# STRUCTURED VS. UNSTRUCTURED PRUNING: AN EXPONENTIAL GAP

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

The Strong Lottery Ticket Hypothesis (SLTH) states that large, randomly initialized neural networks contain sparse subnetworks capable of approximating a target function at initialization without training, suggesting that pruning alone is sufficient. Pruning methods are typically classified as unstructured, where individual weights can be removed from the network, and structured, where parameters are removed according to specific patterns, as in neuron pruning. Existing theoretical results supporting the SLTH rely almost exclusively on unstructured pruning, showing that logarithmic overparameterization suffices to approximate simple target networks. In contrast, neuron pruning has received limited theoretical attention, despite its practical appeal for direct hardware speedups. In this work, we consider the problem of approximating a single bias-free ReLU neuron by pruning hidden units of a randomly initialized two-layer ReLU network, effectively isolating the intrinsic limitations of neuron pruning. We show that achieving an $\varepsilon$-approximation requires a starting network size of $\Omega(1/\varepsilon)$ for neuron pruning, whereas weight pruning succeeds with only $O(\log(1/\varepsilon))$ hidden units, revealing an exponential separation between the two approaches.

## 1 INTRODUCTION

Modern neural networks are typically trained in highly overparameterized regimes, often containing orders of magnitude more parameters than are seemingly necessary to represent the target function. Despite this redundancy, empirical evidence suggests that such large models can be pruned aggressively after training, yielding sparse subnetworks that retain comparable performance. Understanding the origin of this phenomenon has become a central topic in the theory of deep learning.

A prominent line of work addressing this question is the *Lottery Ticket Hypothesis* (LTH), introduced in Frankle & Carbin (2018). The LTH states that sufficiently large randomly initialized networks contain sparse subnetworks, called *winning tickets*, which can achieve the performance of the original dense network, when trained in isolation. Subsequent empirical works refined this perspective and proposed pruning strategies that identify such subnetworks efficiently (Zhou et al., 2019; Ramanujan et al., 2020).

These developments motivated an even stronger conjecture, known as the *Strong Lottery Ticket Hypothesis* (SLTH), which asserts that winning tickets already exist at initialization, without requiring any training of the retained weights. In other words, pruning alone suffices to obtain a performant subnetwork. This formulation removes the need to analyze training dynamics and has enabled a series of rigorous existence proofs for fully connected and convolutional architectures (Malach et al., 2020; Pensia et al., 2020; Orseau et al., 2020; Ferbach et al., 2022). Given a target network and $\varepsilon > 0$, a central question is how much overparameterization is needed for a randomly initialized network to contain, with high probability, a pruned subnetwork that $\varepsilon$-approximates the target network (see Definition 1 for the precise metric). This dependence on $\varepsilon$ is critical in deep settings: since approximation errors compound across layers, the required accuracy must shrink with the target network's width and depth. Consequently, a pruning scheme with linear dependence on $1/\varepsilon$ becomes prohibitively expensive, while the logarithmic dependence achieved by weight pruning remains compatible with existing SLTH constructions (Pensia et al., 2020).

Most existing theoretical results establishing the SLTH rely on *unstructured pruning*, also known as *weight pruning*, in which individual edges are removed from the network. A key technical insight,



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

first identified in Malach et al. (2020) and later sharpened in Pensia et al. (2020), is that weight pruning allows one to approximate target weights by doing subset sums of randomly initialized weights. By leveraging a classical result of Lueker (1998) on the random subset sum problem, Pensia et al. (2020) showed that a random network with only logarithmic overparameterization—on the order of $O(\log(d/\varepsilon))$ per target weight—contains subnetworks that $\varepsilon$-approximate a broad family of target functions. Matching lower bounds show that this logarithmic dependence is essentially optimal for constant-depth networks (Pensia et al., 2020). While unstructured pruning provides a powerful theoretical framework for establishing the existence of accurate subnetworks, its implications for computational efficiency are less clear. Indeed, unstructured pruning typically produces sparsity patterns with zeros in arbitrary locations in the weight matrices. Such sparsity does not, by itself, translate into predictable speedups on standard hardware, where dense linear algebra routines are optimized for contiguous memory access and regular computation.

The aforementioned gap between parameter sparsity and actual speedups motivates the study of *structured pruning* methods, which enforce sparsity at the level of entire blocks, rows, or columns of weight matrices. Such structure directly translates into smaller matrices and fewer arithmetic operations, yielding genuine reductions in memory usage and inference time. Among structured approaches, the simplest and most natural is *neuron pruning*, in which entire hidden units are removed, effectively deleting rows (and corresponding columns) of the network's weight matrices. Understanding the expressive power and limitations of neuron pruning is therefore essential for bridging theoretical guarantees with practically meaningful efficiency gains. In contrast to weight pruning, the role of neuron pruning has received comparatively little theoretical attention. This is not accidental: Malach et al. (2020) explicitly observed that neuron pruning appears fundamentally weaker than weight pruning, and pointed to results on random feature models to support this claim. In particular, Yehudai & Shamir (2019) showed that there exist target ReLU neurons with bias for which approximation by random features is provably hard: unless one allows either an exponential number of random features or exponentially large coefficients, uniform approximation over standard Gaussian inputs is impossible. Subsequent work strengthened this result by showing with a similar proof strategy that even when the magnitude of the coefficients is unrestricted, approximating such target ReLU neurons still requires an exponential number of random features (Kamath et al., 2020). These impossibility results are proved under the assumption that the target ReLU neuron may have a bias as large as $O(d^4)$, where $d$ is the dimension of the input. It therefore leaves open the possibility that the observed hardness of approximation is driven, at least in part, by this bias assumption rather than by an inherent limitation of neuron-pruning[1]. This raises the following natural questions: *does the inefficiency of neuron pruning persist when one rules out biases? What is the dependence of neuron pruning w.r.t. $\varepsilon$?*

**Our Contribution** In this work, we revisit neuron pruning in a clean and natural setting. We focus on approximating a single ReLU neuron using a larger randomly initialized ReLU network with one hidden layer, where both the target neuron and the hidden neurons have zero bias. This setting isolates the expressive limitations of neuron pruning from confounding effects caused by large biases. Importantly, approximating a single neuron is arguably the simplest nontrivial approximation task, so lower bounds in this regime already indicate fundamental limitations of neuron pruning. We show that neuron pruning fundamentally requires a starting network with $\Omega(d/\varepsilon)$ hidden neurons to successfully $\varepsilon$-approximate a target ReLU neuron, when the output weights are bounded away from zero (Theorem 1). We further show that in the fully random initialization setting, where the output weights are also random, a weaker lower bound of $\Omega(\sqrt{d}\log(d)/\varepsilon)$ still holds (Theorem 2). This establishes an exponential gap between weight and neuron pruning in their dependence on the approximation accuracy $\varepsilon$, as weight pruning achieves an $\varepsilon$-approximation with only $O(d\log(d/\varepsilon))$ units Pensia et al. (2020). This suggests that neuron pruning is not a viable SLTH mechanism in deep networks: in existing constructions, the required approximation accuracy $\varepsilon$ must shrink with the target network's width and depth, making a linear dependence on $1/\varepsilon$ prohibitive, whereas weight pruning only incurs logarithmic dependence on $1/\varepsilon$ (see Pensia et al. (2020)).

At a technical level, to handle the bias-free setting, we adopt a novel proof strategy in which we track the location of hidden units' nonlinearities along carefully constructed input families and derive necessary conditions for approximation. We then reformulate these conditions as hitting events of

[1]The proof in Yehudai & Shamir (2019) relies on the presence of such a bias, and removing this assumption would require a fundamentally different proof strategy, as confirmed by the authors via private communication.

suitable stochastic processes, whose hitting probabilities yield bounds on the success probability of approximation.

## 2 Related Work

After the Lottery Ticket Hypothesis was introduced in Frankle & Carbin (2018), a large body of work has investigated algorithms for identifying such subnetworks, including pruning strategies that learn masks or importance scores for individual weights without changing their initial values (Zhou et al., 2019; Ramanujan et al., 2020; Wang et al., 2020). These empirical results lead to the Strong Lottery Ticket Hypothesis (SLTH), which claims that pruning alone can reveal subnetworks that approximate some target network. The first rigorous SLTH guarantees were obtained by Malach et al. (2020), who showed that sufficiently overparameterized random ReLU networks contain accurate subnetworks with high probability. Subsequent works (Orseau et al., 2020; Pensia et al., 2020) refined these guarantees by reducing the required overparameterization and extending the framework to broader architectural settings. In particular, Pensia et al. (2020) established that logarithmic overparameterization in the approximation accuracy $\varepsilon$ is sufficient for dense networks through connections with the random subset sum problem. Later works extended these results to convolutional and residual architectures (da Cunha et al., 2022; Burkholz, 2022a; da Cunha et al., 2023) and to more general equivariant architectures (Ferbach et al., 2022). Further refinements addressed broader activation functions, reduced depth overhead, and introduced notions such as universal lottery tickets (Burkholz, 2022b; Burkholz et al., 2022; Fischer et al., 2022).

Despite strong theoretical progress for weight pruning, unstructured sparsity does not directly translate into computational speedups on modern hardware. This limitation has motivated extensive research on structured pruning methods, which remove entire blocks of parameters such as channels or neurons. Early works related to structured pruning include classical studies on estimating the relevance of individual hidden units and removing those deemed unimportant (Mozer & Smolensky, 1988; 1989). Since then, structured pruning has developed into a broad research direction, particularly for convolutional architectures; we refer to the surveys in (Hoefler et al., 2021; He & Xiao, 2023) for comprehensive overviews. This line of work predominantly focuses on designing pruning algorithms and studying their empirical efficiency. By contrast, structured pruning has received little attention within the SLTH literature. Some recent works have established SLTH guarantees for some forms of structured convolutional pruning using multidimensional extensions of random subset sum techniques (da Cunha et al., 2023). However, understanding the expressive power of neuron pruning, the simplest structured pruning strategy, remains largely open.

In particular, Malach et al. (2020) observed that neuron pruning appears intrinsically weaker than weight pruning, by using previous results (Yehudai & Shamir, 2019) on random features model. In a random features model, hidden weights are fixed and only output coefficients can be trained. Indeed, if the output coefficients associated with retained neurons were allowed to be refitted after pruning, neuron pruning would reduce to selecting a subset of random features and learning their linear combination. Neuron pruning in the SLTH setting is strictly more restrictive, as both the hidden weights and output coefficients are inherited from the original random network and cannot be changed. Several works have established strong lower bounds for random feature models that highlight intrinsic approximation limitations. In particular, Yehudai & Shamir (2019) showed that approximating certain ReLU neurons with bias using random features requires either exponentially many features or exponentially large coefficients under Gaussian inputs. This result was strengthened in Kamath et al. (2020), which proved exponential lower bounds even when the magnitude of output coefficients is unrestricted. More lower bounds for random feature models were also established in Hsu et al. (2021), with a focus on approximating smooth functions. However, the aforementioned results rely on target neurons with large biases, and it remains unclear to what extent such assumptions are necessary or whether similar barriers arise in bias-free settings. These thus provide indirect evidence that neuron pruning may face fundamental limitations compared to weight pruning.

## 3 Preliminaries and Setup

In this work, vectors are denoted in boldface (e.g., $\mathbf{x}, \mathbf{w}$), while scalars are written in regular font. For vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$, we write $\langle \mathbf{u}, \mathbf{v} \rangle$ for their dot product. We use $\sigma(t) := \max\{t, 0\}$ to denote

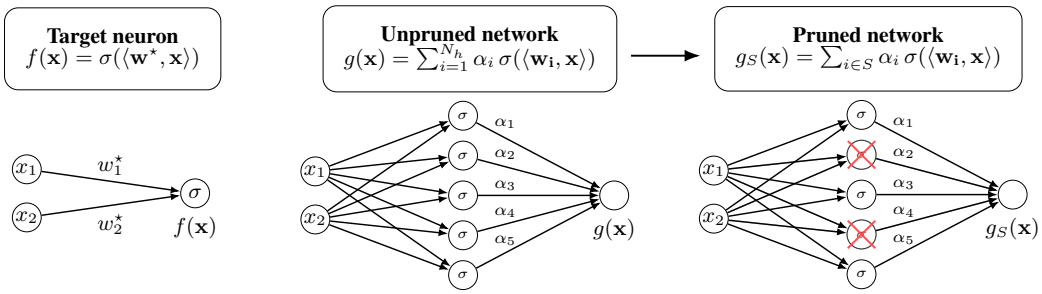


Figure 1: Target ReLU neuron $f$, random network $g$ ($N_h = 5$), and subnetwork $g_S$ pruned from $g$ by retaining a subset $S$ of hidden units. For input dimension $d = 2$, each unit $i$ has incoming weights $\mathbf{w}_i \in \mathbb{R}^2$ (not shown) and output weight $\alpha_i \in \mathbb{R}$.

the ReLU activation function. When we say that a function $h : \mathbb{R} \to \mathbb{R}$ is *linear* on an interval, we mean that it coincides on that interval with an *affine* function of the form $t \mapsto at + b$, for some $a, b \in \mathbb{R}$.

In Section 4, we study the problem of approximating a single target ReLU neuron using a larger randomly initialized neural network $g$, solely by pruning neurons in the hidden layer. Specifically, we consider a bias-free two-layer ReLU network of the form $g(\mathbf{x}) = \sum_{i=1}^{N_h} \alpha_i \, \sigma(\langle \mathbf{w}_i, \mathbf{x} \rangle)$, where $\mathbf{x} \in \mathbb{R}^d$, with hidden-layer neuron weights $\mathbf{w}_i \in \mathbb{R}^d$ and output weights $\alpha_i \in \mathbb{R}$. Figure 1 provides a visual overview of the neuron pruning setting and the notation we use.

The target function we will consider in our work is a single bias-free ReLU neuron $f(\mathbf{x}) := \sigma(\langle \mathbf{w}^\star, \mathbf{x} \rangle)$, with $\|\mathbf{w}^\star\|_2 = \sqrt{d}$. A *neuron-pruned subnetwork* $g_S$ is obtained from $g$ by selecting a subset $S \subseteq \{1, \ldots, N_h\}$ of hidden units and retaining exactly those neurons with all their incident edges, yielding $g_S(\mathbf{x}) = \sum_{i \in S} \alpha_i \, \sigma(\langle \mathbf{w}_i, \mathbf{x} \rangle)$. In line with the Strong Lottery Ticket Hypothesis, pruning is the only operation allowed: the retained weights are not trained or modified.

Throughout the paper, we will often use the following notion of $\varepsilon$-approximation.

**Definition 1** ($\varepsilon$-approximation)**.** *For $\varepsilon > 0$ and radius $R \geq 1$, we say that a function $h$ $\varepsilon$-approximates another function $f$ on the ball of radius $R$ if* $\sup_{\|x\|_2 \leq R} |h(x) - f(x)| \leq \varepsilon$.

## 4 Main Results

Theorem 1 shows that, in a one hidden layer random network with output weight bounded away from zero, neuron pruning alone cannot even $\varepsilon$-approximate a single target ReLU neuron without bias, unless the initial width is $\Omega(d/\varepsilon)$. This section presents a sketch of the proof of the theorem, in which we introduce the main ideas and provide an overview of the argument, referring the reader to Section 5 for the complete, detailed proof.

For simplicity, in Theorem 1 we fix the output weights of network $g$ to be equal to 1. The same proof strategy extends to output-weight distributions that are uniformly bounded away from zero, such as Rademacher; see Appendix C for the corresponding adaptation. Also, we require input dimension $d \geq 2$; for $d = 1$, neuron pruning reduces to weight pruning, and we can directly use the pruning lower bound proved in Pensia et al. (2020). The requirement that $\|\mathbf{w}^\star\|_2 = \sqrt{d}$ is mainly to simplify the notation in the proof; if, instead, we assume $\|\mathbf{w}^\star\|_2 = 1$, then with straightforward modifications one obtains the lower bound $N_h \geq \min\{c_0 \frac{\sqrt{d}}{\varepsilon}, \, 2^{c_1 d}\}$.

**Theorem 1.** *Let $d \geq 2$, $\varepsilon \in (0, 1)$. Consider a one hidden-layer ReLU network without bias and unit output weights of the form $g(\mathbf{x}) = \sum_{i=1}^{N_h} 1 \cdot \sigma(\langle \mathbf{w_i}, \mathbf{x} \rangle)$, where the weights $\{\mathbf{w_i}\}_{i=1}^{N_h}$ are drawn independently from $\mathcal{N}(0, I_d)$, and $\mathbf{x} \in \mathbb{R}^d$. Then, there exist a target neuron with weight vector $\mathbf{w}^\star \in \mathbb{R}^d$ with $\|\mathbf{w}^\star\|_2 = \sqrt{d}$, and universal constants $c_0, c_1 > 0$ such that, if*

$$N_h \; < \; \min\left\{c_0 \tfrac{d}{\varepsilon}, \, 2^{c_1 d}\right\},$$

*then with probability at least* $1-e^{-\Omega(d)}$ *over the draw of* $\{\mathbf{w_i}\}_{i=1}^{N_h}$*, for every subset* $S \subseteq \{1,\dots,N_h\}$ *and every fixed constant* $R > 2$*,*

$$\sup_{\|\mathbf{x}\|_2 \le R} \left| \sum_{i\in S} \sigma(\langle \mathbf{w_i}, \mathbf{x}\rangle) \;-\; \sigma(\langle \mathbf{w}^\star, \mathbf{x}\rangle) \right| \;\ge\; C\varepsilon,$$

*for some universal constant* $C > 0$*.*

*Proof Idea.* As neuron pruning may retain any number $k \in \{1,\dots,N_h\}$ of hidden units, we begin in Section 5.1 by controlling the probability of successful approximation uniformly over all possible pruned network sizes. This is achieved via a union bound over $k$, which reduces the analysis to bounding, for a fixed $k$, the probability $p_k$ that a random network with $k$ hidden units $\varepsilon$-approximates the target.

Our analysis is then based on tracking the evolution of *breakpoints*—the locations at which ReLU activations of the hidden neurons change slope—along carefully chosen one-dimensional input families, as described in Section 5.2. When a high-dimensional ReLU network is restricted to such paths, both the target function and the approximating network reduce to piecewise-linear functions whose behavior is entirely determined by the number, location, and interaction of these breakpoints. Crucially, these breakpoints impose necessary conditions for approximation. Definition 2 formalizes when a breakpoint induces a non-negligible approximation error. Lemma 1 shows that the presence of a breakpoint away from the target nonlinearity prevents $\varepsilon$-approximation, while Lemma 2 establishes that at least one breakpoint must be placed within an $\varepsilon$-neighborhood of the target nonlinearity. Together, these results imply that successful neuron pruning requires introducing exactly one suitably aligned breakpoint and canceling all others.

We formalize this constraint by viewing neuron pruning as a stochastic sequential process in which neurons are selected one by one and the number of unresolved breakpoints evolves over time. The resulting process, faithfully captures the breakpoint dynamics induced by pruning. To make this process tractable, we construct a sequence of couplings: first to a capped process that limits the total number of breakpoints, and then to a homogeneous birth–death process (Sections 5.4 and 5.5). Each coupling is designed so that the dominating process is strictly more favorable to successful approximation, and therefore yields an upper bound on the success probability of the original pruning process.

The above argument bounds allows us to derive an upper bound for the $\varepsilon$-approximation of a single one-dimensional input family (Lemma 7). To obtain dimension-dependent bounds, we exploit the fact that our construction uses $\lfloor d/2 \rfloor$ input families supported on disjoint coordinate pairs. The induced breakpoint processes are independent across these families, and therefore the overall success probability decays exponentially in the input dimension $d$ (Lemma 8); see also Figure 2 in Appendix A for a visual intuition.

Finally, Section 5.6 combines these bounds on $p_k$ with the initial union bound over all pruned subnetworks, yielding the stated lower bound on the number of hidden neurons required for neuron pruning to achieve $\varepsilon$-approximation with non-negligible probability. □

We also consider a variant of Theorem 1 in which the output weights are randomly initialized. In this setting, we obtain a lower bound on the required width $N_h$ with the same dependence on $1/\varepsilon$, while the scaling in the input dimension is weaker. The proof strategy is substantially different from that of Theorem 1, and it relies on a simpler necessary condition for approximation. The proof is provided in Appendix D. In Theorem 2, if we instead consider a target weight vector with unit norm $\|\mathbf{w}^\star\|_2 = 1$, the lower bound on $N_h$ loses a factor of $\sqrt{d}$, scaling instead as $\log d/\varepsilon$.

**Theorem 2.** *Let* $d \ge 2$*,* $R > 2$*,* $\varepsilon \in (0,1)$*,* $\delta \in (0,1)$*, and let* $\mathbf{w}^\star \in \mathbb{R}^d$ *with* $\|\mathbf{w}^\star\|_2 = \sqrt{d}$*. Consider* $g(\mathbf{x}) = \sum_{j=1}^{N_h} \alpha_j\, \sigma(\langle \mathbf{w}_j, \mathbf{x}\rangle)$*, where* $\mathbf{w}_j \sim \mathcal{N}(0, I_d)$ *and* $\alpha_j \sim \mathcal{N}(0,1)$ *are independent. Then there exist a constant* $c_0 = c_0(\delta) > 0$ *such that, if*

$$N_h < c_0 \frac{\sqrt{d}\log d}{\varepsilon},$$

*then with probability at least* $1-\delta$ *over the initialization, for every* $S \subseteq \{1,\dots,N_h\}$*,* $\sup_{\|\mathbf{x}\|_2 \le R} \left| \sum_{j\in S} \alpha_j\, \sigma(\langle \mathbf{w}_j, \mathbf{x}\rangle) - \sigma(\langle \mathbf{w}^\star, \mathbf{x}\rangle) \right| \ge \varepsilon/4.$

# 5 PROOF OF THEOREM 1

In this section, we prove Theorem 1. Because of space constraints, some proofs of intermediate results are omitted and given in Appendix B.

## 5.1 UNION BOUND OVER ALL PRUNED SUBNETWORKS

For any $k \in \{1, \dots, N_h\}$, pruning $g$ down to $k$ hidden neurons produces $\binom{N_h}{k}$ distinct subnetworks. Let $E_k$ be the event that there is a $k$-neuron pruned subnetwork that $\varepsilon$-approximates the target, and let $p_k := \Pr(\text{a fixed } k\text{-neuron network } \varepsilon\text{-approximates the target})$. Then, by a union bound,

$$\Pr\Big(\bigcup_{k=1}^{N_h} E_k\Big) \;\le\; \sum_{k=1}^{N_h} \binom{N_h}{k}\, p_k. \tag{1}$$

The rest of the proof is devoted to quantifying the probability $p_k$.

## 5.2 RESTRICTION OVER SIMPLE INPUT FAMILIES

We introduce some simple families of inputs that will be used to derive necessary conditions on the function $g$ for neuron pruning to succeed. For each $i \in \{1, \dots, \lfloor d/2 \rfloor\}$, define the input family $\mathbf{x}_i(t) := t\mathbf{e}_{2i-1} + \mathbf{e}_{2i}$ for $t \in \mathbb{R}$, where $\mathbf{e}_j$ denotes the $j$-th standard basis vector. This family has exactly two nonzero coordinates: a variable entry $t$ at index $2i-1$ and a fixed entry 1 at index $2i$. There are $\lfloor d/2 \rfloor$ such families, supported on disjoint pairs of coordinates.

Recall that $\mathbf{w}^\star = (w_1^\star, \dots, w_d^\star) \in \mathbb{R}^d$ denotes the target weight vector. We choose $\mathbf{w}^\star = (1, \dots, 1)^\top$. Along the input path $\mathbf{x}_i(t)$, the output of the target neuron is

$$f(\mathbf{x}_i(t)) = \sigma(\langle \mathbf{w}^\star, \mathbf{x}_i(t) \rangle) = \sigma\big(w_{2i-1}^\star\, t + w_{2i}^\star\big),$$

which defines a one-dimensional piecewise-linear function of $t$ with a single *breakpoint* $t_i^\star := -w_{2i}^\star / w_{2i-1}^\star = -1$. On one side of $t_i^\star$ the function is affine with slope $w_{2i-1}^\star = 1$, and on the other side it is identically zero. Similarly, for the randomly initialized network $g(x)$, for each hidden neuron $j$ with an incoming weight vector $\mathbf{w_j} = (w_{j,1}, \dots, w_{j,d})$, and for each input family $\mathbf{x}_i(t)$, we have

$$\sigma(\langle \mathbf{w_j}, \mathbf{x}_i(t) \rangle) = \sigma\big(w_{j,\,2i-1}\, t + w_{j,\,2i}\big).$$

Each hidden neuron $j$ therefore introduces its own breakpoint $t_{i,j} := -w_{j,\,2i} / w_{j,\,2i-1}$ for the input family $\mathbf{x}_i(t)$. Consequently, given a subset $S \subseteq \{1, \dots, N_h\}$ of hidden neurons, the restriction $g_S(\mathbf{x}_i(t))$ of the pruned network $g_S$ on the input family $\mathbf{x}_i(t)$ yields a piecewise-linear function of $t$, whose breakpoints are induced by the neurons in $S$.

## 5.3 BREAKPOINTS AND NECESSARY CONDITIONS FOR APPROXIMATION

To analyze the probability $p_k$, it is convenient to view neuron pruning as a sequential selection process in which the $k$ neurons that are kept in the final subnetwork are chosen one at a time.

Fix an index $i \in \{1, \dots, \lfloor d/2 \rfloor\}$ and consider the restriction of the network to the input family $\mathbf{x}_i(t)$ introduced above, in which each hidden neuron induces a single breakpoint at a location determined by its weights. As neurons are selected sequentially, each newly added neuron can have one of the following effects on the breakpoint structure of the current partial sum of chosen neurons for the input family $\mathbf{x}_i(t)$: (i) it may introduce a new breakpoint, (ii) it may cancel an existing breakpoint through interaction with previously selected neurons, or (iii) it may leave the breakpoint structure unchanged. After all $k$ neurons have been selected, the restriction of the resulting pruned network to $\mathbf{x}_i(t)$ is a piecewise-linear function whose breakpoints are entirely determined by the selected neurons.

Since we are interested in inputs $x$ satisfying $\|x\|_2 \le R$, for an input family $\mathbf{x}_i(t)$ this restriction implies $|t| \le \sqrt{R^2 - 1}$. Consequently, only breakpoints lying in the interval $I_R = [-\sqrt{R^2-1}, \sqrt{R^2-1}]$ can affect the approximation error on bounded inputs. Breakpoints outside this interval have no effect and will therefore be ignored and not counted.

We partition $I_R$ into $\Theta(R/\varepsilon)$ subintervals (bins) of length $\varepsilon$ (up to at most one shorter boundary interval), and denote this partition by $\mathcal{P}_\varepsilon$. The mere presence of a breakpoint does not by itself

make a bin broken: several small slope changes may still leave the function uniformly close to an affine function at the scale relevant to the approximation guarantee. We therefore use the following quantitative definition. The universal threshold $c_{\mathrm{br}} > 0$ is fixed below in Lemma 6.

**Definition 2** (Broken bin). *Let $I_B \subset \mathbb{R}$ be an interval (bin) of length $\varepsilon$, and let $h : I_B \to \mathbb{R}$ be a function. We say that $I_B$ is* broken *for $h$ if there exist three points $t_1 < t_2 < t_3$ in $I_B$ such that for every affine function $\ell(t) = at + b$, $\max_{j=1,2,3} |h(t_j) - \ell(t_j)| \;\geq\; c_{\mathrm{br}}\,\varepsilon$*

By Lemma 1 and Lemma 2, $\varepsilon$-approximation along the path $\mathbf{x}_i(t)$ requires the breakpoint structure of the pruned network to be aligned with that of the target: after selecting $k$ neurons, the only possible broken bin is the one containing the target breakpoint $t_i^\star$, and no additional broken bins may occur.

**Lemma 1** (Broken bin prevents approximation). *Let $f(t) := \sigma(at+b)$, with $a \neq 0$, and let $I_B \subset \mathbb{R}$ be an interval of length $\varepsilon$ such that the breakpoint $t^\star := -b/a$ satisfies $t^\star \notin I_B$. Let $h : \mathbb{R} \to \mathbb{R}$ be any function. If $I_B$ is broken for $h$ in the sense of Definition 2, then $\sup_{t\in I_B} |f(t) - h(t)| \;\geq\; c_{\mathrm{br}}\,\varepsilon$.*

**Lemma 2** (A breakpoint is necessary for approximation). *Let $f(t) := \sigma(at + b)$, $a \neq 0$, and let $t^\star := -b/a$ be its breakpoint. Let $h : \mathbb{R} \to \mathbb{R}$ be a function that is linear on an interval $I$ containing $[t^\star - \delta,\, t^\star + \delta]$ for some $\delta > 0$. Then, $\sup_{t\in[t^\star-\delta,\, t^\star+\delta]} |f(t) - h(t)| \;\geq\; |a|\,\delta/4$.*

It is convenient to rephrase the necessary $\varepsilon$-approximation conditions derived from Lemma 1 and Lemma 2 in an alternative form that simplifies the following analysis. Rather than viewing the goal as approximating the target ReLU itself, we conceptually start from a mirrored version of the target along $\mathbf{x}_i(t)$, obtained by flipping its output sign while preserving the same breakpoint location $t_i^\star$. This mirrored function induces exactly one broken bin, namely the bin containing $t_i^\star$. A neuron whose breakpoint lies in this same bin and whose slope jump is suitably close to the target's slope will cancel the mirrored breakpoint, so that their sum is linear on the bin and hence the bin becomes unbroken.

From this viewpoint, successful $\varepsilon$-approximation on the input family $\mathbf{x}_i(t)$ implies starting from a configuration with a single broken bin (in which the target breakpoint lies) and requiring that, after selecting $k$ neurons, no broken bins remain. We adopt this reformulation throughout the remainder of the proof: the bin containing $t_i^\star$ is treated as initially broken, and we consider approximation to be successful only if after selecting $k$ neurons we have no broken bins.

Using this reformulation, we now describe the sequential selection of $k$ neurons as a stochastic process that tracks the number of broken bins. As we will introduce other stochastic processes later in the proof, we will refer to this one as the *original pruning process*. For each step $s = \{1, \ldots, k\}$, a new neuron is added and it may increase by one, decrease by one, or leave unchanged the number of broken bins in $\mathcal{P}_\varepsilon$, depending of where its breakpoint falls. Since we treat the bin containing $t_i^\star$ as initially broken, the starting number of broken bins at step $s = 0$ will be one.

**Definition 3.** *We define $B_s^{\mathrm{orig}}$ as the number of broken bins after having selected $s$ neurons in the original pruning process, with $B_0^{\mathrm{orig}} = 1$.*

As noted above, for a successful $\varepsilon$-approximation on the input family $\mathbf{x}_i(t)$, we want to have $B_k^{\mathrm{orig}} = 0$. Now consider all $\lfloor d/2 \rfloor$ disjoint input families $\mathbf{x}_i(t)$. For the overall approximation to be successful, we need to have $B_k^{\mathrm{orig}} = 0$ for all such families. By construction, the breakpoints $t_{i,j}$ are independent across all $\lfloor d/2 \rfloor$ families $\mathbf{x}_i(t)$, so $p_k \leq \prod_{i=1}^{\lfloor d/2 \rfloor} \Pr(B_k^{\mathrm{orig}} = 0)$, which gives the following.

**Lemma 3.** *It holds $p_k \leq (\Pr(B_k^{\mathrm{orig}} = 0))^{\lfloor d/2 \rfloor}$.*

### 5.4 Construction of a dominating capped process

In this section, we introduce a simplified process that stochastically dominates the original pruning process described in Section 5.3, in the sense that it produces fewer broken bins. Analyzing this simplified process will yield an upper bound on the probability that the original pruning procedure succeeds. As in the previous section, fix an index $i \in \{1, \ldots, \lfloor d/2 \rfloor\}$ and consider the restriction of the network to the input family $\mathbf{x}_i(t)$.

Fix a positive integer $T$. We define the *capped process* directly from the original process by

$$B_s^{\mathrm{cap}} := \min\{B_s^{\mathrm{orig}}, T\}, \qquad s \in \{0, \dots, k\}.$$

Thus, $B_s^{\mathrm{cap}} = T$ means that at least $T$ bins are actually broken; it does not discard the underlying breakpoint configuration. Since adding one neuron introduces only one breakpoint on a fixed input family, both the original and capped counts change by at most one at each step.

By construction, after having selected $s$ shared neurons, if we look at the capped and original processes we have

$$B_s^{\mathrm{cap}} \le B_s^{\mathrm{orig}}, \qquad \forall s \in \{1, \dots, k\} \tag{2}$$

The capped process is more favorable to maintaining a small number of broken bins. Moreover, $B_s^{\mathrm{cap}} = 0$ if and only if $B_s^{\mathrm{orig}} = 0$. In particular, we have the following identity for the probability of a successful $\varepsilon$-approximation for a random network with $k$ hidden neurons.

**Lemma 4.** *It holds* $\Pr(B_k^{\mathrm{orig}} = 0) = \Pr(B_k^{\mathrm{cap}} = 0)$.

### 5.5 Construction of a dominating birth-death process

In the capped process described in Section 5.4, the probability that a selected neuron creates a new broken bin depends on the current number and locations of broken bins. For example, this probability is high when few bins are broken and lower when most are already broken. This dependence makes the exact process difficult to analyze. To simplify the analysis, we introduce a homogeneous birth-death process that we will use to upper bound $\Pr(B_k^{\mathrm{cap}} = 0)$.

**Definition 4.** *A $(q, p, T)$-chain $(B_s^{\mathrm{bd}})_{s \ge 0}$ is a birth-and-death process on a state space $\{0, 1, \dots, T\}$ such that $B_0^{\mathrm{bd}} = 1$ and*

- *from any state $b < T$, it transitions from state $b$ to $b + 1$ with probability $p$;*
- *from any state $b > 0$, it transitions from state $b$ to $b - 1$ with probability $q$;*
- *otherwise, it remains at $b$.*

We choose the birth and death probabilities so that the homogeneous chain is more favorable to successful approximation than the capped process. More precisely, we construct a coupling for which

$$B_s^{\mathrm{bd}} \le B_s^{\mathrm{cap}}, \qquad \forall s \in \{1, \dots, k\} \tag{3}$$

even though the transition probabilities of the capped process may depend on its complete breakpoint history rather than only on its current count.

**Lemma 5.** *Let $H_s$ denote the complete history after $s$ neurons have been selected. Suppose there are constants $p, q > 0$, with $p + q \le 1$, such that for every possible history,*

$$\Pr(B_{s+1}^{\mathrm{cap}} = B_s^{\mathrm{cap}} + 1 \mid H_s) \ge p \qquad \text{whenever } B_s^{\mathrm{cap}} < T, \tag{4}$$

$$\Pr(B_{s+1}^{\mathrm{cap}} = B_s^{\mathrm{cap}} - 1 \mid H_s) \le q \qquad \text{for every } H_s. \tag{5}$$

*Then the capped process and the $(q, p, T)$-chain admit a coupling satisfying eq. (3). Consequently,* $\Pr(B_k^{\mathrm{cap}} = 0) \le \Pr(B_k^{\mathrm{bd}} = 0)$.

The proof, given in Appendix B.2, conditions on the complete history and constructs the coupling explicitly.

We choose the cap to be a sufficiently small constant fraction of the total number of bins. Namely, for a sufficiently large constant $C_{\mathrm{cap}} > 0$, let

$$T = \max\left\{1, \left\lfloor \frac{R}{C_{\mathrm{cap}} \varepsilon} \right\rfloor\right\}. \tag{6}$$

The harmless rounding, as well as the regime in which the maximum equals one, can be absorbed into the universal constants below.

The next lemma supplies the geometric fact needed for a history-uniform birth bound. Its hypothesis permits the unbroken bin to contain an arbitrary collection of pre-existing breakpoints.

**Lemma 6** (Adding a sufficiently steep neuron breaks an unbroken bin)**.** *There exist universal constants $\gamma, c_{\mathrm{br}} > 0$, with $c_{\mathrm{br}} \leq \gamma/32$, such that the following holds. Let $B = [u, u + \varepsilon]$, and let $h : B \to \mathbb{R}$ be a piecewise-linear function for which $B$ is unbroken. Let $r : B \to \mathbb{R}$ be piecewise-linear with exactly one breakpoint $t_0 \in [u + \varepsilon/4, u + 3\varepsilon/4]$ and slope jump $\Delta$. If $|\Delta| \geq \gamma$, then $B$ is broken for $h + r$ in the sense of Definition 2.*

The complete history-uniform verification of the birth and death bounds is given in Appendix B.1. It yields constants $p = p_0$ and $q = q_0$ with $q_0 < p_0/2$ and $p_0 \leq 1/3$, so the hypotheses of Lemma 5 hold.

The resulting birth–death process is coupled below the capped process and is therefore more favorable to successful approximation. It thus provides an upper bound on the probability that neuron pruning yields a network $g_S$ that $\varepsilon$-approximates the target ReLU.

We can now finally bound the probability $\Pr(B_k^{\mathrm{bd}} = 0)$ of such a $(p, q, T)$-chain to be in state 0 after $k$ steps, which by Lemma 5 gives us an upper bound on the probability of a successful $\varepsilon$-approximation.

A crucial part of the proof is now to show the following, which we prove via an elegant use of total probabilities. Because of space constraints, the proof is deferred to the appendix.

**Lemma 7.** *It holds* $\Pr(B_k^{\mathrm{bd}} = 0) \;\leq\; e^{-\Omega(\min(k,T))}$.

With Lemma 7, we now have all the ingredients to establish an exponential upper bound for $p_k$. Combining Lemmas 3 to 5 and 7 yields the following.

**Lemma 8.** *It holds* $p_k \leq e^{-\Omega(\min(k,T)d)}$.

Lemma 8 gives the desired exponentially small upper bound on the probability of successful $\varepsilon$-approximation, as shown in the next section.

### 5.6 Back to the union bound

We now conclude the proof by returning to the union bound of eq. (1). We split the sum in two, depending on the value of $k$:

$$\textstyle\sum_{k=1}^{N_h} \binom{N_h}{k} p_k = \sum_{k=1}^{T-1} \binom{N_h}{k} p_k + \sum_{k=T}^{N_h} \binom{N_h}{k} p_k \tag{7}$$

From Lemma 8, we can prove the following.

**Lemma 9.** *It holds* $\sum_{k=1}^{T-1} \binom{N_h}{k} p_k \leq e^{-\Omega(d)}$ *and* $\sum_{k=T}^{N_h} \binom{N_h}{k} p_k \leq e^{-\Omega(d)}$.

From Lemma 9, we get that eq. (7) is at most $e^{-\Omega(d)}$, proving that with probability at least $1 - e^{-\Omega(d)}$, no neuron-pruned subnetwork $\varepsilon$-approximates the target ReLU uniformly on the ball of radius $R$. This completes the proof of Theorem 1.

## 6 Conclusion and Future Work

This work establishes a fundamental separation between neuron pruning and weight pruning within the framework of the Strong Lottery Ticket Hypothesis. While weight pruning is known to $\varepsilon$-approximate a single ReLU neuron with logarithmic dependence on $1/\varepsilon$, we show that neuron pruning necessitates $\Omega(1/\varepsilon)$ hidden neurons. Our results therefore identify neuron pruning as a strictly weaker pruning mechanism from an approximation-theoretic perspective, even in the clean, bias-free setting.

An open question is the optimal dependence on the input dimension $d$. Our lower bound in Theorem 1 is linear in $d$ (when $\varepsilon = \Omega(2^{-cd})$), whereas random-feature lower bounds suggest that neuron pruning may require exponentially many neurons. In Appendix E, we prove such an exponential lower bound when only one neuron may be retained, suggesting that this trivial strategy could be asymptotically optimal. Extending the dimension-dependent lower bound, as well as our results for deeper architectures and other activations, remains future work.

### AI USE STATEMENT

Generative AI tools were used for LaTeX reformatting, editorial revision, and in assisting the development of some theoretical arguments. All mathematical statements, proofs, and final wording were reviewed and verified by the authors, who take full responsibility for the content.

### REPRODUCIBILITY STATEMENT

We state explicitly all assumptions used in our analysis and provide full proofs for all our results. Proofs that do not fit in the main text can be found in the appendix.

## REFERENCES

Rebekka Burkholz. Convolutional and Residual Networks Provably Contain Lottery Tickets. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 2414–2433, Baltimore, July 2022a. PMLR.

Rebekka Burkholz. Most Activation Functions Can Win the Lottery Without Excessive Depth. In *Thirty-Sixth Conference on Neural Information Processing Systems*, December 2022b.

Rebekka Burkholz, Nilanjana Laha, Rajarshi Mukherjee, and Alkis Gotovos. On the Existence of Universal Lottery Tickets. In *International Conference on Learning Representations*, virtual, April 2022.

Arthur da Cunha, Emanuele Natale, and Laurent Viennot. Proving the Strong Lottery Ticket Hypothesis for Convolutional Neural Networks. In *ICLR 2022 - 10th International Conference on Learning Representations*, Virtual, France, April 2022.

Arthur da Cunha, Francesco D'Amore, and Emanuele Natale. Polynomially Over-Parameterized Convolutional Neural Networks Contain Structured Strong Winning Lottery Tickets. In *Thirty-Seventh Conference on Neural Information Processing Systems*, November 2023.

Damien Ferbach, Christos Tsirigotis, Gauthier Gidel, and Joey Bose. A General Framework For Proving The Equivariant Strong Lottery Ticket Hypothesis. In *The Eleventh International Conference on Learning Representations*, September 2022.

Jonas Fischer, Advait Gadhikar, and Rebekka Burkholz. Lottery Tickets with Nonzero Biases, June 2022.

Jonathan Frankle and Michael Carbin. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. In *International Conference on Learning Representations*, September 2018.

Yang He and Lingao Xiao. Structured pruning for deep convolutional neural networks: A survey. *CoRR*, abs/2303.00566, 2023. doi: 10.48550/arXiv.2303.00566. URL https://doi.org/10.48550/arXiv.2303.00566.

Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *Journal of Machine Learning Research*, 22(241):1–124, 2021. URL http://jmlr.org/papers/v22/21-0366.html.

Daniel Hsu, Clayton Sanford, Rocco A. Servedio, and Emmanouil V. Vlatakis-Gkaragkounis. On the approximation power of two-layer networks of random ReLUs. In Mikhail Belkin and Samory Kpotufe (eds.), *Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 2423–2461. PMLR, 2021. URL http://proceedings.mlr.press/v134/hsu21a.html.

Pritish Kamath, Omar Montasser, and Nathan Srebro. Approximate is good enough: Probabilistic variants of dimensional and margin complexity. In *Conference on Learning Theory*, pp. 2236–2262. PMLR, 2020.

George S. Lueker. Exponentially small bounds on the expected optimum of the partition and subset sum problems. *Random Structures & Algorithms*, 12:51–62, 1998.

Eran Malach, Gilad Yehudai, Shai Shalev-shwartz, and Ohad Shamir. Proving the lottery ticket hypothesis: Pruning is all you need. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20, pp. 6682–6691. JMLR.org, July 2020.

Michael C. Mozer and Paul Smolensky. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In *Advances in Neural Information Processing Systems 1*, pp. 107–115. MIT Press, 1988.

Michael C. Mozer and Paul Smolensky. Using relevance to reduce network size automatically. *Connection Science*, 1(1):3–16, 1989.

Laurent Orseau, Marcus Hutter, and Omar Rivasplata. Logarithmic pruning is all you need. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, pp. 2925–2934, Red Hook, NY, USA, December 2020. Curran Associates Inc. ISBN 978-1-71382-954-6.

Ankit Pensia, Shashank Rajput, Alliot Nagle, Harit Vishwakarma, and Dimitris Papailiopoulos. Optimal lottery tickets via subsetsum: logarithmic over-parameterization is sufficient. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Vivek Ramanujan, Mitchell Wortsman, Aniruddha Kembhavi, Ali Farhadi, and Mohammad Rastegari. What's Hidden in a Randomly Weighted Neural Network? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11890–11899, June 2020. doi: 10.1109/CVPR42600.2020.01191.

Yulong Wang, Xiaolu Zhang, Lingxi Xie, Jun Zhou, Hang Su, Bo Zhang, and Xiaolin Hu. Pruning from scratch. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(7):12273–12280, 2020. doi: 10.1609/aaai.v34i07.6910. URL https://ojs.aaai.org/index.php/AAAI/article/view/6910.

Gilad Yehudai and Ohad Shamir. On the power and limitations of random features for understanding neural networks. *CoRR*, abs/1904.00687, 2019. URL http://arxiv.org/abs/1904.00687.

Hattie Zhou, Janice Lan, Rosanne Liu, and Jason Yosinski. Deconstructing Lottery Tickets: Zeros, Signs, and the Supermask. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019 (NIPS 2019)*, pp. 3592–3602, 2019.

# A Additional Figures

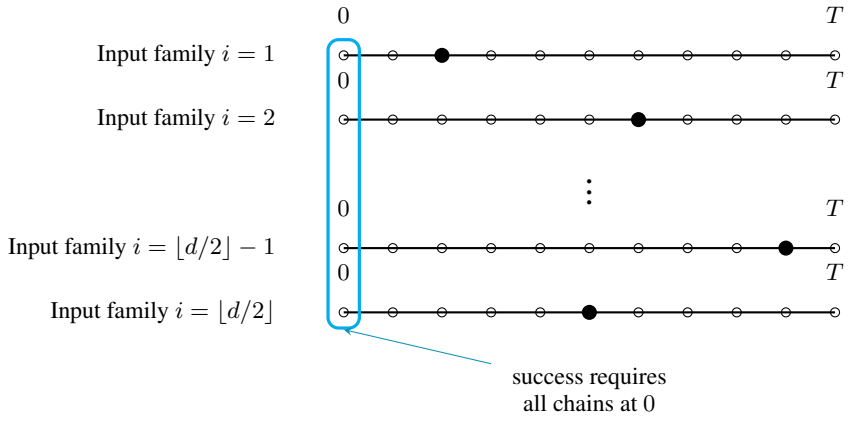


Figure 2: Stacked state-line representation of the birth–death chain $(B_s^{\mathrm{bd}})_{s\geq 0}$. Each horizontal line is the state space $\{0, 1, \ldots, T\}$ for one input family, and the filled dot marks the current value of $B_s^{\mathrm{bd}}$. A necessary condition for $\varepsilon$-approximation is that all $\lfloor d/2 \rfloor$ independent chains reach state 0.

# B Omitted Proofs

## B.1 History-uniform transition bounds

Fix any complete history $H_s$ for which $B_s^{\mathrm{cap}} < T$. Then $B_s^{\mathrm{orig}} = B_s^{\mathrm{cap}} < T$, so all but a sufficiently small constant fraction of the bins are genuinely unbroken. Conditioned on $H_s$, these bins and their internal breakpoint configurations are fixed, whereas the next neuron is independent of the history. Its breakpoint location and slope jump have a joint density that is continuous and strictly positive on the compact range obtained by restricting the jump magnitude, for example, to $[\gamma, \gamma + 1]$. Hence, with a probability $p_0 > 0$ uniform over $H_s$, its breakpoint lies in the central half of an unbroken bin and its slope jump has magnitude at least $\gamma$. By Lemma 6, this bin becomes broken. The new neuron is affine on every other bin, so their broken or unbroken status is unchanged. This proves eq. (4) with $p = p_0$. We may decrease $p_0$ if necessary so that $p_0 \leq 1/3$.

Regarding deaths, let $q_0$ be the uniform upper bound obtained by considering $T$ broken bins and making the favorable assumption that every new breakpoint landing in one of them repairs that bin. Their total length is at most $T\varepsilon$. Since the breakpoint-location density is bounded above on $I_R$, choosing $C_{\mathrm{cap}}$ sufficiently large gives $q_0 < p_0/2$.

If $B_s^{\mathrm{cap}} < T$, then $B_s^{\mathrm{orig}} = B_s^{\mathrm{cap}} < T$, so there are fewer than $T$ genuinely broken bins and the probability of a decrease is at most $q_0$. If $B_s^{\mathrm{cap}} = T$, the capped count can decrease only when $B_s^{\mathrm{orig}} = T$, in which case the same bound applies. If $B_s^{\mathrm{orig}} > T$, repairing one broken bin still leaves at least $T$ broken bins, so the capped count remains equal to $T$. Thus eq. (5) holds with $q = q_0$. In particular, $p + q < 1$, so the hypotheses of Lemma 5 hold.

## B.2 Proof of the history-conditioned coupling

*Proof of Lemma 5.* We construct the coupling inductively, conditioning at step $s$ on the complete history $H_s$ and on the two current counts. If the counts agree below $T$, the increment of the capped process stochastically dominates the increment of the birth–death chain: its probability of $+1$ is at least $p$, by eq. (4), and its probability of $-1$ is at most $q$, by eq. (5). Hence the two increments can be coupled in increasing order. The same argument applies when both counts equal $T$, except that the birth–death chain has no upward transition there.

If $B_s^{\mathrm{cap}} - B_s^{\mathrm{bd}} \geq 2$, the ordering cannot be reversed in one step because both counts move by at most one. If the gap is one, the only pair of increments that could reverse it is a birth of the birth–death chain together with a death of the capped process. These events can be made disjoint: the former has

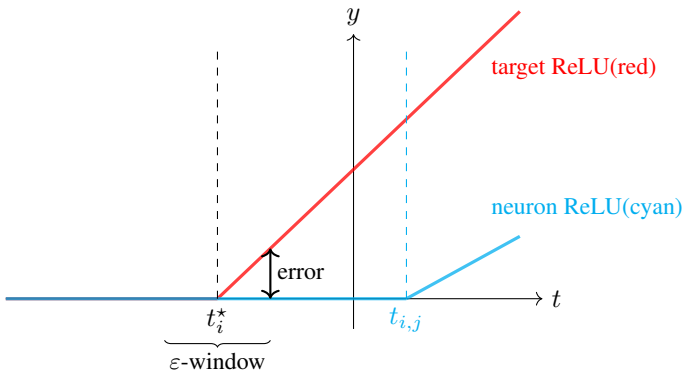


Figure 3: Breakpoint alignment intuition along a one-dimensional input family $\mathbf{x}_i(t)$. Along $\mathbf{x}_i(t)$, a bias-free ReLU neuron $\sigma(\langle w, x\rangle)$ reduces to the one-dimensional function $t \mapsto \sigma(w_{2i-1}t + w_{2i})$, whose breakpoint is $t_i(w) = -w_{2i}/w_{2i-1}$. The target (red) has breakpoint $t_i^\star$, while a randomly drawn neuron (cyan) has breakpoint $t_{i,j}$. If $|t_{i,j} - t_i^\star| > \varepsilon$, then on the $\varepsilon$-neighborhood of $t_i^\star$ the cyan function is linear (in the picture, flat) whereas the target changes slope, yielding a nontrivial uniform error, as stated in Lemma 2.

probability $p$, while the latter has conditional probability at most $q$, and $p + q \le 1$. This completes the inductive coupling. Since both processes start at one, it yields $B_s^{\mathrm{bd}} \le B_s^{\mathrm{cap}}$ for every $s$. The claimed probability inequality follows because $B_k^{\mathrm{cap}} = 0$ implies $B_k^{\mathrm{bd}} = 0$ under this coupling. □

**Lemma 1** (Broken bin prevents approximation)**.** *Let $f(t) := \sigma(at + b)$, with $a \neq 0$, and let $I_B \subset \mathbb{R}$ be an interval of length $\varepsilon$ such that the breakpoint $t^\star := -b/a$ satisfies $t^\star \notin I_B$. Let $h : \mathbb{R} \to \mathbb{R}$ be any function. If $I_B$ is broken for $h$ in the sense of Definition 2, then* $\sup_{t \in I_B} |f(t) - h(t)| \ge c_{\mathrm{br}}\, \varepsilon$.

*Proof.* Since $t^\star \notin I_B$, the function $f$ is linear on $I_B$. By Definition 2, for any affine function $\ell$, there exist three points in $I_B$ on which $h$ differs from $\ell$ by at least $c_{\mathrm{br}}\varepsilon$. Taking $\ell = f$ yields the claim. □

**Lemma 2** (A breakpoint is necessary for approximation)**.** *Let $f(t) := \sigma(at + b)$, $a \neq 0$, and let $t^\star := -b/a$ be its breakpoint. Let $h : \mathbb{R} \to \mathbb{R}$ be a function that is linear on an interval $I$ containing $[t^\star - \delta,\, t^\star + \delta]$ for some $\delta > 0$. Then,* $\sup_{t \in [t^\star - \delta,\, t^\star + \delta]} |f(t) - h(t)| \ge |a|\,\delta/4$.

*Proof.* By translation, we may assume $t^\star = 0$, so that $f(t) = \sigma(at)$. Let $h(t) = \alpha t + \beta$ be any linear function. Consider the three points $t = -\delta,\, 0,\, \delta$. At these points, $f(-\delta) = 0$, $f(0) = 0$, $f(\delta) = a\delta$ if $a > 0$, and the roles of $\pm\delta$ are reversed if $a < 0$. In either case, $|f(\delta) - f(-\delta)| = |a|\delta$. Define the errors $\Delta_- := h(-\delta) - f(-\delta)$, $\Delta_0 := h(0) - f(0)$, $\Delta_+ := h(\delta) - f(\delta)$. Since $h$ is linear, $h(0) = \frac{1}{2}\big(h(-\delta) + h(\delta)\big)$, which implies $\Delta_0 = \frac{1}{2}\big(\Delta_- + \Delta_+ + |a|\delta\big)$ (up to relabeling the endpoints when $a < 0$).

If $|\Delta_-| < \frac{|a|\delta}{4}$ and $|\Delta_+| < \frac{|a|\delta}{4}$, then $\Delta_0 > \frac{1}{2}\Big(-\frac{|a|\delta}{4} - \frac{|a|\delta}{4} + |a|\delta\Big) = \frac{|a|\delta}{4}$, so $|\Delta_0| \ge \frac{|a|\delta}{4}$. Therefore, at least one of $|\Delta_-|, |\Delta_0|, |\Delta_+|$ is at least $\frac{|a|\delta}{4}$, and hence

$$\sup_{t \in [-\delta, \delta]} |f(t) - h(t)| \ge \frac{|a|\delta}{4}.$$

Taking $c = |a|/4$ concludes the proof. □

**Lemma 6** (Adding a sufficiently steep neuron breaks an unbroken bin)**.** *There exist universal constants $\gamma, c_{\mathrm{br}} > 0$, with $c_{\mathrm{br}} \le \gamma/32$, such that the following holds. Let $B = [u, u + \varepsilon]$, and let $h : B \to \mathbb{R}$ be a piecewise-linear function for which $B$ is unbroken. Let $r : B \to \mathbb{R}$ be piecewise-linear with exactly one breakpoint $t_0 \in [u + \varepsilon/4, u + 3\varepsilon/4]$ and slope jump $\Delta$. If $|\Delta| \ge \gamma$, then $B$ is broken for $h + r$ in the sense of Definition 2.*

*Proof.* Set $t_1 = t_0 - \varepsilon/4$, $t_2 = t_0$, and $t_3 = t_0 + \varepsilon/4$. Let $m_-$ and $m_+$ be the slopes of $r$ to the left and right of $t_0$, respectively, so that $\Delta = m_+ - m_-$. Fix an affine function $\widetilde{\ell}$ with slope $q$, and set

$$E := \max_{j=1,2,3} |r(t_j) - \widetilde{\ell}(t_j)|.$$

Since $r$ is affine on each side of $t_0$ and $t_2 - t_1 = t_3 - t_2 = \varepsilon/4$, we have

$$|q - m_-| = \left| \frac{(\widetilde{\ell}(t_2) - r(t_2)) - (\widetilde{\ell}(t_1) - r(t_1))}{t_2 - t_1} \right| \le \frac{2E}{\varepsilon/4} = \frac{8E}{\varepsilon},$$
$$|q - m_+| = \left| \frac{(\widetilde{\ell}(t_3) - r(t_3)) - (\widetilde{\ell}(t_2) - r(t_2))}{t_3 - t_2} \right| \le \frac{2E}{\varepsilon/4} = \frac{8E}{\varepsilon}.$$

Therefore,

$$|\Delta| = |m_+ - m_-| \le |m_+ - q| + |q - m_-| \le \frac{16E}{\varepsilon},$$

which proves that, for every affine function $\widetilde{\ell}$,

$$\max_{j=1,2,3} |r(t_j) - \widetilde{\ell}(t_j)| \ge \frac{|\Delta|\varepsilon}{16}. \tag{8}$$

Because $B$ is unbroken for $h$, the negation of Definition 2, applied to this particular triple, gives an affine function $a$ such that

$$\max_{j=1,2,3} |h(t_j) - a(t_j)| < c_{\mathrm{br}}\varepsilon. \tag{9}$$

Now fix any affine function $\ell$. Since $\ell - a$ is affine, the reverse triangle inequality together with eqs. (8) and (9) gives

$$\begin{aligned}\max_j |h(t_j) + r(t_j) - \ell(t_j)| &\ge \max_j |r(t_j) - (\ell - a)(t_j)| - \max_j |h(t_j) - a(t_j)| \\ &\ge \left(\frac{\gamma}{16} - c_{\mathrm{br}}\right)\varepsilon \ge c_{\mathrm{br}}\varepsilon.\end{aligned}$$

Thus the fixed triple witnesses that $B$ is broken for $h + r$. □

**Lemma 7.** *It holds* $\Pr(B_k^{\mathrm{bd}} = 0) \;\le\; e^{-\Omega(\min(k,T))}$.

*Proof.* Let $R_k$ and $L_k$ denote the number of right moves and left moves, respectively, during the $k$ steps of the process. A right move happens when a new broken bin is introduced, and a left move when it is removed. Then $B_k^{\mathrm{bd}} = 1 + R_k - L_k$, which means that in order to have $B_k^{\mathrm{bd}} = 0$ we must have $L_k > R_k$. By the law of total probability, we thus have

$$\begin{aligned}\Pr(B_k^{\mathrm{bd}} = 0) &= \Pr(1 + R_k = L_k) \\ &\le \Pr(L_k > R_k) \\ &= \Pr\left(L_k > R_k \mid R_k < pk/2\right)\Pr(R_k < pk/2) + \\ &\qquad + \Pr\left(L_k > R_k \mid R_k \ge pk/2\right)\Pr(R_k \ge pk/2) \\ &\le \Pr(R_k < pk/2) + \Pr\left(L_k > R_k \mid R_k \ge pk/2\right) \\ &\le \Pr(R_k < pk/2) + \Pr\left(L_k \ge pk/2 \mid R_k \ge pk/2\right) \\ &\le \Pr(R_k < pk/2) + \Pr\left(L_k \ge pk/2\right), \end{aligned} \tag{10}$$

where, in the last inequality, we used that conditioning on performing at least a certain number of right moves can only decrease the probability of performing at least a given number of left moves, since those moves cannot be to the left. More formally, if we define $h(r) := \Pr(L_k \ge pk/2 \mid R_k = r)$ we have

$$\Pr\Big(L_k \ge \tfrac{pk}{2}\Big) = \sum_{r=0}^{k} h(r)\Pr(R_k = r) = \mathbb{E}[h(R_k)],$$

and, similarly,

$$\Pr\Big(L_k \ge \tfrac{pk}{2} \mid R_k \ge \tfrac{pk}{2}\Big) = \sum_{r=\frac{pk}{2}}^{k} h(r)\Pr\Big(R_k = r \mid R_k \ge \tfrac{pk}{2}\Big) = \mathbb{E}\Big[h(R_k) \mid R_k \ge \tfrac{pk}{2}\Big].$$

Now, since $h(r)$ is a decreasing function of $r$, it follows that

$$\mathbb{E}\Big[h(R_k) \mid R_k \geq \tfrac{pk}{2}\Big] \leq \mathbb{E}[h(R_k)] \iff \Pr\Big(L_k \geq \tfrac{pk}{2} \mid R_k \geq \tfrac{pk}{2}\Big) \leq \Pr\Big(L_k \geq \tfrac{pk}{2}\Big),$$

justifying the last step in eq. (10).

Intuitively, eq. (10) states that the probability that the process ends in a state $B_k^{\mathrm{bd}} = 0$ can be bounded by the event that the number of right moves is too small, or that the number of left moves is at least the number of right ones.

**Case 1:** $k \leq 2T/3$**.** Since $k \leq 2T/3$ the cap at $T$ of the birth-death process is never reached. Since each step produces a right move independently with probability $p$, then $\mathbb{E}[R_k] = pk$, and by a Chernoff bound $\Pr(R_k < pk/2) \leq e^{-\Omega(k)}$. On the other hand, each step produces a left move with probability at most $q \leq p/2$ is a constant. Hence $\mathbb{E}[L_k] \leq q_0 k$. Applying a Chernoff bound to $L_k$, we obtain $\Pr\big(L_k \geq \frac{p}{2}k\big) \leq e^{-\Omega(k)}$, hence

$$\Pr(R_k < pk/2) + \Pr\left(L_k \geq pk/2\right) \leq\ e^{-\Omega(k)}. \tag{11}$$

**Case 2:** $k > 2T/3$**.** At step $\hat{k} = k - 2T/3$, there are two possibilities: either $B_{\hat{k}}^{\mathrm{bd}} \leq T/3$ or $B_{\hat{k}}^{\mathrm{bd}} > T/3$. If $B_{\hat{k}}^{\mathrm{bd}} \leq T/3$, then we can use the same argument as in Case 1 to bound the probability of success, as the cap at $T$ of the birth-death process will not be reached after $k$ steps. If $B_{\hat{k}}^{\mathrm{bd}} > T/3$, then the probability of reaching state 0 in the remaining $2T/3$ is bounded by the probability that the process does at least $T/3$ left moves, which is at most $e^{-\Omega(T)}$. Formally, we have

$$\begin{aligned}
\Pr(B_k^{\mathrm{bd}} = 0) &\leq \Pr(L_k > R_k) \\
&= \Pr\Big(L_k > R_k \mid B_{\hat{k}}^{\mathrm{bd}} \leq T/3\Big) \Pr(B_{\hat{k}}^{\mathrm{bd}} \leq T/3) + \\
&\quad + \Pr\Big(L_k > R_k \mid B_{\hat{k}}^{\mathrm{bd}} > T/3\Big) \Pr(B_{\hat{k}}^{\mathrm{bd}} > T/3) \\
&\leq \Pr\Big(L_k > R_k \mid B_{\hat{k}}^{\mathrm{bd}} \leq T/3\Big) + \Pr\Big(L_k > R_k \mid B_{\hat{k}}^{\mathrm{bd}} > T/3\Big) \\
&\leq \Pr\Big(L_k - L_{\hat{k}} \geq R_k - R_{\hat{k}} \mid B_{\hat{k}}^{\mathrm{bd}} \leq T/3\Big) + \Pr\Big(L_k - L_{\hat{k}} > T/3 \mid B_{\hat{k}}^{\mathrm{bd}} > T/3\Big).
\end{aligned}$$

The first term in the last line are bounded by $e^{-\Omega(T)}$ as shown in Case 1 using similar arguments: note that starting at position larger than 1, only makes it less likely to reach 0. The second term is bounded by a standard Chernoff bound.

Putting both cases together, we get

$$\Pr\left(B_k^{\mathrm{bd}} = 0\right) \leq e^{-\Omega(\min(k,T))}.$$

□

**Lemma 9.** *It holds* $\sum_{k=1}^{T-1} \binom{N_h}{k} p_k \leq e^{-\Omega(d)}$ *and* $\sum_{k=T}^{N_h} \binom{N_h}{k} p_k \leq e^{-\Omega(d)}$.

*Proof.* For $k \leq T$, from Lemma 8 we get that $p_k \ \leq\ e^{-c_4 kd}$ for some constant $c_4 > 0$. Using the bound $\binom{N_h}{k} \leq (eN_h/k)^k$, we obtain

$$\sum_{k=1}^{T-1} \binom{N_h}{k} p_k \ \leq\ \sum_{k=1}^{T-1} e^{-k\left(c_4 d - \log N_h + O(1)\right)} \overset{(a)}{\leq} \sum_{k=1}^{T-1} e^{-\Omega(kd)} \overset{(b)}{\leq} e^{-\Omega(d)}, \tag{12}$$

where in $(a)$ we used that $\log N_h \ \leq\ cd$, and in $(b)$ we upper bounded $\sum_{k=1}^{T-1} e^{-\Omega(kd)} \ \leq\ \sum_{k=1}^{\infty} (e^{-\Omega(d)})^k$ and used that $\sum_{k=1}^{\infty} x^k = \frac{x}{1-x}$.

For $k > T$, given Lemma 8 and the choice of $T$ in eq. (6), we get $p_k \ \leq\ e^{-c_5 d/\varepsilon}$ for some constant $c_5 > 0$. Using the crude bound $\sum_{k>k_0} \binom{N_h}{k} \leq 2^{N_h}$, we obtain

$$\sum_{k=T}^{N_h} \binom{N_h}{k} p_k \ \leq\ 2^{N_h} e^{-c_5 d/\varepsilon} \leq e^{-\Omega(d/\varepsilon)} \leq e^{-\Omega(d)}, \tag{13}$$

where in the second to last inequality we used the hypothesis $N_h \leq cd/\varepsilon$. □

## C Extension of Theorem 1

We explain how the proof of Theorem 1 extends to the case where the output weights are not fixed to 1, but have magnitude bounded away from 0. Consider the network $g(\mathbf{x}) = \sum_{j=1}^{N_h} \alpha_j \, \sigma(\langle \mathbf{w}_j, \mathbf{x} \rangle)$, where the hidden weights $\mathbf{w}_j \sim \mathcal{N}(0, I_d)$ are independent, and where the coefficients $\alpha_j$ are either deterministic or random and independent of the hidden weights. Assume that $|\alpha_j| \geq a_-$ for some constant $a_- > 0$ and all $j$.

The proof proceeds exactly as in the unit-weight case after observing that, along any input family $\mathbf{x}_i(t)$, the restriction of the network is $g(\mathbf{x}_i(t)) = \sum_{j \in S} \alpha_j \, \sigma(w_{j,2i-1} t + w_{j,2i})$. The breakpoint locations remain $t_{i,j} = -w_{j,2i}/w_{j,2i-1}$, so they do not depend on the coefficients $\alpha_j$. Writing the slope jump as the right derivative minus the left derivative, we have $\Delta_{i,j} = \alpha_j |w_{j,2i-1}|, \qquad |\Delta_{i,j}| = |\alpha_j| \, |w_{j,2i-1}|$. Since $|\alpha_j| \geq a_-$, the event

$$|w_{j,2i-1}| \geq \gamma / a_-$$

guarantees $|\Delta_{i,j}| \geq \gamma$. The sign of $\alpha_j$ is irrelevant for the birth estimate: the strengthened Lemma 6 depends only on the magnitude of the new slope jump and applies to any previous unbroken configuration. Consequently, a sufficiently steep, well-placed breakpoint creates a birth with a history-uniform probability $p_0(a_-) > 0$.

The sign is also irrelevant for the death upper bound. The capped count can decrease only if the new breakpoint lies in a currently broken bin, and the comparison already makes the favorable assumption that every such hit repairs that bin. Different signs can change whether cancellation actually occurs, but they cannot increase its probability beyond this location-based bound. Thus the conditional transition inequalities eqs. (4) and (5) continue to hold, with only the birth constant depending on $a_-$.

If the coefficients are random, we condition on their complete realization. The input-family processes are then independent because they involve disjoint coordinates of the hidden weights, and all the preceding bounds are uniform over coefficient realizations satisfying $|\alpha_j| \geq a_-$. Integrating over the coefficients therefore preserves the product estimate across input families and the final counting argument. The same width lower bound follows, with constants depending on $a_-$.

## D Proof of Theorem 2

**Theorem 2.** *Let $d \geq 2$, $R > 2$, $\varepsilon \in (0,1)$, $\delta \in (0,1)$, and let $\mathbf{w}^\star \in \mathbb{R}^d$ with $\|\mathbf{w}^\star\|_2 = \sqrt{d}$. Consider $g(\mathbf{x}) = \sum_{j=1}^{N_h} \alpha_j \, \sigma(\langle \mathbf{w}_j, \mathbf{x} \rangle)$, where $\mathbf{w}_j \sim \mathcal{N}(0, I_d)$ and $\alpha_j \sim \mathcal{N}(0,1)$ are independent. Then there exist a constant $c_0 = c_0(\delta) > 0$ such that, if*

$$N_h < c_0 \frac{\sqrt{d} \log d}{\varepsilon},$$

*then with probability at least $1 - \delta$ over the initialization, for every $S \subseteq \{1, \ldots, N_h\}$, $\sup_{\|\mathbf{x}\|_2 \leq R} \left| \sum_{j \in S} \alpha_j \, \sigma(\langle \mathbf{w}_j, \mathbf{x} \rangle) - \sigma(\langle \mathbf{w}^\star, \mathbf{x} \rangle) \right| \geq \varepsilon/4$.*

*Proof.* Because the Gaussian distribution $\mathcal{N}(0, I_d)$ and the input domain $\{\mathbf{x} : \|\mathbf{x}\|_2 \leq R\}$ are perfectly symmetric, the problem is invariant to orthogonal transformations. Therefore, we can rotate the coordinate system so that the vector $\mathbf{w}^\star$ aligns with the first standard basis vector. Thus, without loss of generality, we assume $\mathbf{w}^\star = \sqrt{d}\mathbf{e}_1$.

For each $i \in \{2, \ldots, d\}$, define the one-dimensional input family $\mathbf{x}_i(t) := t \, \mathbf{e}_1 + \mathbf{e}_i, t \in \mathbb{R}$. Along this path, the output of the target neuron is $f(\mathbf{x}_i(t)) = \sigma(\langle \sqrt{d}\mathbf{e}_1, \mathbf{x}_i(t) \rangle) = \sigma(\sqrt{d}t)$, so its breakpoint is at $t^\star = 0$. Note that despite the shared notation, these are distinct from the input families considered in the proof of Theorem 1. Let $\varepsilon' = \varepsilon/\sqrt{d}$, and let $I := [-\varepsilon', \varepsilon']$. By Lemma 2, applied to $f(t) = \sigma(\sqrt{d}t)$ and $I$, any function that is linear on an interval containing $I$ cannot approximate $\sigma(\sqrt{d}t)$ on $I$ with error smaller than $\varepsilon/4$.

Now fix a subset $S \subseteq \{1, \ldots, N_h\}$ and consider the pruned network $g_S(\mathbf{x}) = \sum_{j \in S} \alpha_j \, \sigma(\langle \mathbf{w}_j, \mathbf{x} \rangle)$, where each hidden neuron $j$ has an incoming weight vector $\mathbf{w_j} = (w_{j,1}, \ldots, w_{j,d})$. Its restriction to

the $i$-th family $\mathbf{x}_i(t)$ is

$$g_{S,i}(t) := g_S(\mathbf{x}_i(t)) = \sum_{j \in S} \alpha_j \, \sigma(w_{j,1} t + w_{j,i}),$$

whose breakpoints are exactly $t_{i,j} := -w_{j,i}/w_{j,1}$, $j \in S$. If none of these breakpoints falls in $I$, then $g_{S,i}$ is linear on $I$ and cannot $(\varepsilon/4)$-approximate $\sigma(\sqrt{d}t)$ there. Hence, for each input family $i \in \{2, \dots, d\}$, successful approximation implies the event

$$E_i := \Big\{ \exists\, j \in \{1, \dots, N_h\} \text{ such that } t_{i,j} \in I \Big\}.$$

In particular, overall success implies $\bigcap_{i=2}^{d} E_i$. We now bound the probability of these events. Let $Z_j := w_{j,1}$ and write

$$L_1 := \sum_{j=1}^{N_h} |Z_j| \qquad \text{and} \qquad L_2 := \sum_{j=1}^{N_h} Z_j^2.$$

Intuitively, by controlling the sums $L_1 = \sum_j |Z_j|$ and $L_2 = \sum_j Z_j^2$, we rule out the possibility that the shared denominators are collectively large enough to make breakpoint hits likely across all input families.

For fixed $i$ and $j$, conditional on $Z_j = z$, the breakpoint condition $t_{i,j} \in I$ is equivalent to $|w_{j,i}| \le \varepsilon' |z|$. Because $w_{j,i} \sim \mathcal{N}(0,1)$, the probability of the complementary event $\Pr(|w_{j,i}| > x)$ equals 1 at $x = 0$ and decays as roughly $e^{-x^2/2}$ for large $x$. Consequently, there exist universal constants $a, b > 0$ such that

$$\Pr(|w_{j,i}| > x) \ge \exp(-ax - bx^2) \qquad \text{for all } x \ge 0.$$

Thus, the conditional probability that a single unit has no breakpoint in $I$ is bounded below by

$$\Pr(t_{i,j} \notin I \mid Z_j = z) \ge \exp(-a\varepsilon' |z| - b\varepsilon'^2 z^2).$$

Consequently, conditioning on $Z := (Z_1, \dots, Z_{N_h})$ and using independence across neurons, we can bound the probability of $E_i$ by:

$$\Pr(E_i \mid Z) = 1 - \prod_{j=1}^{N_h} \Pr(t_{i,j} \notin I \mid Z_j) \le 1 - \exp\Big(-a\varepsilon' L_1 - b\varepsilon'^2 L_2\Big).$$

Moreover, conditioned on $Z$, the events $E_2, \dots, E_d$ are independent, because they depend on disjoint coordinate families $\{w_{j,i}\}_{j=1}^{N_h}$ for different $i \ge 2$. Therefore,

$$\Pr(\text{success} \mid Z) \le \Big(1 - \exp(-a\varepsilon' L_1 - b\varepsilon'^2 L_2)\Big)^{d-1}.$$

It remains to control $L_1$ and $L_2$. Since $Z_j \sim \mathcal{N}(0,1)$, we have $\mathbb{E}[|Z_j|] = \sqrt{2/\pi} < 1$ and $\mathbb{E}[Z_j^2] = 1$, so $\mathbb{E}[L_1] < N_h$ and $\mathbb{E}[L_2] = N_h$. By Markov's inequality,

$$\Pr\left(L_1 > \frac{4N_h}{\delta}\right) \le \frac{\mathbb{E}[L_1]}{4N_h/\delta} < \frac{\delta}{4}, \qquad \text{and} \qquad \Pr\left(L_2 > \frac{4N_h}{\delta}\right) \le \frac{\mathbb{E}[L_2]}{4N_h/\delta} = \frac{\delta}{4}.$$

By a union bound on these two events, with probability at least $1 - \delta/2$, both $L_1 \le KN_h$ and $L_2 \le KN_h$ hold, where $K = 4/\delta$. On this high-probability event, and using $\varepsilon' \in (0,1)$, we have

$$a\varepsilon' L_1 + b\varepsilon'^2 L_2 \le a\varepsilon' KN_h + b\varepsilon' KN_h = (a+b)KN_h\varepsilon'.$$

Let $c_1 = a + b$. We then have

$$\Pr(\text{success} \mid Z) \le \Big(1 - \exp(-c_1 KN_h \varepsilon')\Big)^{d-1}.$$

Now choose $c_0 > 0$ small enough so that $c_1 K c_0 \le 1/2$. If $N_h < c_0 \frac{\log d}{\varepsilon'}$, then $\exp(-c_1 KN_h\varepsilon') \ge d^{-1/2}$, and hence for $d \ge 2$,

$$\Pr(\text{success} \mid Z) \le (1 - d^{-1/2})^{d-1} \le (e^{-d^{-1/2}})^{d-1} \le e^{-\sqrt{d}/2}.$$

For all sufficiently large $d$, this is at most $\delta/2$; the finitely many remaining dimensions can be absorbed by shrinking $c_0$ once more if needed. Therefore, using the total law of probability

$$\Pr(\text{success}) \le 1 \cdot \Pr(L_1 > KN_h \cup L_2 > KN_h) + \Pr(\text{success} \mid L_1, L_2 \le KN_h) \cdot 1 \le \delta.$$

This proves the theorem. □

## E ReLU Approximation with a Single Neuron

**Lemma 10** (Single-neuron approximation requires exponential width)**.** *Let $d \geq 2$. Given $\mathbf{w}^\star \in \mathbb{R}^d$, with $\|\mathbf{w}^\star\|_2 = 1$, let $f(\mathbf{x}) := \sigma\big(\langle \mathbf{w}^\star, \mathbf{x}\rangle\big)$ be a target ReLU neuron without bias. Let $\mathbf{w}$ be sampled uniformly at random from the unit sphere $\mathbb{S}^{d-1}$, and consider approximations of the form*

$$g(\mathbf{x}) := a\,\sigma\big(\langle \mathbf{w}, \mathbf{x}\rangle\big), \qquad a \in \mathbb{R}.$$

*There exist universal constants $c, C > 0$ such that for any $\varepsilon \in (0,1)$,*

$$\Pr\left(\inf_{a\in\mathbb{R}} \sup_{\|\mathbf{x}\|_2\leq 1} \big|f(\mathbf{x}) - g(\mathbf{x})\big| \leq \varepsilon\right) \;\leq\; (C\varepsilon)^{cd}.$$

*Consequently, if a width-$N_h$ random network is initialized with i.i.d. weights $\mathbf{w_1}, \ldots, \mathbf{w_{N_h}} \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ and only a single neuron may be retained, then, for every fixed sufficiently small $\varepsilon$, achieving uniform $\varepsilon$-approximation of $f$ with probability bounded below by a positive constant requires $N_h \geq e^{\Omega(d)}$.*

*Proof.* Fix $\varepsilon \in (0,1)$. The function of $a$ inside the infimum is continuous and tends to infinity as $|a| \to \infty$ (as can be seen by evaluating at $\mathbf{x} = \mathbf{w}$), so the infimum is attained. Suppose that for some $a \in \mathbb{R}$,

$$\sup_{\|\mathbf{x}\|_2\leq 1} \big|\sigma\big(\langle \mathbf{w}^\star, \mathbf{x}\rangle\big) - a\,\sigma\big(\langle \mathbf{w}, \mathbf{x}\rangle\big)\big| \;\leq\; \varepsilon. \tag{14}$$

Let $\theta \in [0,\pi]$ denote the angle between $\mathbf{w}$ and $\mathbf{w}^\star$. Evaluating (14) at $\mathbf{x} = \mathbf{w}^\star$ yields

$$\big|1 - a\,\sigma(\cos\theta)\big| \leq \varepsilon.$$

Because $\varepsilon < 1$, we must have $a\,\sigma(\cos\theta) \geq 1 - \varepsilon > 0$. In particular, $\cos\theta > 0$, so $\theta \in [0, \pi/2)$. If $\theta = 0$, the desired angular bound is immediate. Otherwise, set

$$\mathbf{x} := \frac{\mathbf{w}^\star - \cos\theta\,\mathbf{w}}{\sin\theta}.$$

Then $\|\mathbf{x}\|_2 = 1$, $\langle \mathbf{w}, \mathbf{x}\rangle = 0$, and $\langle \mathbf{w}^\star, \mathbf{x}\rangle = \sin\theta$. For this $\mathbf{x}$ we have $\sigma\big(\langle \mathbf{w}^\star, \mathbf{x}\rangle\big) = \sin\theta$ and $\sigma\big(\langle \mathbf{w}, \mathbf{x}\rangle\big) = 0$, and therefore

$$|f(\mathbf{x}) - g(\mathbf{x})| = \sin\theta.$$

Combining with (14) yields

$$\sin\theta \leq \varepsilon.$$

Since $\theta \in [0, \pi/2)$ and $\sin\theta \geq 2\theta/\pi$ on this interval, it follows that $\theta \leq (\pi/2)\varepsilon < 2\varepsilon$.

Since $\mathbf{w}$ is uniformly distributed on $\mathbb{S}^{d-1}$, the probability that $\angle(\mathbf{w}, \mathbf{w}^\star) \leq 2\varepsilon$ equals the normalized surface area of a spherical cap of half-angle $2\varepsilon$, where by $\angle(\mathbf{w}, \mathbf{w}^\star)$ we denote the angle between the two vectors $\mathbf{w}$ and $\mathbf{w}^\star$. It is standard that there exist universal constants $c, C > 0$ such that for all $\varepsilon \in (0,1)$,

$$\Pr\big(\angle(\mathbf{w}, \mathbf{w}^\star) \leq 2\varepsilon\big) \;\leq\; (C\varepsilon)^{c(d-1)}.$$

Since $d \geq 2$, decreasing $c$ if necessary gives the claimed exponent $cd$.

If $N_h$ neurons are sampled independently, a union bound shows that the probability that *any* single neuron can $\varepsilon$-approximate $f$ is at most $N_h(C\varepsilon)^{cd}$. Hence, for every fixed $\varepsilon$ such that $C\varepsilon < 1$, making this probability bounded below by a positive constant requires $N_h \geq (C\varepsilon)^{-cd} = e^{\Omega(d)}$. □