbb{R}^{d\times d}$. For functions in each all-effective CNN function set (see Definition 10), we present their model rank in the corresponding CNN with and without weight sharing and the corresponding fully-connected NN.

6. Experimental Results

6.1 Optimistic Sample Size Informs Practical Performance

In Figure 3, we conduct experiments to assess the practical significance of the previously estimated optimistic sample sizes in relation to the actual fitting performance of two-layer tanh neural networks (NNs) with varying architectures. Our experiments are centered around the target function defined as:

$$f^*(\boldsymbol{x}) = \boldsymbol{W}^{*[2]} \tanh(\boldsymbol{W}^{*[1]}\boldsymbol{x}), \tag{8}$$

where $\boldsymbol{W}^{*[2]} = [1,1,1]$ and

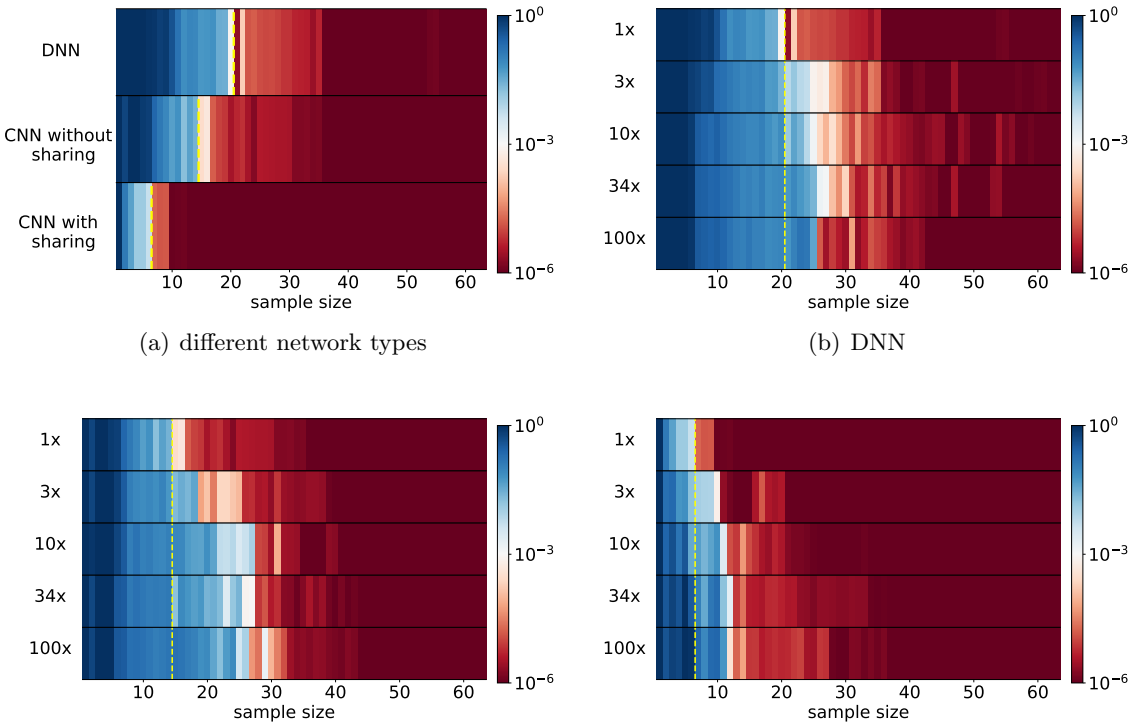
$$\boldsymbol{W}^{*[1]} = \begin{bmatrix} 0.6 & 0.8 & 1 & 0 & 0 \\ 0 & 0.6 & 0.8 & 1 & 0 \\ 0 & 0 & 0.6 & 0.8 & 1 \end{bmatrix}.$$

We generate both training and test data sets by sampling input data from a standard normal distribution and computing the output using the target function. We employ two-layer tanh-NNs, each with a bias term for the hidden neurons, in a variety of architectures and with different kernels/widths, to fit training data sets of sizes ranging from 1 to 63.

It is noteworthy that for a single-kernel CNN, with or without weight sharing, or a fully-connected NN with width 3 (denoted as 1x in Figure 3(b-d)), the model rank coincides with the number of model parameters. Under these conditions, as depicted in Figure 3(a), the CNN demonstrates a notably earlier transition to almost-0 test error compared to other architectures. However, this finding is somewhat expected since recoveries occur within the traditional over-determined/underparameterized regime.

In Figure 3(b-d), we scale up the kernels/widths of the NNs by a factor of N, indicated by Nx for each architecture. For $N = 100$, the parameter counts for the models are 700, 1500, and 2100, respectively. According to Table 3, the model rank for the target remains at

7, 15, and 21 (marked by yellow dashed lines), irrespective of the value of N. In Figure 3(b-d), we observe a postponement in the transition to accurate target recovery for $N > 1$, meaning that the test error decreases to nearly zero at a sample size larger than the model rank. Notably, the transition to almost-0 test error is much closer to the model rank than to the parameter count, particularly for large N values.



(a) different network types

(b) DNN

(c) CNN-NS

(d) CNN

Figure 3: Average test error (color) for NNs of different architectures (ordinate) and sample sizes (abscissa) in fitting the target function Equation (8). The yellow dashed line for each row indicates the model rank of the target in the corresponding NN. (a) Two-layer 1-kernel tanh-CNN vs. two-layer 1-kernel no-sharing tanh-CNN vs. two-layer width-3 fully-connected tanh-NN. Note that these NNs are referred to as 1x for each architecture in (b-d). (b) Two-layer N-kernel tanh-CNN, (c) two-layer N-kernel no-sharing tanh-CNN, and (d) two-layer width-$3N$ fully-connected tanh-NN labeled by Nx for $N = 1, 3, 10, 34, 100$. For all experiments, network parameters are initialized by a normal distribution with mean 0 and variance 10^{-20}, and trained by full-batch gradient descent with a fine-tuned learning rate. For the training data set and the test data set, we construct the input data through the standard normal distribution and obtain the output values from the target function. The size of the training data set varies whereas the size of the test data set is fixed to 1000. The learning rate for the experiments in each setup is fine-tuned from 0.05 to 0.5 for a better generalization performance.

6.2 Enabling Earlier Recovery via Stronger Condensation

In practice, achieving recovery as close as possible to the optimistic sample size remains an important challenge. Fortunately, Remark 2, which details the relation between model rank and condensation, provides valuable insight. It suggests that facilitating condensation, which more strongly prioritizes lower model rank solutions, could benefit the recovery of the target. Based on this insight, dropout—which strongly facilitates condensation (Zhang and Xu, 2024)—should be an effective means to achieve an earlier recovery.

We verify this experimentally in Figure 4, where we compare the fitting performances of neural networks (NNs) trained with and without dropout. The target function in our experiments comprises three tanh functions, defined as:

$$f^*(x) = \tanh(x-7) + \tanh(x) + \tanh(x+7). \tag{9}$$

Both training and test data sets were generated by sampling input data from an equally spaced distribution in the interval $[-15, 14]$ and computing the corresponding outputs using the target function. We employ two-layer tanh-NNs with a width of 300 $f_{\boldsymbol{\theta}}(x) = \sum_{i=1}^{300} a_i \tanh(w_i x + b_i)$ with and without dropout, to fit training data sets of sizes ranging from 1 to 50. In the dropout scenario, neurons were randomly deactivated with a probability of 10% during training.

Figure 4(a) shows that dropout leads to earlier recovery of the target function. To further investigate the parameter state learned by the network at the optimistic sample size (9 for the 3-neuron target), we quantified the degree of neuronal condensation using cosine similarity. The orientation similarity between two neurons was calculated as the inner product of their normalized input weights

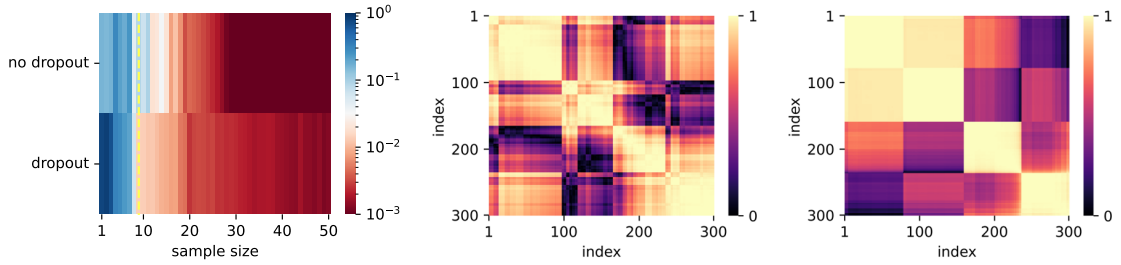
$$s_{ij} = \frac{w_i w_j + b_i b_j}{\sqrt{(w_i^2 + b_i^2)(w_j^2 + b_j^2)}}.$$

Note that we only visualize the absolute value $|s_{ij}|$ since both $s_{ij} = 1$ and $s_{ij} = -1$ indicate exact alignment for tanh activation. As illustrated in Figure 4(b-c), at the optimistic sample size of the target (indicated by the yellow dashed line in Figure 4(a)), the solution obtained with dropout exhibits stronger condensation compared to the one obtained without dropout.

These results further demonstrate that our estimation of the optimistic sample size of the target—based on the parameter point with the lowest model rank, which is also the most condensed one in the target set—well informs the practical performance of deep neural networks under strong condensation.

7. Conclusion

In this study, we have established a local linear recovery (LLR) guarantee for deep neural networks (DNNs), demonstrating that all functions expressible by narrower DNNs possess an LLR-guarantee at overparameterization. Figure 5 presents a schematic overview of our theoretical results and their interconnections. Our work lays a solid groundwork for advancing the recovery theory of DNNs. Suggested by our results for two-layer NNs (Section 5), the linear independence of neurons as in Proposition 3 plays a key role in the exact calculation of model rank. For future research, three significant challenges remain open: (i) exploring



(a) performance comparison (b) no dropout (c) dropout

Figure 4: (a) Heatmap depicting average test error (color scale) for width-300 neural networks with and without dropout (y-axis) across varying sample sizes (x-axis) when fitting the target function defined in Equation (9). Yellow dashed lines indicate the model rank of the target for each network configuration. (b-c) Cosine similarity between input weights of neurons for two-layer tanh neural networks trained without (b) and with (c) dropout. For all experiments, network parameters are initialized by Pytorch default initialization, and trained by full-batch Adam optimizer with a fine-tuned learning rate. For the training data set and the test data set, we construct the input data through a equally spaced sampling in $[-15, 14]$ and obtain the output values from the target function. The size of the training data set varies whereas the size of the test data set is fixed to 1000. The learning rate for the experiments in each setup is fine-tuned from 10^{-3} to 10^{-4} for a better generalization performance. Note that, for the dropout experiments, it is very difficult to achieve less than 10^{-3} test error even with a very large sample size because of the difficulty in achieving low training error $\sim 10^{-4}$.

training methods for an earlier recovery; (ii) investigating whether DNNs can offer stronger forms of recovery guarantees at overparameterization. (iii) exploring neuron independence in deeper networks to exactly estimate their model ranks.

In a more recent work (Zhang et al., 2023a), we make a step further to investigate a stronger form of recovery guarantee, known as the local recovery guarantee, for two-layer neural networks (NNs). This approach employs only the simplification of localization, eschewing linearization. The study of local recovery guarantees demanded a more sophisticated analysis of the geometry and dynamics in the vicinity of global minima. As research in this area progresses, we anticipate a significant deepening of our theoretical understanding of target recovery in DNNs in overparameterized regimes.

Acknowledgments

This work is sponsored by the National Key R&D Program of China Grant No. 2022YFA1008200, the National Natural Science Foundation of China Grant No. 12101402, the Lingang Laboratory Grant No.LG-QS-202202-08, Shanghai Municipal of Science and Technology. We are grateful for the insightful comments provided by Zhi-Qin John Xu and Tao Luo.

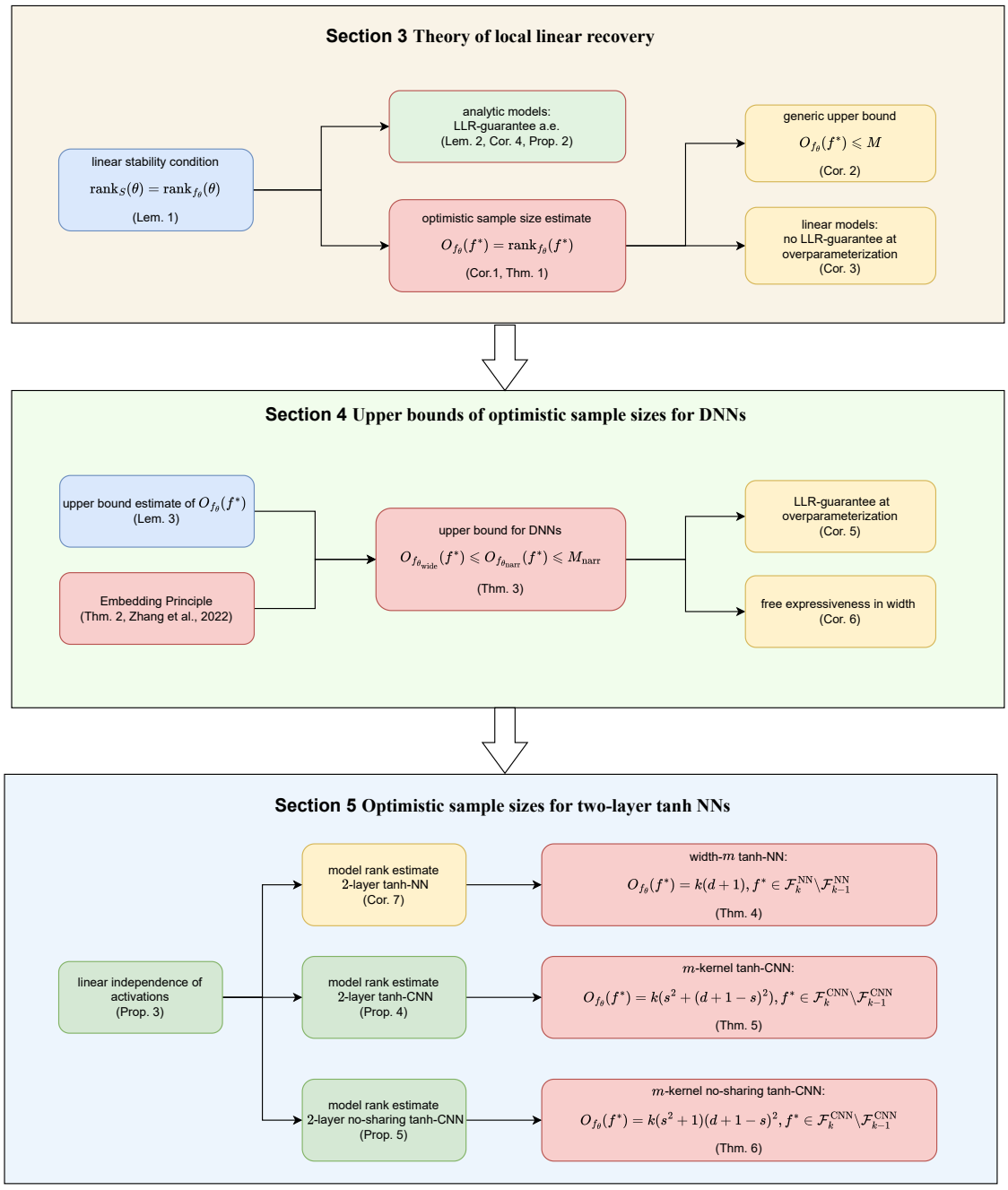


Figure 5: Schematic overview of our theoretical results and interconnections.

References

Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *ICML*, 2019.

Zhiwei Bai, TaoXu Luo, Zhi-Qin John, and Yaoyu Zhang. Embedding principle in depth for the loss landscape analysis of deep neural networks. *CSIAM Transactions on Applied Mathematics*, 5(2):350–389, 2024. ISSN 2708-0579. doi: https://doi.org/10.4208/csiam-am.SO-2023-0020.

Alon Brutzkus and Amir Globerson. Why do larger models generalize better? a theoretical perspective via the xor problem. In *ICML*, 2019.

Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *NeurIPS*, pages 2937–2947, 2019.

Yaim Cooper. Global minima of overparameterized neural networks. *SIAM Journal on Mathematics of Data Science*, 3(2):676–691, 2021.

Kenji Fukumizu, Shoichiro Yamaguchi, Yoh-ichi Mototake, and Mirai Tanaka. Semi-flat minima and saddle points by embedding neural networks to overparameterization. *NeurIPS*, 32:13868–13876, 2019.

Arthur Jacot. Implicit bias of large depth networks: a notion of rank for nonlinear functions. In *ICLR*, 2023.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: convergence and generalization in neural networks. In *NeurIPS*, 2018.

Tao Luo, Zhi-Qin John Xu, Zheng Ma, and Yaoyu Zhang. Phase diagram for two-layer relu neural networks at infinite-width limit. *Journal of Machine Learning Research*, 22:71–1, 2021.

Hartmut Maennel, Olivier Bousquet, and Sylvain Gelly. Gradient descent quantizes relu network features. *arXiv preprint arXiv:1803.08367*, 2018.

B. S. Mityagin. The zero set of a real analytic function. *Mathematical Notes*, 107(3): 529–530, Mar 2020. ISSN 1573-8876.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Claude E Shannon. Communication in the presence of noise. *Proceedings of the IEEE*, 72 (9):1192–1201, 1984.

Berfin Simsek, François Ged, Arthur Jacot, Francesco Spadaro, Clement Hongler, Wulfram Gerstner, and Johanni Brea. Geometry of the loss landscape in overparameterized neural networks: Symmetries and invariances. In *ICML*, pages 9722–9732. PMLR, 2021.

Ruoyu Sun, Dawei Li, Shiyu Liang, Tian Ding, and Rayadurgam Srikant. The global landscape of neural networks: An overview. *IEEE Signal Processing Magazine*, 37(5): 95–108, 2020.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *ICLR*, 2017.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021a.

Leyang Zhang, Yaoyu Zhang, and Tao Luo. Structure and gradient dynamics near global minima of two-layer neural networks. *arXiv preprint arXiv:2309.00508*, 2023a.

Yaoyu Zhang, Zhongwang Zhang, Tao Luo, and Zhi-Qin John Xu. Embedding principle of loss landscape of deep neural networks. *NeurIPS*, 2021b.

Yaoyu Zhang, Yuqing Li, Zhongwang Zhang, Tao Luo, and Zhi-Qin John Xu. Embedding principle: a hierarchical structure of loss landscape of deep neural networks. *Journal of Machine Learning*, 1:1–45, 2022.

Yaoyu Zhang, Zhongwang Zhang, Leyang Zhang, Zhiwei Bai, Tao Luo, and Zhi-Qin John Xu. Optimistic estimate uncovers the potential of nonlinear models. *arXiv preprint arXiv:2307.08921*, 2023b.

Zhongwang Zhang and Zhi-Qin John Xu. Implicit regularization of dropout. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13, 2024.

Kai Zhong, Zhao Song, Prateek Jain, Peter L Bartlett, and Inderjit S Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *ICML*, 2017.

Hanxu Zhou, Qixuan Zhou, Zhenyuan Jin, Tao Luo, Yaoyu Zhang, and Zhi-Qin John Xu. Empirical phase diagram for three-layer neural networks with infinite width. *NeurIPS*, 2022a.

Hanxu Zhou, Qixuan Zhou, Tao Luo, Yaoyu Zhang, and Zhi-Qin John Xu. Towards understanding the condensation of neural networks at initial training. *NeurIPS*, 2022b.