ON THE RATE OF CONVERGENCE OF GD IN NON-LINEAR NEURAL NETWORKS: AN ADVERSARIAL ROBUSTNESS PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the convergence dynamics of Gradient Descent (GD) in a minimal binary classification setting, consisting of a two-neuron ReLU network and two training instances. We prove that while GD successfully converges to an optimal robustness margin, effectively maximizing the distance between the decision boundary and the training points, this convergence occurs at a prohibitively slow rate, scaling strictly as $\Theta(1/\ln(t))$. To the best of our knowledge, this establishes the first explicit lower bound on the convergence rate of the robustness margin in a non-linear model. Through empirical simulations, we further demonstrate that this inherent failure mode is present beyond our theoretical setting, exhibiting qualitatively similar slow convergence across a range of settings, including different network initializations, the Adam optimizer, and higher-dimensional inputs with wider networks. Our theoretical guarantees are derived via a rigorous analysis of the GD trajectories across the distinct activation patterns of the model. Specifically, we develop tight control over the system's dynamics to bound the trajectory of the decision boundary, overcoming the primary technical challenge introduced by the non-linear nature of the architecture.

1 INTRODUCTION

The implicit bias of optimization algorithms has become a central pillar of deep learning theory in the effort to better understand the generalization capabilities of neural networks. It is now well-understood that continuing optimization long after achieving perfect classification can significantly improve standard generalization performance. Furthermore, research has increasingly shown that this over-training can also enhance a model's adversarial robustness—its resilience against imperceptible perturbations designed to induce misclassification (Biggio et al., 2013; Szegedy et al., 2013). Gradient descent (GD) remains the fundamental algorithm for training these models; however, its theoretical analysis is often confounded by the non-differentiability of ReLU activations and the complexities of discrete step-size tuning. To bypass these technical hurdles, many researchers employ gradient flow (GF) as a continuous-time surrogate. While GF yields more amenable analyses, it often abstracts away the discrete dynamics of actual training. Thus, determining the precise convergence rates of its discrete-time counterpart, GD, remains a critical and non-trivial challenge.

Unlike its discrete counterpart, the implicit bias of GF is significantly better characterized. In the context of homogeneous networks, GF has been shown to maximize the classification margin in the sense of the Karush-Kuhn-Tucker (KKT) conditions (Lyu and Li, 2019; Ji and Telgarsky, 2020). Apart from improving generalization (Lyu et al., 2021; Sarussi et al., 2021; Frei et al., 2022), because this geometric margin dictates the distance to the decision boundary, such margin-maximizing behavior is widely hypothesized to also foster natural resilience against adversarial perturbations. Building on this intuition, recent work has explicitly guaranteed that gradient flow can inherently drive certain networks toward adversarially robust solutions (Min and Vidal, 2025).

It remains unresolved whether general non-linear ReLU networks trained by GD inherit the same bias, and how quickly they realize it. Even when explicit robustness guarantees are established for GF, and in specific regimes where GD and GF are known to align (Elkabetz and Cohen, 2021), GF fundamentally abstracts away computational constraints. Consequently, it remains unclear whether

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.

practical algorithms like GD can converge to these robust optima within a tractable number of iterations. Moreover, the introduction of adaptive methods such as Adam complicates this landscape, as their distinct implicit biases may render robustness guarantees derived from GF analysis inapplicable. Recent work confirms that optimizer choice can select different norm-based margins or qualitatively different predictors (Zhang et al., 2024; Tsilivis et al., 2026; Vasudeva et al., 2025; Gronich and Vardi, 2026). Despite ample evidence that the convergence rate for linear models is of order $\Theta(1/\ln(t))$ (Soudry et al., 2018; Ji and Telgarsky, 2018; Nacson et al., 2019b;a; Ji and Telgarsky, 2021; Ji et al., 2021; Wang et al., 2024), determining the convergence rate in non-linear models, which are of primary interest from a practical perspective, remains a fundamental open problem.

In this paper, we seek to bridge these gaps by studying the convergence of GD toward an adversarially robust margin. We focus on a depth-2, width-2 ReLU network of the form

$$\Phi(\boldsymbol{\theta}; x) \coloneqq [w_1 x + b_1]_+ - [w_2 x + b_2]_+ , \tag{1}$$

where the weights of the output neuron are fixed, and optimization is performed over the hidden layer parameters $\boldsymbol{\theta} = \{w_1, b_1, w_2, b_2\}$. We analyze the optimization dynamics of GD and GF on a binary classification task with training samples $(x_1, y_1) = (-1, -1)$ and $(x_2, y_2) = (1, 1)$. The objective is the empirical risk under the exponential loss, $\ell(z) \coloneqq \exp(-z)$,[1] a standard choice for studying implicit bias in classification:

$$\mathcal{L}(\boldsymbol{\theta}) \coloneqq \frac{1}{2}\exp\left(\Phi(\boldsymbol{\theta}; -1)\right) + \frac{1}{2}\exp\left(-\Phi(\boldsymbol{\theta}; 1)\right). \tag{2}$$

In this paper, we demonstrate that while GF converges to robust solutions that maximize the margin as $t \to \infty$ in this simplified setting, these results provide a false sense of security for practical optimization. We show that while GD eventually converges to the same robust solutions, the rate of convergence is almost always prohibitively slow: the distance to the optimal margin decays only at a rate of $\Theta(1/\ln(t))$. To the best of our knowledge, this provides the first provable slow convergence guarantee for the robustness margin in a non-linear neural network, and significantly strengthens previous findings which were restricted to linear models.

Furthermore, we complement our theoretical analysis with empirical evaluations showing that the finite-time constants can make convergence prohibitively slow under standard initialization. Additional experiments with Adam and with a wider network on higher-dimensional data exhibit qualitatively similar slow margin evolution. These observations suggest that the bottleneck may persist beyond the two-neuron GD setting, although they are not intended as convergence-rate guarantees for broader architectures or optimizers.

The remainder of this paper is structured as follows: After surveying additional related work, we establish our notation and provide the necessary theoretical background in Section 2. In Section 3, we analyze the convergence of GF toward a robust network within our specified setting. Section 4 details the optimization dynamics of discrete-time GD, providing the technical foundation for Section 5, where we derive our main result: the $\Theta(1/\ln(t))$ convergence rate to the optimal robustness margin. Finally, in Section 6, we illustrate the finite-time behavior predicted by our analysis and examine its qualitative behavior across different initialization schemes, the Adam optimizer, and wider networks with higher-dimensional inputs.

1.1 Related work

Implicit Bias Towards Robust Solutions The seminal works of Lyu and Li (2019) and Ji and Telgarsky (2020) provide a strong characterization of the implicit bias of GF in homogeneous neural networks. Specifically, Lyu and Li (2019) link the implicit bias in non-linear networks to margin maximization. They prove that for ReLU networks, GF converges in direction to a KKT point of the maximum margin problem (see Subsection 2.2 for a formal definition), a geometric property that inherently corresponds to pushing the network's decision boundary farther away from the training instances.

Building on this, Ji and Telgarsky (2020) utilize Kurdyka-Łojasiewicz inequalities to analyze deep homogeneous networks, demonstrating that the network's parameters and internal features directionally align to maximize this margin, thereby guaranteeing that the margin distribution asymptotically

[1] As the logistic loss $z \mapsto \ln(1 + \exp(-z))$ exhibits the same exponential tail decay, our results can be naturally extended to that setting; however, we focus on the exponential loss for clarity and analytical simplicity.

converges. While these topological approaches suggest an implicit regularization toward improved adversarial robustness, they do not yield explicit finite-time convergence guarantees.

Recently, Min and Vidal (2025) formalized this connection between implicit bias and adversarial robustness by proving that for shallow networks trained on clustered data, the use of polynomial ReLU (pReLU) activations allows gradient flow to implicitly find solutions with certified $\mathcal{O}(1)$ adversarial robustness. However, because this guarantee pertains exclusively to the continuous-time dynamics of GF, it inherently disregards computational constraints.

Despite the optimistic guarantees discussed above, the connection between margin maximization in the KKT sense and adversarial robustness is not absolute. For instance, Vardi et al. (2022) show that, in certain depth-2 ReLU settings, every KKT point of the max-margin problem is non-robust, while Frei et al. (2023) show that GF converges to non-robust solutions even though robust networks fitting the data exist. Complementarily, Li et al. (2025) identify a feature-averaging bias of GD in two-layer ReLU networks that produces non-robust classifiers even when multiple discriminative features are available. While these works highlight failure modes in the asymptotic direction of GF, our work exposes an orthogonal computational vulnerability: even in settings where the implicit bias perfectly aligns with the optimal robust decision boundary, the finite-time dynamics of GD prevent convergence in tractable time.

Slow Margin Convergence As noted in the introduction, there is ample work studying the rate of convergence to an optimal margin (Soudry et al., 2018; Ji and Telgarsky, 2018; Lyu and Li, 2019; Nacson et al., 2019b;a; Ji and Telgarsky, 2021; Ji et al., 2021; Wang et al., 2024). While these papers are foundational for the modern understanding of implicit bias, their core quantitative results regarding the convergence rate of the weight vector to the max-margin direction are predominantly derived for linear classifiers. Moreover, these analyses typically bound the growth rate of the weights, which in linear models inherently dictates a slow, logarithmic convergence to the optimal margin. In contrast, our setting is significantly more technically involved. In non-linear models, a logarithmic growth bound on the weights does not trivially preclude fast convergence of the robustness margin, as the non-linearities and bias terms could theoretically allow the decision boundary to rapidly shift to the optimal location. Consequently, existing analyses based solely on parameter growth are inadequate for establishing convergence lower bounds in our setting.

Acceleration Algorithms to Improve Convergence Rate Nacson et al. (2019a), Ji et al. (2021), and Wang et al. (2024) propose alternatives to standard GD designed to accelerate convergence to the optimal margin in the linear setting. Specifically, Nacson et al. (2019a) and Ji et al. (2021) prove that NGD, a variant of GD with a step size scaled by the training loss, converges at a faster rate of $\mathcal{O}(1/t)$. Wang et al. (2024) further improve this rate to $\mathcal{O}(\exp(-t))$ by introducing a novel algorithm, PRGD, that progressively rescales the network parameters to artificially bypass the logarithmic norm growth bottleneck of standard GD. Despite these vastly improved convergence rates, it is not obvious whether such algorithms necessarily enjoy the same implicit bias benefits as GF or unaccelerated GD. Moreover, because these methods scale the step size by a significant quantity, they are highly susceptible to "overshooting" in non-linear loss landscapes. In such scenarios, an excessively large step size can prevent convergence entirely or severely impede optimization by permanently deactivating ReLU neurons.

2 Preliminaries and Notation

2.1 Notation and Terminology

We use bold-faced letters to denote vectors (e.g., $\mathbf{x}$) and capital letters to denote random variables (e.g., X). We let $\mathcal{N}(\mu, \sigma^2)$ denote a normal distribution with mean μ and variance σ^2. For a model $\Phi(\boldsymbol{\theta}; \cdot) : \mathbb{R}^d \to \mathbb{R}$, we let the predicted class of a point $\mathbf{x}$ be given by $\operatorname{sign}(\Phi(\boldsymbol{\theta}; \mathbf{x}))$. The robustness margin $\gamma(\boldsymbol{\theta})$ with respect to a dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ is the minimum distance required to perturb an instance $\mathbf{x}_i$ such that the model's prediction changes:

$$\gamma(\boldsymbol{\theta}) \coloneqq \min_{i \in \{1,\dots,n\}} \inf\{\|\Delta\| : \operatorname{sign}(\Phi(\boldsymbol{\theta}; \mathbf{x}_i + \Delta)) \neq y_i\}.$$

For the target architecture in Equation (1), we let $x^\star$ denote the decision threshold where the model's prediction flips. When this point is unique, the robustness margin simplifies to the distance from the

training instances to $x^\star$, namely $\gamma = \min\{|x^\star - 1|, |x^\star + 1|\}$. The optimal margin, denoted $\gamma^\star$, is attained when $x^\star = 0$, yielding $\gamma^\star = 1$. A network $\Phi(\boldsymbol{\theta}; \mathbf{x})$ is called *homogeneous* if there exists $\alpha > 0$ such that for all $c > 0$, all $\boldsymbol{\theta}$ and all $\mathbf{x}$, it holds that $\Phi(c \cdot \boldsymbol{\theta}; \mathbf{x}) = c^\alpha \Phi(\boldsymbol{\theta}; \mathbf{x})$. We utilize standard asymptotic notation (e.g., $\mathcal{O}, \Omega, \Theta$).

2.2 Preliminaries on KKT Conditions

Definition 2.1 (Directional convergence)**.** *We say that GF or GD* converge in direction *to $\hat{\boldsymbol{\theta}}$ if*

$$\lim_{t \to \infty} \frac{\boldsymbol{\theta}_t}{\|\boldsymbol{\theta}_t\|} = \frac{\hat{\boldsymbol{\theta}}}{\|\hat{\boldsymbol{\theta}}\|},$$

where $\boldsymbol{\theta}_t = \boldsymbol{\theta}(t)$ denotes the flow at time t for GF, and $\boldsymbol{\theta}_t = \boldsymbol{\theta}^{(t)}$ denotes the tth iterate of GD, depending on the optimization algorithm.

The following seminal result characterizes the implicit bias of GF on homogeneous neural networks.

Theorem 2.2 (Lyu and Li (2019); Ji and Telgarsky (2020))**.** *Let $\Phi(\boldsymbol{\theta}; \cdot)$ be a depth-2 ReLU neural network parameterized by $\boldsymbol{\theta}$. Consider minimizing either the exponential or the logistic loss over a binary classification dataset $\{(x_i, y_i)\}_{i=1}^n$ using GF. Assume that there exists a time t_0 such that $\mathcal{L}(\boldsymbol{\theta}(t_0)) < \frac{1}{n}$, i.e., $y_i \Phi(\boldsymbol{\theta}(t_0); x_i) > 0$ for every x_i. Then, GF converges in direction to a first-order stationary point (a KKT point) of the following max-margin problem in parameter space:*

$$\min_{\boldsymbol{\theta}} \ \frac{1}{2}\|\boldsymbol{\theta}\|^2 \quad \textit{s.t.} \quad y_i \, \Phi(\boldsymbol{\theta}; x_i) \geq 1 \ \textit{ for all } i \in [n]. \tag{3}$$

Moreover, $\mathcal{L}(\boldsymbol{\theta}(t)) \to 0$ and $\|\boldsymbol{\theta}(t)\| \to \infty$ as $t \to \infty$.

When its conditions are satisfied, the above theorem implies that the model converges in direction to a parameter vector $\boldsymbol{\theta}$ satisfying the following explicit KKT constraints:

$$\boldsymbol{\theta} = \sum_{i=1}^{n} \lambda_i y_i \nabla_{\boldsymbol{\theta}} \Phi(\boldsymbol{\theta}; \mathbf{x}_i), \tag{4}$$

$$y_i \Phi(\boldsymbol{\theta}; \mathbf{x}_i) \geq 1 \ \text{ for all } i \in [n], \tag{5}$$

$$\lambda_1, \ldots, \lambda_n \geq 0, \tag{6}$$

$$\text{for all } i \in [n], \ \text{ if } \ y_i \Phi(\boldsymbol{\theta}; \mathbf{x}_i) \neq 1 \ \text{ then } \lambda_i = 0. \tag{7}$$

3 Convergence of GF to Robust Solutions

We begin by characterizing the implicit bias of GF on the objective defined in Equation (2). In this section, we make the fixed output-layer weights explicit by writing

$$\Phi(\boldsymbol{\theta}; x) = \sum_{j=1}^{2} a_j \left[w_j x + b_j \right]_+, \qquad a_1 = 1, \quad a_2 = -1.$$

As discussed in the introduction, recent literature has established that for homogeneous networks, the implicit bias of GF drives the parameters toward a solution that maximizes the classification margin in the KKT sense. The architecture in Equation (1) is positively homogeneous of degree one by Lemma B.1. By applying Theorem 2.2, we demonstrate that our specific architecture also adheres to this behavior, converging to a solution with an optimal robustness margin.

Theorem 3.1. *Consider optimizing the objective in Equation (2) using GF. Assume that there exists a time $t_0 \geq 0$ such that $\mathcal{L}(\boldsymbol{\theta}(t_0)) < 0.5$. Then,*

$$\lim_{t \to \infty} \frac{\boldsymbol{\theta}(t)}{\|\boldsymbol{\theta}(t)\|} = \boldsymbol{\theta}^\star := (0.5, 0.5, -0.5, 0.5). \tag{8}$$

In particular, $\boldsymbol{\theta}^\star$ achieves an optimal robustness margin $\gamma(\boldsymbol{\theta}^\star) = \gamma^\star$.

The proof of Theorem 3.1, provided in Appendix C, relies on an exhaustive case-by-case analysis of the parameter space to identify solutions satisfying the KKT conditions in Equations (4–7). Our analysis demonstrates that the solution is unique up to a positive linear rescaling, providing a precise characterization of the directional convergence of GF. Furthermore, the limiting network, illustrated in Figure 3 (deferred to Appendix C), achieves the optimal robustness margin $\gamma^\star = 1$. This convergence behavior suggests that GD might similarly benefit from such an implicit bias, eventually driving the network toward an optimal robust solution. Motivated by this possibility, we analyze the discrete-time dynamics in the following section.

4 The Optimization Dynamics of GD

Having characterized the implicit bias of GF, we now analyze the robustness properties of its discrete-time counterpart, GD. To do so, we must first establish the underlying optimization dynamics. Throughout this section, we assume that $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$. In our setting, this condition aligns our analysis with the assumptions of Theorems 2.2 and 3.1, and implies that all data instances are correctly classified, which is a necessary intermediate state for reaching vanishing training error. Given that the primary motivation for studying implicit bias is to understand the beneficial properties emerging as the loss approaches zero, it is natural to focus our analysis on the regime following perfect classification. By investigating the dynamics in this stage, we can precisely quantify how the robustness guarantees discussed in the introduction are (or are not) efficiently realized by GD.

4.1 Neuron specialization

We begin by analyzing the activation patterns once the training loss in Equation (2) drops below 0.5. This constraint ensures that each neuron classifies its corresponding data instance correctly: specifically, the instance whose label matches the sign of the neuron's contribution to the network output. This significantly restricts the set of possible activation patterns. The subsequent dynamics are governed by the fact that GD updates drive each neuron to "specialize" on its corresponding data instance. As optimization progresses, each neuron eventually becomes inactive for the oppositely labeled instance. These dynamics are visualized in Figure 1.

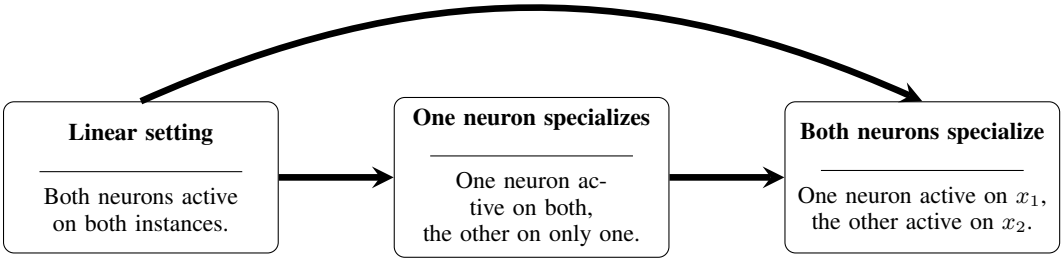


Figure 1: Simplified optimization dynamics after the loss decreases below 0.5. The training trajectory transitions between different states as depicted by the figure. The intermediate state can bifurcate depending on the initial orientations of the neurons. Moreover, in the final state, a degenerate interval may arise in which no neuron is active; this case requires a separate technical analysis to establish that after a finite number of iterations, the degenerate interval vanishes.

4.2 Optimization dynamics at specialization

Once each neuron specializes on a distinct data instance, the optimization dynamics become more analytically tractable. Specifically, we prove that the activation pattern attained at this stage is invariant under GD updates for any sufficiently small step size, and the dynamics are governed by the

following explicit update rules:

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{9}$$

$$b_1^{(t+1)} = b_1^{(t)} + \frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{10}$$

$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right), \tag{11}$$

$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right). \tag{12}$$

These rules yield several key insights. First, the weights and biases are monotone, with their signs asymptotically aligning with the limiting direction of $\boldsymbol{\theta}^\star$, defined in Equation (8). Second, the activation threshold (or breakpoint) of each neuron is pushed toward the instance on which it has become inactive. Third, each neuron's weight and bias move in tandem, whereas the two neurons can update at completely different magnitudes. As we explore in the next section, the imbalanced initialization of these parameters is the fundamental driver of the slow convergence to an optimal robustness margin, as it determines the specific limiting behavior and the resulting discrepancy in the update magnitudes.

4.3 Reaching a GD update equilibrium

Through a careful analysis of the GD dynamics, we prove that the system eventually converges to an equilibrium state. More concretely, we establish the following limiting behavior:

$$\begin{aligned}\lim_{t\to\infty} b_2^{(t)} - b_1^{(t)} &= \lim_{t\to\infty} w_1^{(t)} + w_2^{(t)} \\ &= \frac{w_1^{(0)} - b_1^{(0)} + w_2^{(0)} + b_2^{(0)}}{2}.\end{aligned} \tag{13}$$

This implies the system reaches an equilibrium where $b_2^{(t)} - b_1^{(t)} \approx w_1^{(t)} + w_2^{(t)}$. Specifically, this entails $-w_1^{(t)} - b_1^{(t)} \approx w_2^{(t)} - b_2^{(t)}$, ensuring that the gradient updates in Equations (9–12) become balanced across both neurons.

Beyond synchronizing the neurons' updates, this result allows us to show that GD shares the same implicit bias as GF in this setting, converging in direction to the robust solution $\boldsymbol{\theta}^\star$ by Lemma B.5. Crucially, this equilibrium provides the tight control over the parameters necessary to characterize the model's robustness margin and its rate of convergence.

5 Exponentially Slow Convergence to Robust Solutions

In this section, we leverage the analytical machinery developed in Section 4 to derive the convergence rate of GD toward a robust solution.

5.1 Computational Cost of Margin Maximization with GD

We first demonstrate that once both neurons specialize and their activation thresholds are sufficiently close to the instances for which they are inactive, there exists a unique intersection point between the network output and the x-axis. This point, denoted by $x^\star(t)$ at iteration t, determines the robustness margin of our model and is defined by:

$$x^\star(t) = \frac{b_2^{(t)} - b_1^{(t)}}{w_1^{(t)} - w_2^{(t)}}. \tag{14}$$

According to Equation (13), the numerator in our expression for $x^\star(t)$ converges to an initialization-dependent constant. Given that neuron specialization implies $w_2^{(t)} < 0 < w_1^{(t)}$, the denominator remains strictly positive. By further characterizing the growth rate of the weights, we demonstrate that their magnitudes are $\Theta(\ln(t))$. Consequently, unless $w_1^{(0)} - b_1^{(0)} + w_2^{(0)} + b_2^{(0)} = 0$ (a condition that occurs only on an initialization set of Lebesgue measure zero), the convergence rate of $x^\star(t)$ to the origin is $\Omega(1/\ln(t))$. More formally, we establish the following main result.

Theorem 5.1. *Consider the optimization of the objective in Equation (2) using GD with a fixed step size $\eta \in (0, 0.5]$. Then, for almost all initializations $\boldsymbol{\theta}^{(0)}$ satisfying $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$, we have*

$$\gamma^\star - \gamma\left(\boldsymbol{\theta}^{(t)}\right) = \Theta\left(\frac{1}{\ln(t)}\right).$$

The proof of the above theorem, deferred to Appendix E.1, leverages the machinery developed in the previous section to provide a comprehensive characterization of the optimization dynamics within the small-loss regime. Specifically, we establish the eventual convergence of GD while demonstrating that the asymptotic limit governing the decision boundary remains permanently biased away from the optimal robustness margin.

The above result establishes that GD cannot efficiently optimize the robustness margin even in this minimal homogeneous network. Crucially, this logarithmic rate bottleneck is inherent to the optimization dynamics: the probability of encountering such slow convergence is not merely positive, but equals 1 under any continuous distribution over the initial parameters. Thus, reducing the margin gap below $\varepsilon > 0$ requires a number of iterations exponential in $1/\varepsilon$.

5.2 Non-asymptotic Analysis

While the preceding theorem establishes a strong negative result, its asymptotic nature leaves open a practical loophole: it could be that achieving a margin arbitrarily close to the optimum requires prohibitive computational resources, but reaching a practical near-optimal margin is relatively easy. To rule out this possibility, we provide a non-asymptotic analysis of the model's robustness margin under conditions commonly satisfied by standard initialization schemes (He et al., 2015). Concretely, our analysis reveals that under these assumptions, the convergence toward the equilibrium limit in Equation (13) is strictly monotone. Because He initialization typically initializes biases to zero, the model trivially begins with an optimal margin. However, any initial imbalance in the weights triggers the dynamics discussed in the previous section, actively driving the margin away from the optimum at a linear rate.

Finally, once the system reaches a step-size equilibrium, the margin begins to gradually improve toward the optimum, but it does so at a prohibitively slow rate.

Formally, we establish the following non-asymptotic lower bound on the displacement of the decision boundary from its optimal location, demonstrating the margin's prohibitively slow recovery after the initial degradation phase.

Theorem 5.2. *Consider optimizing the objective in Equation (2) using GD with step size $\eta \in (0, 0.3]$. Suppose that $w_1^{(0)} \in (2.7, 3)$, $w_2^{(0)} \in (-0.5, 0)$, and that $b_1^{(0)} = b_2^{(0)} = 0$. Then, for all $t \geq 7/\eta$, we have*

$$x^\star(t) \geq \frac{1}{25 + 5\ln(1 + 4\eta t)}.$$

The proof of the above theorem, deferred to Appendix E.2, relies on a careful analysis of the decision boundary dynamics under the specified initialization conditions. Notably, the primary technical challenge arises because merely bounding the individual growth rates of the weights and biases is insufficient for establishing tight control over the decision boundary. Instead, we explicitly track the trajectory of the parameters to exploit the zero-initialization of the biases alongside the initial discrepancy between the weights. This allows us to prove that during the early iterations of the algorithm, the second bias term "escapes" the first, permanently establishing a lower bound on their difference that drives the $\Omega(1/\ln(t))$ bottleneck.

As a direct consequence of Theorem 5.2, we establish the following corollary for standard initialization schemes.

Corollary 5.3. *Consider optimizing the objective in Equation (2) using GD with step size $\eta \in (0, 0.3]$, initializing the network parameters via standard He initialization ($w_1^{(0)}, w_2^{(0)} \sim \mathcal{N}(0, 2)$ and $b_1^{(0)} = b_2^{(0)} = 0$). Then, with constant probability over the random initialization, for all iterations $t \geq 7/\eta$, the decision boundary satisfies*

$$x^\star(t) \geq \frac{1}{25 + 5\ln(1 + 4\eta t)}.$$

Crucially, this indicates that under standard gradient descent, our model cannot efficiently return a robust predictor with arbitrarily high probability, as the initialization has a constant probability of trapping the dynamics in this logarithmically slow regime.

6 EXPERIMENTS

Our theory gives a sharp characterization of slow margin convergence in a deliberately minimal setting chosen for analytical tractability, but it does not by itself show that the same behavior persists in larger models. We therefore complement it with experiments that examine the finite-time dynamics under He initialization and probe sensitivity to the optimizer, step size, and a wider, higher-dimensional setting. We present the GD experiment most closely aligned with our theory here and defer the full details of the Adam and wider-network experiments to Appendix F.

6.1 CONVERGENCE OF GD TO UPDATE EQUILIBRIUM

Since our analysis is asymptotic in nature, it does not explicitly capture the transient rate at which the system converges to the equilibrium governed by the updates in Equations (9–12). Moreover, while Equation (13) dictates the limiting behavior of the numerator in the decision boundary expression Equation (14), the speed of this convergence is not obvious. In particular, it is possible that while the denominator grows slowly, it could still be much larger than the numerator during early training, potentially keeping the distance from the optimal margin small for a significant duration.

We now turn to study this rate empirically to demonstrate that the convergence to equilibrium is sufficiently rapid, ensuring that the overall margin convergence is primarily governed by the slow growth rate of the denominator. To this end, we conducted $10,000$ independent trials of GD with fixed step sizes $\eta \in \{0.01, 0.02, 0.05, 0.1\}$ to optimize the objective in Equation (2). We employed standard He initialization (He et al., 2015), setting $b_1^{(0)} = b_2^{(0)} = 0$ and drawing weights from a normal distribution, $w_1^{(0)}, w_2^{(0)} \sim \mathcal{N}(0, 2)$.

Any trial still running at iteration $10{,}000$ was discarded if its loss was not below 0.5 at that checkpoint. For successful trials, we terminated the process once the equilibrium condition was met: $|w_1^{(t)} + w_2^{(t)} + b_1^{(t)} - b_2^{(t)}| \leq 0.001$. This simulation reveals that a necessary condition for correctly classifying both data instances under this initialization is that the weights are initialized with appropriate signs: $w_2^{(0)} < 0 < w_1^{(0)}$. Consequently, only approximately one-quarter of the trials progressed beyond the initial $10{,}000$ iterations, namely, those in which the initialization fell within the correct quadrant.

In all successful trials, GD eventually reached the equilibrium point where $|w_1^{(t)}+w_2^{(t)}+b_1^{(t)}-b_2^{(t)}| \leq 0.001$, confirming that the system always stabilizes. While most runs converged to this threshold within a few hundred thousand iterations, exceptional cases with a substantial initial discrepancy in weight magnitudes converged much more slowly. These are precisely the cases with the largest weight difference, for which convergence to the optimal margin remains significantly slower than in more balanced cases.

Under our initialization assumptions, Equation (13) implies

$$\lim_{t\to\infty} b_2^{(t)} - b_1^{(t)} = \frac{1}{2}w_1^{(0)} + \frac{1}{2}w_2^{(0)}.$$

Conditioning on each weight having the correct sign for specialization at initialization ($w_1^{(0)} > 0$ and $w_2^{(0)} < 0$), we let $W_1 = w_1^{(0)}$ and $W_2 = -w_2^{(0)}$. In this regime, W_1, W_2 are independent half-normally distributed random variables with parameter $\sigma^2 = 2$. The asymptotic numerator in Equation (14) is thus distributed as $B \sim \frac{1}{2}|W_1 - W_2|$. A direct calculation gives the PDF and CDF of B as

$$f_B(x) \coloneqq 2\sqrt{\frac{2}{\pi}} \exp\left(-\frac{x^2}{2}\right) \operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right) \tag{15}$$

and

$$F_B(x) \coloneqq 1 - \operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right)^2,$$

respectively, where $\operatorname{erfc}(x) := \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2}\, dt$ denotes the complementary error function.

A visualization of this PDF against the empirical values from our simulation is provided in Figure 2. According to the derived CDF, we have $\Pr[B > 1] \approx 0.10069$, indicating that more than 10% of all successful GD runs converge to an equilibrium where the numerator is at least 1. In such instances, achieving a margin that is suboptimal by only $\varepsilon = 0.01$ would require a denominator magnitude of approximately 100, which, given the $\Theta(\ln(t))$ growth rate, necessitates a prohibitively large number of iterations.

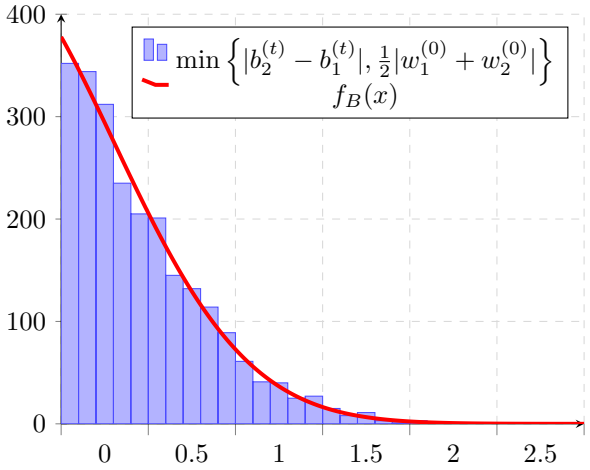


Figure 2: Empirical distribution of the lower-bound guarantees obtained in the experiment with a step size of $\eta = 0.05$. Among 10,000 GD initializations, 2,368 attained loss below 0.5 and satisfied the equilibrium stopping criterion. Since our simulation conditions all satisfy the monotonicity criterion we establish in Lemma E.3, this ensures monotone convergence of the absolute value of the numerator in Equation (14) to $\frac{1}{2}|w_1^{(0)} + w_2^{(0)}|$. This implies the lower bound $\min\left\{|b_2^{(t)} - b_1^{(t)}|,\ \frac{1}{2}|w_1^{(0)} + w_2^{(0)}|\right\}$. The red curve is the theoretical density (scaled to match histogram counts) of $\frac{1}{2}|w_1^{(0)} + w_2^{(0)}|$, given in Equation (15). The close match suggests that this is indeed the distribution of the numerator, not just asymptotically, but one that is also attained in practice.

6.2 Wider networks

To probe beyond our restricted theoretical setting, we train a width-10 ReLU network on 20 points in $\mathbb{R}^5$ for 10^6 updates and then collapse it into a two-neuron comparator on the synthetic points $x = -1$ and $x = 1$, initialized to match the wide network's margin and output at the margin-defining point while satisfying the equilibrium condition analyzed by our theory. We deliberately keep the experiment at a moderate scale so that we can enumerate every activation pattern and compute the Euclidean robustness margin exactly (up to numerical solver tolerance), rather than relying on attack-based estimates, while still testing the effects of increased width, input dimension, and sample size. We restrict the comparison to runs whose loss is below $1/n = 0.05$ at iteration 10^4. This threshold guarantees correct classification of all 20 training points and, for GD, places training in the regime where margin-maximizing implicit bias is expected to govern the dynamics. The threshold is reached by $160/200$ GD runs and all 200 Adam runs. Over the next 10^6 full-batch updates, the margin trajectories of the wider and margin-matched models closely track; the wider network improves slightly faster, while exhibiting similarly slow margin evolution. The same qualitative agreement holds in additional runs with step sizes $\eta \in \{0.05, 0.5\}$. This ablation suggests that these changes do not remove the bottleneck in this setting. Appendix F and Figures 4 and 5 report both the complete trajectory from initialization and a zoomed view of the post-collapse phase.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, literature discovery, informal proof verification, and consistency checks between the manuscript and experimental code. All AI-assisted suggestions, citations, mathematical statements, and experimental descriptions were independently checked by the authors. The authors take full responsibility for the paper's content.

REFERENCES

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*, pages 387–402. Springer, 2013.

Omer Elkabetz and Nadav Cohen. Continuous vs. discrete optimization of deep neural networks. *Advances in Neural Information Processing Systems*, 34:4947–4960, 2021.

Spencer Frei, Gal Vardi, Peter L Bartlett, Nathan Srebro, and Wei Hu. Implicit bias in leaky relu networks trained on high-dimensional data. *arXiv preprint arXiv:2210.07082*, 2022.

Spencer Frei, Gal Vardi, Peter Bartlett, and Nati Srebro. The double-edged sword of implicit bias: Generalization vs. robustness in relu networks. *Advances in neural information processing systems*, 36:8885–8897, 2023.

Eitan Gronich and Gal Vardi. The implicit bias of Adam and Muon on smooth homogeneous neural networks. *arXiv preprint arXiv:2602.16340*, 2026. URL https://arxiv.org/abs/2602.16340.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 1026–1034, 2015.

Ziwei Ji and Matus Telgarsky. Risk and parameter convergence of logistic regression. *arXiv preprint arXiv:1803.07300*, 2018.

Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. *Advances in Neural Information Processing Systems*, 33:17176–17186, 2020.

Ziwei Ji and Matus Telgarsky. Characterizing the implicit bias via a primal-dual analysis. In *Algorithmic Learning Theory*, pages 772–804. PMLR, 2021.

Ziwei Ji, Nathan Srebro, and Matus Telgarsky. Fast margin maximization via dual acceleration. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2021.

Binghui Li, Zhixuan Pan, Kaifeng Lyu, and Jian Li. Feature averaging: An implicit bias of gradient descent leading to non-robustness in neural networks. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.10322.

Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. *arXiv preprint arXiv:1906.05890*, 2019.

Kaifeng Lyu, Zhiyuan Li, Runzhe Wang, and Sanjeev Arora. Gradient descent on two-layer nets: Margin maximization and simplicity bias. *Advances in Neural Information Processing Systems*, 34:12978–12991, 2021.

Hancheng Min and René Vidal. Gradient flow provably learns robust classifiers for orthonormal gmms. In *Forty-second International Conference on Machine Learning*, 2025.

Boris Mityagin. The zero set of a real analytic function. *arXiv preprint arXiv:1512.07276*, 2015.

Mor Shpigel Nacson, Jason Lee, Suriya Gunasekar, Pedro Henrique Pamplona Savarese, Nathan Srebro, and Daniel Soudry. Convergence of gradient descent on separable data. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3420–3428. PMLR, 2019a.

Mor Shpigel Nacson, Nathan Srebro, and Daniel Soudry. Stochastic gradient descent on separable data: Exact convergence with a fixed learning rate. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3051–3059. PMLR, 2019b.

Roei Sarussi, Alon Brutzkus, and Amir Globerson. Towards understanding learning in neural networks with linear teachers. In *International Conference on Machine Learning*, pages 9313–9322. PMLR, 2021.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70): 1–57, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. 2013.

Nikolaos Tsilivis, Eitan Gronich, Julia Kempe, and Gal Vardi. Flavors of margin: Implicit bias of steepest descent in homogeneous neural networks. *Journal of Machine Learning Research*, 27 (104):1–37, 2026. URL https://jmlr.org/papers/v27/25-0634.html.

Gal Vardi, Gilad Yehudai, and Ohad Shamir. Gradient methods provably converge to non-robust networks. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-1521.

Bhavya Vasudeva, Jung Hoon Lee, Vatsal Sharan, and Mahdi Soltanolkotabi. The rich and the simple: On the implicit bias of Adam and SGD. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5401.

Mingze Wang, Zeping Min, and Lei Wu. Achieving margin maximization exponentially fast via progressive norm rescaling. In *International Conference on Machine Learning*, pages 51124–51160. PMLR, 2024.

Chenyang Zhang, Difan Zou, and Yuan Cao. The implicit bias of Adam on separable data. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0756.

A Appendix-specific Notation

We use the shorthand $\boldsymbol{\theta}^{(t)} \coloneqq (w_1^{(t)}, b_1^{(t)}, w_2^{(t)}, b_2^{(t)})$ to denote the network's weights and biases at step t, where $\boldsymbol{\theta}^{(0)}$ are its parameters at initialization. For brevity, we denote $v_1^{(t)} \coloneqq w_1^{(t)} - b_1^{(t)}$, $v_2^{(t)} \coloneqq w_2^{(t)} - b_2^{(t)}$, $u_1^{(t)} \coloneqq w_1^{(t)} + b_1^{(t)}$ and $u_2^{(t)} \coloneqq w_2^{(t)} + b_2^{(t)}$. We use the shorthands $\beta_j^{(t)} \coloneqq -\frac{b_j^{(t)}}{w_j^{(t)}}$ to denote the breakpoints of the jth neuron for $j \in \{1, 2\}$. We use $x^\star(t)$ to denote the intersection point of the architecture defined in Equation (1) with the x-axis (when it is well-defined).

B Auxiliary Lemmas

Lemma B.1. *The architecture in Equation (1) is positively homogeneous of degree one in its parameters.*

Proof. For every $c > 0$, the positive homogeneity of the ReLU function gives

$$\Phi(c\boldsymbol{\theta}; x) = [c(w_1 x + b_1)]_+ - [c(w_2 x + b_2)]_+ = c\Phi(\boldsymbol{\theta}; x).$$

□

The following lemma bounds the updates of GD once a neuron has specialized over a specific instance.

Lemma B.2. *Consider the objective in Equation (2), and suppose that $w_1^{(t)} > 0$ and $\beta_1^{(t)} \in (-1, 1)$, and that the GD update at iteration t is $w_1^{(t+1)} = w_1^{(t)} + x$ and $b_1^{(t+1)} = b_1^{(t)} + x$, for some $x > 0$. Then, $w_1^{(t+1)} > 0$ and $\beta_1^{(t+1)} \in (-1, \beta_1^{(t)})$.*

Likewise, suppose that $w_2^{(t)} < 0$ and $\beta_2^{(t)} \in (-1, 1)$, and that the GD update at iteration t is $w_2^{(t+1)} = w_2^{(t)} - x$ and $b_2^{(t+1)} = b_2^{(t)} + x$, for some $x > 0$. Then, $w_2^{(t+1)} < 0$ and $\beta_2^{(t+1)} \in (\beta_2^{(t)}, 1)$.

Proof. The invariance of the signs of $w_1^{(t)}$ and $w_2^{(t)}$ follows immediately from the monotonicity of their update rules. We now turn to show the monotonicity of the breakpoints towards ± 1. Starting with $\beta_1^{(t)}$, let $w > 0$ and $b \in (-w, w)$, which implies $-\frac{b}{w} \in (-1, 1)$. Define

$$f(x) \coloneqq -\frac{b + x}{w + x}.$$

To compute the derivative, we rewrite

$$f(x) = -\frac{b - w + x + w}{w + x} = \frac{w - b}{w + x} - 1.$$

Differentiating with respect to x yields

$$f'(x) = \frac{b - w}{(w + x)^2}.$$

Since $b \in (-w, w)$ and $w > 0$, the numerator is negative, and therefore $f'(x) < 0$ for all x, namely f is decreasing. This also implies that f is lower bounded by its limit when $x \to \infty$. We compute $\lim_{x\to\infty} f(x) = -1$, and we have that

$$\beta_1^{(t+1)} = -\frac{b_1^{(t+1)}}{w_1^{(t+1)}} = -\frac{b_1^{(t)} + x}{w_1^{(t)} + x} \in \left(-1, \beta_1^{(t)}\right),$$

where the second equality uses the update rules for the weights. Likewise, a similar analysis shows that $\beta_2^{(t+1)} \in \left(\beta_2^{(t)}, 1\right)$, concluding the proof of the lemma. □

The following lemma allows us to exclude many configurations of the architecture in Equation (1), by showing that they imply a large training loss in Equation (2).

Lemma B.3. *Consider the objective in Equation (2), and suppose that either one of the following conditions hold for $\boldsymbol{\theta} = (w_1, b_1, w_2, b_2)$:*

- *Neuron 1 is right-active ($w_1 > 0$) and $\beta_1 \in [1, \infty)$, or neuron 2 is left-active ($w_2 < 0$) and $\beta_2 \in (-\infty, -1]$,*
- *Neuron 1 is left-active and $\beta_1 \in (-\infty, 1]$, or neuron 2 is right-active and $\beta_2 \in [-1, \infty)$,*
- *$w_1 \leq 0$ and $w_2 \geq 0$,*
- *$w_1 = 0$ and $b_1 \leq 0$, or $w_2 = 0$ and $b_2 \leq 0$.*

Then, we have that $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.

Proof.

- First consider all cases where $w_1 \leq 0$ and $w_2 \geq 0$. By these assumptions, we have that $[w_1 x + b_1]_+$ and $-[w_2 x + b_2]_+$ are both non-increasing, which implies that $\Phi(\boldsymbol{\theta}; -1) \geq \Phi(\boldsymbol{\theta}; 1)$. Namely, for $y_1 \Phi(\boldsymbol{\theta}; -1) > 0$ to hold, we must have $\Phi(\boldsymbol{\theta}; -1) < 0$, which implies $\Phi(\boldsymbol{\theta}; 1) < 0$ and $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$ in either case.
- Consider all remaining cases in which either $w_1 = 0$ or $w_2 = 0$. If $b_1 \leq 0$ or $b_2 \leq 0$, then at least one neuron computes the zero function, and our network reduces to a single neuron. Since such a neuron is always non-positive or non-negative (depending on the sign of a_j), at least one instance is misclassified and $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.
- Consider the remaining cases in which $w_1 < 0$ and $w_2 < 0$.
  - Suppose that $\beta_1 \leq 1$. Since both neurons are left-active, we have that $\Phi(\boldsymbol{\theta}; x) = 0$ for all sufficiently large x. Since $\beta_1 \leq 1$ and $a_2 = -1$, we have that $\Phi(\boldsymbol{\theta}; x) \leq 0$ for all $x \leq 1$, which misclassifies x_2, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.
  - If $\beta_1 > 1$ and $\beta_2 \leq -1$, then $\Phi(\boldsymbol{\theta}; x_i) > 0$ for all $i \in \{1, 2\}$, which misclassifies $x_1 = -1$, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.
- Likewise, consider the remaining cases in which $w_1 > 0$ and $w_2 > 0$.
  - Suppose that $\beta_2 \geq -1$. Since both neurons are right-active, we have that $\Phi(\boldsymbol{\theta}; x) = 0$ for all sufficiently small x. Since $\beta_2 \geq -1$ and $a_1 = 1$, we have that $\Phi(\boldsymbol{\theta}; x) \geq 0$ for all $x \geq -1$, which misclassifies x_1, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.
  - If $\beta_2 < -1$ and $\beta_1 \geq 1$, then $\Phi(\boldsymbol{\theta}; x_i) < 0$ for all $i \in \{1, 2\}$, which misclassifies $x_2 = 1$, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.

To conclude the derivation so far, we have excluded all cases in which $w_1 \leq 0$ or $w_2 \geq 0$, except for 6 ($w_1, w_2 < 0$, $\beta_1 > 1$, $\beta_2 > -1$ and $w_1, w_2 > 0$, $\beta_2 < -1$, $\beta_1 < 1$), in which the loss can be smaller than 0.5. In all remaining cases, since the breakpoints β_1, β_2 are well-defined, we will split our analysis according to their values.

Moving on to the cases where $w_1 > 0$ and $w_2 < 0$:

- If $\beta_1 \geq 1$, then $\Phi(\boldsymbol{\theta}; x) \leq 0$ for all $x \leq 1$, hence $x_2 = 1$ is misclassified, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.
- Likewise, if $\beta_2 \leq -1$, then $\Phi(\boldsymbol{\theta}; x) \geq 0$ for all $x \geq -1$, hence $x_1 = -1$ is misclassified, and therefore $\mathcal{L}(\boldsymbol{\theta}) \geq 0.5$.

We have exhausted all possibilities stated in the lemma, and therefore the proof is concluded. □

The following technical lemma allows us to bound the growth rate of recursions that are repeatedly encountered in our analysis.

Lemma B.4. *Suppose that a_t is a sequence such that $a_{t+1} = a_t + C_1 \exp(-C_2 a_t)$ for some $C_1 \neq 0$ and $C_2 > 0$. Then, if $C_1 > 0$, we have*

$$C_2^{-1} \ln\left(\exp(C_2 a_0) + tC_2 C_1\right) \leq a_t \leq C_2^{-1} \ln\left(\exp(C_2 a_0) + tC_2 C_1 \exp\left(C_2 C_1 \exp(-C_2 a_0)\right)\right),$$

and if $C_1 < 0$, we have

$$-C_2^{-1} \ln\left(\exp(C_2 a_0) - tC_2 C_1 \exp\left(-C_2 C_1 \exp(-C_2 a_0)\right)\right) \leq a_t \leq -C_2^{-1} \ln\left(\exp(C_2 a_0) - tC_2 C_1\right).$$

Proof. First, assume that $C_1 > 0$, and denote $\alpha_t \coloneqq \exp(C_2 a_t)$. By the recurrence relation, we have

$$\begin{aligned}\alpha_{t+1} &= \exp(C_2 a_{t+1}) = \exp\left(C_2 a_t + C_2 C_1 \exp(-C_2 a_t)\right) \\ &= \alpha_t \exp\left(C_2 C_1 \exp(-C_2 a_t)\right) = \alpha_t \exp\left(\frac{C_2 C_1}{\alpha_t}\right).\end{aligned}$$

By a standard Taylor expansion argument, we have for all $x \in \mathbb{R}$ that $1 + x \leq \exp(x) \leq 1 + x\exp(x)$. Using these inequalities and the above displayed equation, we arrive at

$$\alpha_t + C_2 C_1 \leq \alpha_{t+1} \leq \alpha_t + C_2 C_1 \exp\left(\frac{C_2 C_1}{\alpha_t}\right).$$

Since a_t is increasing due to $C_1 \exp(-C_2 a_t) > 0$, we have $\alpha_0 \leq \alpha_t$ for all $t \geq 0$, and therefore

$$C_2 C_1 \leq \alpha_{t+1} - \alpha_t \leq C_2 C_1 \exp\left(\frac{C_2 C_1}{\alpha_0}\right).$$

Summing the above for all t, the middle expression telescopes and we obtain

$$tC_2 C_1 \leq \alpha_t - \alpha_0 \leq tC_2 C_1 \exp\left(\frac{C_2 C_1}{\alpha_0}\right).$$

Adding α_0 and taking the logarithm yields

$$C_2^{-1} \ln\left(\alpha_0 + tC_2 C_1\right) \leq a_t \leq C_2^{-1} \ln\left(\alpha_0 + tC_2 C_1 \exp\left(\frac{C_2 C_1}{\alpha_0}\right)\right),$$

and the claim follows by substituting α_0 with its definition.

For the case $C_1 < 0$, consider the sequence $a'_t = -a_t$. Then we have

$$a'_{t+1} = a'_t - C_1 \exp(-C_2 a_t) = a'_t + |C_1| \exp(-C_2 a_t),$$

and the lemma follows by applying the same bounds, substituting a'_t with a_t and $|C_1|$ with $-C_1$, and rearranging. □

Lemma B.5. *Suppose that, from some iteration T onward, the first neuron is active only on $x_2 = 1$ and the second neuron is active only on $x_1 = -1$. Then, for every fixed step size $\eta > 0$, GD converges in direction to $\boldsymbol{\theta}^\star$:*

$$\lim_{t \to \infty} \frac{\boldsymbol{\theta}^{(t)}}{\left\|\boldsymbol{\theta}^{(t)}\right\|} = \boldsymbol{\theta}^\star = \left(\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}, \frac{1}{2}\right).$$

Proof. Reindexing the iterates, assume without loss of generality that $T = 0$. The specialized update rules in Equations (9–12) imply that

$$v_1^{(t)} = w_1^{(t)} - b_1^{(t)} = v_1^{(0)} \qquad \text{and} \qquad u_2^{(t)} = w_2^{(t)} + b_2^{(t)} = u_2^{(0)}$$

for every t. Consequently,

$$w_1^{(t+1)} = w_1^{(t)} + \frac{\eta}{2} \exp\left(v_1^{(0)}\right) \exp\left(-2w_1^{(t)}\right).$$

Likewise, $w_2^{(t)}$ satisfies the recursion

$$-w_2^{(t+1)} = -w_2^{(t)} + \frac{\eta}{2} \exp\left(-u_2^{(0)}\right) \exp(2w_2^{(t)}).$$

Applying Lemma B.4 to these two recursions gives

$$w_1^{(t)} = \frac{1}{2}\ln(t) + \mathcal{O}(1) \qquad \text{and} \qquad -w_2^{(t)} = \frac{1}{2}\ln(t) + \mathcal{O}(1).$$

Using the two invariants above, we also have

$$b_1^{(t)} = w_1^{(t)} - v_1^{(0)} = \frac{1}{2}\ln(t) + \mathcal{O}(1), \qquad b_2^{(t)} = -w_2^{(t)} + u_2^{(0)} = \frac{1}{2}\ln(t) + \mathcal{O}(1).$$

Therefore,

$$\boldsymbol{\theta}^{(t)} = \frac{1}{2}\ln(t)(1, 1, -1, 1) + \mathcal{O}(1), \qquad \left\|\boldsymbol{\theta}^{(t)}\right\| = \ln(t) + \mathcal{O}(1),$$

where the first $\mathcal{O}(1)$ term is understood coordinatewise. Dividing the first relation by the second yields the claimed limit. □

C Proof of Theorem 3.1

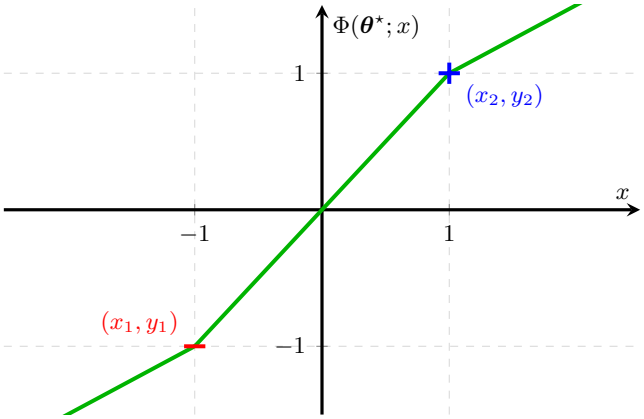


Figure 3: The limiting network $\Phi(\boldsymbol{\theta}^\star; x)$ for $\boldsymbol{\theta}^\star = (0.5, 0.5, -0.5, 0.5)$. The network achieves an optimal robustness margin $\gamma^\star = 1$, perfectly separating the training samples (x_1, y_1) and (x_2, y_2) indicated by the red and blue markers, respectively.

Proof. The proof of the theorem relies on a careful case-by-case analysis, where we will exhaustively go over all possible activation patterns of the neurons and examine them for possible solutions. To do this more cleanly, we will derive new equations that are implied by computing the subgradients and substituting them in Equation (4). To this end, we begin with computing the partial derivatives.

We denote the subgradient of the jth neuron on the ith instance by

$$g_{i,j} \coloneqq \begin{cases} 0, & \text{if } w_j x_i + b_j < 0, \\ [0, 1], & \text{if } w_j x_i + b_j = 0, \\ 1, & \text{if } w_j x_i + b_j > 0. \end{cases}$$

Recall that $a_1 = 1$ and $a_2 = -1$ are the fixed output-layer weights. For $j \in \{1, 2\}$, the partial derivatives are given by

$$\frac{\partial}{\partial w_j}\Phi(\boldsymbol{\theta}; x) = a_j x g_{i,j}, \qquad \frac{\partial}{\partial b_j}\Phi(\boldsymbol{\theta}; x) = a_j g_{i,j}.$$

Combining these with Equation (4), we obtain for all j

$$w_j = a_j \sum_{i=1}^{n} \lambda_i y_i x_i g_{i,j}, \tag{16}$$

$$b_j = a_j \sum_{i=1}^{n} \lambda_i y_i g_{i,j}. \tag{17}$$

Since the above equations are implied whenever a certain point $\boldsymbol{\theta}$ satisfies the KKT conditions, satisfying them is a necessary condition for satisfying KKT. The converse, however, fails to hold as readily seen by examining the trivial all-zero solution. This solution clearly satisfies Equations (16,17), but fails to satisfy Equation (5) since the network computes the zero function under this assignment of weights. In light of this, our strategy will be to exhaustively find all feasible solutions to Equations (16,17), and subsequently verify their correctness in Equations (4–7).

Since we have two data instances, each neuron's breakpoint can reside in five different intervals with distinct activation pattern and subgradient behavior: $(-\infty,-1),-1,(-1,1),1,(1,\infty)$. Moreover, each neuron can be right-active ($w_j > 0$), left-active ($w_j < 0$), or constant with no breakpoint ($w = 0$). This amounts to 11 distinct patterns for each neuron, and 121 cases to analyze in total. However, many cases can be excluded immediately by considering the training loss attained on the sample. To this end, we invoke Lemma B.3, which guarantees that when its conditions hold then the KKT conditions cannot be satisfied. This follows immediately from Equation (5), which implies that the training loss is strictly smaller than 0.5.

In the remainder of the proof, we turn to examine a much smaller number of cases, that are more complex and require a more careful analysis. This part will also include the only feasible solution for our system.

- We begin with ruling out all 6 cases in which either $w_1 = 0$ or $w_2 = 0$. The only such remaining cases are when $b_1 > 0$ and $b_2 > 0$. Let $j \in \{1,2\}$ be the index satisfying $w_j = 0$. Then by Equation (16), we have $0 = a_j(\lambda_1 + \lambda_2)$. Multiplying this by a_j yields $\lambda_1 + \lambda_2 = 0$. Equation (6) then implies that $\lambda_1 = \lambda_2 = 0$, which in turn implies that $\boldsymbol{\theta} = 0$ by Equation (4), but then $\Phi(\boldsymbol{\theta}, \cdot)$ is the zero function, violating Equation (5).

- We are now left with $6 + 9 = 15$ cases. Suppose that $w_1 < 0$, $w_2 < 0$ and $\beta_1 > 1$. We have 3 remaining cases under this assumption, determined by the value of β_2. However, it suffices to consider only the first neuron to show that these cases are infeasible, as we show next.

  Since $w_1 < 0$ and is active on both instances, we get

  $$\begin{aligned} w_1 &= \sum_{i=1}^{n} \lambda_i y_i x_i g_{i,1} = \lambda_1 + \lambda_2, \\ b_1 &= \sum_{i=1}^{n} \lambda_i y_i g_{i,1} = -\lambda_1 + \lambda_2. \end{aligned}$$

  The assumption in this case requires strict activity on both points, and in particular

  $$w_1 x_1 + b_1 > 0.$$

  Substituting the expressions for w_1, b_1 in the above displayed equation yields

  $$w_1 x_1 + b_1 = -(\lambda_1 + \lambda_2) + (-\lambda_1 + \lambda_2) = -2\lambda_1 > 0, \tag{18}$$

  which implies $\lambda_1 < 0$, contradicting Equation (6). Therefore these cases are infeasible.

- Suppose that $w_1 > 0$ and $w_2 > 0$. We have 3 remaining cases under this assumption, determined by the value of β_1. However, it suffices to consider only the second neuron to show that these cases are infeasible, as we show next.

  Since $w_2 > 0$ and is active on both instances we get

  $$\begin{aligned} w_2 &= -\sum_{i=1}^{n} \lambda_i y_i x_i g_{i,2} = -\lambda_1 - \lambda_2, \\ b_2 &= -\sum_{i=1}^{n} \lambda_i y_i g_{i,2} = \lambda_1 - \lambda_2. \end{aligned}$$

  The assumption in this case requires strict activity on both points, and in particular

  $$w_2 x_2 + b_2 > 0.$$

Substituting the expressions for w_2, b_2 in the above displayed equation yields

$$w_2 x_2 + b_2 = -\lambda_1 - \lambda_2 + \lambda_1 - \lambda_2 = -2\lambda_2 > 0, \tag{19}$$

which implies $\lambda_2 < 0$, contradicting Equation (6). Therefore these cases are infeasible.

- Suppose that $w_1 > 0$ and $w_2 < 0$. We have 9 remaining cases under this assumption.
  - Suppose that $\beta_1 < -1$; namely, the first neuron is strictly active on both instances. Despite the opposite orientation of the first neuron, the analysis in this case is identical to the one done in a previous case (see the contradiction derived in Equation (18)), and it is thus skipped.
  - Likewise, the analysis in the case $\beta_2 > 1$ is identical to the one done in a previous case (see the contradiction derived in Equation (19)), and therefore it is also skipped.
  - There are only 4 cases remaining, which include the case with the only feasible solution. We will now rule out two cases.
    Suppose that $\beta_1 \in [-1, 1)$ and $\beta_2 \in (-1, 1)$. Focusing on the second neuron, we have by our assumption that $|-\frac{b_2}{w_2}| < 1$. Since w_2 is left-active, it must hold that $w_2 x_1 + b_2 > 0$ and $w_2 x_2 + b_2 < 0$, implying that $g_{1,2} = 1$ and $g_{2,2} = 0$. Substituting these subgradients in Equations (16,17) yields
    $$w_2 = -\lambda_1 g_{1,2} - \lambda_2 g_{2,2} = -\lambda_1, \qquad b_2 = \lambda_1 g_{1,2} - \lambda_2 g_{2,2} = \lambda_1.$$
    In particular, it must hold that $b_2 = -w_2$ and therefore $\beta_2 = 1 \notin (-1, 1)$, contradicting our assumption in this case, hence this activation configuration is infeasible.
  - Suppose that $\beta_1 \in (-1, 1)$ and $\beta_2 = 1$. Focusing on the first neuron, we have by our assumption that $|-\frac{b_1}{w_1}| < 1$. Since w_1 is right-active, it must hold that $w_1 x_1 + b_1 < 0$ and $w_1 x_2 + b_1 > 0$, implying that $g_{1,1} = 0$ and $g_{2,1} = 1$. Substituting these subgradients in Equations (16,17) yields
    $$w_1 = \lambda_1 g_{1,1} + \lambda_2 g_{2,1} = \lambda_2, \qquad b_1 = -\lambda_1 g_{1,1} + \lambda_2 g_{2,1} = \lambda_2.$$
    In particular, it must hold that $b_1 = w_1$ and therefore $\beta_1 = -1 \notin (-1, 1)$, contradicting our assumption in this case, hence this activation configuration is infeasible.
  - We are now left with the final case, which contains the only feasible solution. Suppose that $\beta_1 = -1$ and $\beta_2 = 1$. In this case, our assumptions imply strong constraints on the weights. In particular, $\beta_1 = -1$ implies $b_1 = w_1$; and $\beta_2 = 1$ implies $b_2 = -w_2$. Moreover, the first neuron is active on x_2 and lies at its kink on x_1, while the second neuron is active on x_1 and lies at its kink on x_2. Hence, $g_{1,2} = g_{2,1} = 1$, whereas $g_{1,1}, g_{2,2} \in [0, 1]$.
    Applying the above to Equations (16,17) yields
    $$w_1 = \lambda_1 g_{1,1} + \lambda_2 g_{2,1} = \lambda_1 g_{1,1} + \lambda_2, \tag{20}$$
    $$w_2 = -\lambda_1 g_{1,2} - \lambda_2 g_{2,2} = -\lambda_1 - \lambda_2 g_{2,2}, \tag{21}$$
    $$b_1 = -\lambda_1 g_{1,1} + \lambda_2 g_{2,1} = -\lambda_1 g_{1,1} + \lambda_2, \tag{22}$$
    $$b_2 = \lambda_1 g_{1,2} - \lambda_2 g_{2,2} = \lambda_1 - \lambda_2 g_{2,2}. \tag{23}$$
    Plugging the above into the four activation constraints yields
    $$\begin{aligned}
    0 &= w_1 x_1 + b_1 = -w_1 + b_1 = -(\lambda_1 g_{1,1} + \lambda_2) + (-\lambda_1 g_{1,1} + \lambda_2) = -2\lambda_1 g_{1,1}, \\
    0 &< w_1 x_2 + b_1 = w_1 + b_1 = (\lambda_1 g_{1,1} + \lambda_2) + (-\lambda_1 g_{1,1} + \lambda_2) = 2\lambda_2, \\
    0 &< w_2 x_1 + b_2 = -w_2 + b_2 = -(-\lambda_1 - \lambda_2 g_{2,2}) + (\lambda_1 - \lambda_2 g_{2,2}) = 2\lambda_1, \\
    0 &= w_2 x_2 + b_2 = w_2 + b_2 = (-\lambda_1 - \lambda_2 g_{2,2}) + (\lambda_1 - \lambda_2 g_{2,2}) = -2\lambda_2 g_{2,2}.
    \end{aligned}$$
    In particular, the middle equations imply $\lambda_1, \lambda_2 > 0$, which together with the first and last equations forces $g_{1,1} = g_{2,2} = 0$. Next, by the middle equations and Equation (7), it must hold that $1 = w_1 x_2 + b_1 = 2\lambda_2$ and $1 = w_2 x_1 + b_2 = 2\lambda_1$, implying $\lambda_1 = \lambda_2 = 0.5$. Finally, plugging the lambdas and the subgradients in Equations (20–23), we obtain the only feasible solution $\boldsymbol{\theta} = (w_1, b_1, w_2, b_2) = (0.5, 0.5, -0.5, 0.5)$. Following our derivation, it is straightforward to plug this solution in Equations (4–7) and verify that all constraints are indeed satisfied.

We thus have that the set of all feasible KKT points is given precisely by

$$\{(\alpha, \alpha, -\alpha, \alpha) : \alpha > 0\}.$$

Since Equations (4–7) assume a normalized margin, the unique normalized solution is given when $\alpha = 0.5$.

We can finally prove the theorem. Suppose that GF reaches a time t_0 such that $\mathcal{L}(\boldsymbol{\theta}(t_0)) < 0.5$. By Theorem 2.2, GF converges in direction to a normalized KKT point. Since this KKT point is unique, we have

$$\lim_{t\to\infty} \frac{\boldsymbol{\theta}(t)}{\|\boldsymbol{\theta}(t)\|} = (0.5, 0.5, -0.5, 0.5).$$

□

D Proofs for Section 4: The Convergence Dynamics of GD

In this section, we analyze the optimization dynamics of GD on the objective in Equation (2). The section is split into different subsections, depending on the activation patterns studied.

D.1 Both Neurons Have Specialized

In this subsection of the appendix, we analyze the gradient descent dynamics after the neurons have specialized. Namely, we consider the regime in which the first neuron is active only on x_2, while the second neuron is active only on x_1.

We further divide this regime into two sub-cases: (i) there exists an interval on which both neurons are simultaneously active, and (ii) there exists an interval on which both neurons are simultaneously inactive. The following two assumptions formalize these sub-cases.

Assumption D.1 (Both neurons are active on an interval)**.** *Given the training set* $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$, *consider the network* $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$. *We assume that at initialization we have* $w_1^{(0)} > 0$, $w_2^{(0)} < 0$ *and* $-1 < \beta_1^{(0)} < \beta_2^{(0)} < 1$.

Assumption D.2 (Both neurons are inactive on an interval)**.** *Given the training set* $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$, *consider the network* $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$. *We assume that at initialization we have* $w_1^{(0)} > 0$, $w_2^{(0)} < 0$ *and* $-1 < \beta_2^{(0)} \leq \beta_1^{(0)} < 1$.

In section D.1.1 we analyze the dynamics of the network under Assumption D.1 and in section D.1.2 we analyze the dynamics of the network under Assumption D.2, but first, we begin with a few technical lemmas that control the parameters' dynamics. The following proposition establishes that once each neuron 'specializes' on its target training instance, the training dynamics become invariant: the activation pattern and the signs of the weights of the neurons remain fixed.

Proposition D.3. *Under Assumption D.1 or Assumption D.2, consider the optimization dynamics of GD on the exponential loss. Then, for any step size* $\eta > 0$*, we have that*

1. $\left\{w_1^{(t)}\right\}_{t=0}^{\infty}$ *is strictly monotonically increasing,*
2. $\left\{b_1^{(t)}\right\}_{t=0}^{\infty}$ *is strictly monotonically increasing,*
3. $\left\{w_2^{(t)}\right\}_{t=0}^{\infty}$ *is strictly monotonically decreasing,*
4. $\left\{b_2^{(t)}\right\}_{t=0}^{\infty}$ *is strictly monotonically increasing.*

In particular, the signs of the weights of both neurons remain constant throughout optimization. Moreover, for all t*, we have*

1. $\beta_1^{(t+1)} \in \left(-1, \beta_1^{(t)}\right)$,
2. $\beta_2^{(t+1)} \in \left(\beta_2^{(t)}, 1\right)$.

Proof. We begin with evaluating $\Phi(\boldsymbol{\theta}^{(t)}; x)$ at x_1 and at x_2.

$$\Phi(\boldsymbol{\theta}^{(t)}; x_1) = \left[-w_1^{(t)} + b_1^{(t)}\right]_+ - \left[-w_2^{(t)} + b_2^{(t)}\right]_+ = w_2^{(t)} - b_2^{(t)},$$
$$\Phi(\boldsymbol{\theta}^{(t)}, x_2) = \left[w_1^{(t)} + b_1^{(t)}\right]_+ - \left[w_2^{(t)} + b_2^{(t)}\right]_+ = w_1^{(t)} + b_1^{(t)}.$$

The loss at iteration t is thus given by

$$\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) = \frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right).$$

With the above, we compute the partial derivatives as follows

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_1^{(t)}} = -\frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_1^{(t)}} = -\frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_2^{(t)}} = \frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_2^{(t)}} = -\frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right),$$

which imply the following gradient descent updates

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{24}$$
$$b_1^{(t+1)} = b_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{25}$$
$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta\exp\left(w_2^{(t)} - b_2^{(t)}\right), \tag{26}$$
$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta\exp\left(w_2^{(t)} - b_2^{(t)}\right). \tag{27}$$

Using the above update rules together with Lemma B.2, we obtain the monotonicity of each parameter in the desired direction, as well as the monotonicity of the breakpoints, concluding the proof of the proposition. □

The following two lemmas bound the growth rate of $w_1^{(t)}$ and $w_2^{(t)}$ when both neurons have specialized.

Lemma D.4. *Under Assumption D.1 or Assumption D.2, for any step size $\eta > 0$, define the quantity $C_1 = \exp\left(\eta\exp\left(v_1^{(0)}\right)\exp\left(-2w_1^{(0)}\right)\right)$ that depends on the initialization parameters and η. Suppose that the network is optimized using GD with a fixed step size η and the exponential loss. Then, we have that $\lim_{t\to\infty} w_1^{(t)} = \infty$. Moreover,*

$$\frac{1}{2}\ln\left(\exp\left(2w_1^{(0)}\right) + \eta\exp\left(v_1^{(0)}\right)\cdot t\right) \le w_1^{(t)} \le \frac{1}{2}\ln\left(\exp\left(2w_1^{(0)}\right) + \eta C_1\exp\left(v_1^{(0)}\right)\cdot t\right)$$

for all t.

Proof. Denote $E_1^{(t)} \coloneqq w_1^{(t+1)} - w_1^{(t)} \overset{\text{Equation 24}}{=} \eta\frac{1}{2}\exp(-w_1^{(t)} - b_1^{(t)})$. We observe that by Equations (24,25),

$$w_1^{(t)} - b_1^{(t)} = w_1^{(t-1)} + E_1^{(t-1)} - b_1^{(t-1)} - E_1^{(t-1)} = w_1^{(t-1)} - b_1^{(t-1)},$$

hence $w_1^{(t)} - b_1^{(t)}$ is constant and it equals $w_1^{(0)} - b_1^{(0)} = v_1^{(0)}$. We rewrite $E_1^{(t)}$ as

$$E_1^{(t)} = \frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right) = \frac{1}{2}\eta \exp\left(-2w_1^{(t)} + v_1^{(0)}\right) = \frac{1}{2}\eta \exp\left(v_1^{(0)}\right) \exp\left(-2w_1^{(t)}\right),$$

yielding

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta \exp\left(v_1^{(0)}\right) \exp\left(-2w_1^{(t)}\right).$$

Using Lemma B.4 with $C_1 = \frac{1}{2}\eta \exp(v_1^{(0)})$ and $C_2 = 2$ concludes the proof. □

Lemma D.5. *Under Assumption D.1 or Assumption D.2, for any step size $\eta > 0$, define the quantity $C_2 = \exp\left(\eta \exp\left(-u_2^{(0)}\right) \exp\left(2w_2^{(0)}\right)\right)$ that depends on the initialization parameters and η. Suppose that the network is optimized using GD with a fixed step size η and the exponential loss. Then, we have that $\lim_{t\to\infty} w_2^{(t)} = -\infty$. Moreover,*

$$-\frac{1}{2}\ln\left(\exp\left(-2w_2^{(0)}\right) + \eta \exp\left(-u_2^{(0)}\right) C_2 \cdot t\right) \le w_2^{(t)} \le -\frac{1}{2}\ln\left(\exp\left(-2w_2^{(0)}\right) + \eta \exp\left(-u_2^{(0)}\right) \cdot t\right),$$

for all t.

Proof. We denote $E_2^{(t)} := -w_2^{(t+1)} + w_2^{(t)} = \eta\frac{1}{2}\exp(w_2^{(t)} - b_2^{(t)})$. We observe that by Equations (26,27),

$$-w_2^{(t)} - b_2^{(t)} = -w_2^{(t-1)} + E_2^{(t-1)} - b_2^{(t-1)} - E_2^{(t-1)} = -w_2^{(t-1)} - b_2^{(t-1)},$$

hence $-w_2^{(t)} - b_2^{(t)}$ is a constant and it equals $-w_2^{(0)} - b_2^{(0)} = -u_2^{(0)}$. We rewrite $E_2^{(t)}$ as

$$E_2^{(t)} = \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right) = \frac{1}{2}\eta \exp\left(2w_2^{(t)} - u_2^{(0)}\right) = \frac{1}{2}\eta \exp\left(-u_2^{(0)}\right) \exp\left(2w_2^{(t)}\right),$$

yielding

$$-w_2^{(t+1)} = -w_2^{(t)} + \frac{1}{2}\eta \exp\left(-u_2^{(0)}\right) \exp\left(2w_2^{(t)}\right).$$

Using Lemma B.4 with $C_1 = \frac{1}{2}\eta \exp(-u_2^{(0)})$ and $C_2 = 2$ concludes the proof. □

D.1.1 Both neurons are active on an interval

We analyze the dynamics of the network under Assumption D.1.

The margin is defined as the minimal distance between a training instance and the decision boundary. In this particular subsection of the appendix, the decision boundary is the intersection point of the network with the x-axis, so by analyzing the dynamics of the intersection point, we are able to analyze the dynamics of the margin. Recall our shorthand for the intersection point after t GD steps, $x^\star(t)$. Under the assumption that both neurons are active on a non-empty set ($\beta_1^{(0)} < \beta_2^{(0)}$), the intersection point is distinct, and it satisfies the equality

$$w_1^{(t)} x^\star(t) + b_1^{(t)} = w_2^{(t)} x^\star(t) + b_2^{(t)}.$$

Solving for $x^\star(t)$, we obtain

$$x^\star(t) = -\frac{b_1^{(t)} - b_2^{(t)}}{w_1^{(t)} - w_2^{(t)}}. \tag{28}$$

We note that under Assumption D.1 $x^\star(t)$ is well-defined for all $t \ge 0$.

Our key proposition in this subsection of the appendix is the following, which establishes tight asymptotic bounds on the rate of convergence to an optimal margin.

Proposition D.6. *Under Assumption D.1, consider the optimization dynamics of GD on the exponential loss. For any step size $\eta > 0$, suppose that $v_1^{(t)} + u_2^{(t)} \neq 0$. Then,*

$$x^\star(t) = \Theta\left(\frac{1}{\ln(t)}\right).$$

The proof of the proposition relies on a few auxiliary lemmas. In what follows, we first establish these required claims, and then turn to prove the proposition.

The following lemma establishes that as t tends to infinity, the system reaches an equilibrium in which the difference of the biases equals the sum of the weights of the neurons.

Lemma D.7. *Under Assumption D.1, consider the optimization dynamics of GD on the exponential loss with any step size $\eta > 0$. Then,*

$$\lim_{t\to\infty} b_2^{(t)} - b_1^{(t)} = \lim_{t\to\infty} w_1^{(t)} + w_2^{(t)} = \frac{v_1^{(0)} + u_2^{(0)}}{2}.$$

Proof. First, observe that by the GD updates in Equations (24–27), we have that for all $t \geq 0$,

$$w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)} = w_1^{(0)} - b_1^{(0)} + w_2^{(0)} + b_2^{(0)} = v_1^{(0)} + u_2^{(0)}, \tag{29}$$

Thus proving any of the limits in the statement of the lemma immediately implies the other. It will therefore suffice to prove the lemma for $w_1^{(t)} + w_2^{(t)}$.

For the sake of brevity, we denote $S^\star \coloneqq \frac{v_1^{(0)}+u_2^{(0)}}{2}$ and $S_t \coloneqq w_1^{(t)} + w_2^{(t)}$. Compute

$$\begin{aligned}
S_{t+1} &= w_1^{(t+1)} + w_2^{(t+1)} \\
&= w_1^{(t)} + \frac{1}{2}\eta\exp(-2w_1^{(t)} + v_1^{(0)}) + w_2^{(t)} - \frac{1}{2}\eta\exp(2w_2^{(t)} - u_2^{(0)}) \\
&= S_t + \frac{1}{2}\eta\left[\exp(-2w_1^{(t)} + v_1^{(0)}) - \exp(2w_2^{(t)} - u_2^{(0)})\right].
\end{aligned}$$

Denote $D_t \coloneqq w_1^{(t)} - w_2^{(t)} > 0$. We can rewrite $w_1^{(t)} = \frac{1}{2}(S_t + D_t)$ and $w_2^{(t)} = \frac{1}{2}(S_t - D_t)$, and therefore we can also rewrite $S_{t+1} - S_t$ using $v_1^{(0)}$, $u_2^{(0)}$, S_t and D_t as follows

$$\begin{aligned}
S_{t+1} - S_t &= \frac{1}{2}\eta\left[\exp(-2w_1^{(t)} + v_1^{(0)}) - \exp(2w_2^{(t)} - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\left[\exp(-S_t - D_t + v_1^{(0)}) - \exp(S_t - D_t - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\exp(-D_t)\left[\exp(v_1^{(0)} - S_t) - \exp(S_t - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\exp(-D_t)g(S_t),
\end{aligned} \tag{30}$$

where $g(s) \coloneqq \exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)})$. We observe that by a simple substitution, we have $g(S^\star) = 0$, and that the derivative of g satisfies $g'(s) = -\exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)}) < 0$. Since $g(s)$ is strictly decreasing and continuous, its root $S^\star$ is unique. From $g(S^\star) = 0$ and the monotonicity of g, it follows that $g(S) > 0$ for all $S < S^\star$, and $g(S) < 0$ for all $S > S^\star$, and thus the following inequality holds for all $S \neq S^\star$:

$$(S - S^\star)g(S) < 0. \tag{31}$$

Define the shorthand $\alpha_t \coloneqq \frac{\eta}{2}\exp(-D_t)$. Using Lemmas D.4 and D.5, and their defined quantities C_1 and C_2, we can bound the growth rate of D_t, to obtain

$$\frac{1}{\sqrt{(K_2 + \tilde{G}_2 t)(K_1 + \tilde{G}_1 t)}} \leq \exp(-D_t) \leq \frac{1}{\sqrt{(K_2 + G_2 t)(K_1 + G_1 t)}},$$

where $K_1 = \exp(2w_1^{(0)})$, $K_2 = \exp(-2w_2^{(0)})$, $G_1 = \eta\exp(v_1^{(0)})$, $G_2 = \eta\exp(-u_2^{(0)})$, $\tilde{G}_1 = \eta C_1\exp(v_1^{(0)})$ and $\tilde{G}_2 = \eta C_2\exp(-u_2^{(0)})$. For all $t \geq 1$, the above implies

$$\frac{1}{t\sqrt{(K_2 + \tilde{G}_2)(K_1 + \tilde{G}_1)}} \leq \exp(-D_t) \leq \frac{1}{t\sqrt{G_2 G_1}},$$

and therefore $\exp(-D_t) = \Theta\left(\frac{1}{t}\right)$, which subsequently implies that

$$\sum_{t=0}^{\infty} \alpha_t = \infty, \tag{32}$$

$$\sum_{t=0}^{\infty} \alpha_t^2 < \infty. \tag{33}$$

We proceed to show that S_t is bounded. By Lemmas D.4 and D.5 again, there exist $c_1, c_2, c_3, c_4 > 0$ and $c_1', c_2', c_3', c_4' > 0$ such that

$$0.5\ln(c_1 + c_2 t) - 0.5\ln(c_3 + c_4 t) \leq S_t = w_1^{(t)} + w_2^{(t)} \leq 0.5\ln(c_1' + c_2' t) - 0.5\ln(c_3' + c_4' t)$$

for all $t \geq 0$. Define

$$f(t) \coloneqq \ln(c_1 + c_2 t) - \ln(c_3 + c_4 t) = \ln\left(\frac{c_1 + c_2 t}{c_3 + c_4 t}\right).$$

It suffices to show that the natural logarithm of any such rational function is bounded, since this would imply both a lower and an upper bound on S_t for all t. To this end, compute

$$f'(t) = \frac{c_2(c_3 + c_4 t) - c_4(c_1 + c_2 t)}{(c_3 + c_4 t)^2} = \frac{c_2 c_3 - c_4 c_1}{(c_3 + c_4 t)^2}.$$

Since the numerator is constant and the denominator is positive, f is monotone, and its value is therefore bounded between $f(0) = \ln(c_1/c_3)$ and $\lim_{t\to\infty} f(t)$. Compute

$$\lim_{t\to\infty} f(t) = \ln\left(\frac{c_1 + c_2 t}{c_3 + c_4 t}\right) = \ln\left(\frac{c_2}{c_4}\right).$$

Thus, S_t is bounded.

We move on to show that S_t is convergent. Define $V_t \coloneqq (S_t - S^\star)^2$. Expanding V_{t+1} using Equation (30), we obtain

$$V_{t+1} = (S_t - S^\star + \alpha_t g(S_t))^2 = V_t + 2\alpha_t(S_t - S^\star)g(S_t) + \alpha_t^2 g(S_t)^2.$$

Summing from 0 to $T-1$ yields

$$V_T = V_0 + \sum_{i=0}^{T-1} 2\alpha_i(S_i - S^\star)g(S_i) + \sum_{i=0}^{T-1} \alpha_i^2 g(S_i)^2 = V_0 + \sum_{i=0}^{T-1} A_i + \sum_{i=0}^{T-1} B_i,$$

where $A_i \coloneqq 2\alpha_i(S_i - S^\star)g(S_i) \leq 0$ by Equation (31), and $B_i \coloneqq \alpha_i^2 g(S_i)^2 \geq 0$. Rearranging, we obtain

$$V_T + \sum_{i=0}^{T-1}(-A_i) = V_0 + \sum_{i=0}^{T-1} B_i.$$

Since S_t is bounded and $g(s)$ is continuous, $(g(S_t))^2 \leq M$ is bounded by some $M > 0$ for all $t \geq 0$; and because $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$ by Equation (33), we deduce that $\sum_{i=0}^{\infty} B_i < \infty$. Rearranging the above displayed equation again, we have

$$V_T + \sum_{i=0}^{T-1} -A_i \leq V_0 + \sum_{i=0}^{\infty} B_i < \infty.$$

Since both terms on the left-hand side are non-negative, they must be bounded. In particular, this implies that

$$\sum_{i=0}^{\infty} -A_i < \infty,$$

and by multiplying by -1 that

$$\sum_{i=0}^{\infty} A_i > -\infty.$$

Since

$$V_T = V_0 + \sum_{i=0}^{T-1} A_i + \sum_{i=0}^{T-1} B_i$$

and both $\sum A_i$ and $\sum B_i$ are convergent, then V_t must also converge.

Suppose by contradiction that $\lim V_t > 0$. Then, there exist $\delta > 0$ and $t_0 \geq 0$, such that for all $t \geq t_0$, $|S_t - S^\star| > \delta$. Combining this with Equation (31), we have that for all $t \geq t_0$,

$$(S_t - S^\star) g(S_t) \leq -c < 0$$

for some constant $c > 0$. Therefore, for all $t \geq t_0$ we obtain the inequality

$$V_{t+1} - V_t = A_t + B_t \leq -2c\alpha_t + M\alpha_t^2.$$

Since $\alpha_t \to 0$, there exists $t_1 \geq t_0$ such that $M\alpha_t^2 \leq c\alpha_t$ for all $t \geq t_1$. We now have for all $t \geq t_1$ that

$$V_{t+1} - V_t < -c\alpha_t.$$

Summing the above from t_1 to T, we get

$$V_T < V_{t_1} - c \sum_{i=t_1}^{T} \alpha_i.$$

By Equation (32), $\sum \alpha_t = \infty$ is divergent, thus the right-hand side tends to $-\infty$ as $T \to \infty$, implying that for sufficiently large T_0 we have $V_{T_0} < 0$, contradicting the definition of $V_t = (S_t - S^\star)^2$.

Overall, we obtained that $\lim_{t\to\infty} V_t = 0$, which implies that

$$\lim_{t\to\infty} S_t = S^\star,$$

and the lemma follows. □

With the above lemmas, we are ready to prove the proposition.

Proof of Proposition D.6. By Proposition D.3, we obtain that the training dynamics are invariant under Assumption D.1. In particular, the GD updates remain the same throughout the entire optimization process. Using Lemmas D.4 and D.5, we deduce that

$$w_1^{(t)} - w_2^{(t)} = \Theta\left(\ln(t)\right) \underset{t\to\infty}{\longrightarrow} \infty. \tag{34}$$

Next, by Lemma D.7 we get that

$$\lim_{t\to\infty} b_2^{(t)} - b_1^{(t)} = \frac{v_1^{(0)} + u_2^{(0)}}{2} \neq 0,$$

which is a constant independent of t by our assumption. Overall, we obtain that

$$\lim_{t\to\infty} x^\star(t) \overset{\text{Equation (28)}}{=} \lim_{t\to\infty} -\frac{b_1^{(t)} - b_2^{(t)}}{w_1^{(t)} - w_2^{(t)}} = \lim_{t\to\infty} \frac{1}{2} \frac{v_1^{(0)} + u_2^{(0)}}{w_1^{(t)} - w_2^{(t)}} = 0. \tag{35}$$

We now turn to analyze the convergence rate. Assume without loss of generality that $v_1^{(0)} + u_2^{(0)} > 0$ (the other direction is symmetric). By Lemma D.7, there exists large enough $t_0 \geq 0$ such that for all $t \geq t_0$, we have

$$\frac{v_1^{(0)} + u_2^{(0)}}{4} \leq b_2^{(t)} - b_1^{(t)} \leq \frac{3(v_1^{(0)} + u_2^{(0)})}{4}.$$

Combining the two-sided numerator bound above with $w_1^{(t)} - w_2^{(t)} = \Theta(\ln t)$ from Equation (34), and plugging it in Equation (28), we obtain

$$x^\star(t) = \Theta\left(\frac{1}{\ln(t)}\right).$$

This concludes the proof of the proposition. □

D.1.2 BOTH NEURONS ARE INACTIVE ON AN INTERVAL

In this subsection, we analyze the dynamics of the network under Assumption D.2.

Proposition D.8. *Under Assumption D.2, consider the optimization dynamics of GD on the exponential loss with a fixed step size $\eta > 0$. Then, there exists a constant $T \geq 0$ that depends only on $\boldsymbol{\theta}^{(0)}$ and η, such that after at most T iterations, the network satisfies Assumption D.1.*

Proof. It will suffice to prove that after a constant number of iterations T we have $\beta_1^{(T)} < 0 < \beta_2^{(T)}$, since this will immediately imply that we have switched to the conditions in Assumption D.1.

First, by Proposition D.3, we have that the next iteration either maintains the conditions in Assumption D.2, or switches us to Assumption D.1, so we cannot change to any other setting. We will show that both biases eventually become positive, and therefore due to the signs of their corresponding weights, we have $\beta_1^{(T)} < 0 < \beta_2^{(T)}$.

Denote $E_t \coloneqq \frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right)$. Then by Equations (24,25), we have

$$b_1^{(t+1)} - b_1^{(t)} = E_t = w_1^{(t+1)} - w_1^{(t)}.$$

Summing over the above from 0 to $t-1$ and rearranging yields

$$b_1^{(t)} = w_1^{(t)} - w_1^{(0)} + b_1^{(0)}.$$

By Lemma D.4, we have $w_1^{(t)} = \Omega(\ln(t))$, where the Omega notation hides constants that depend only on the initialization and the fixed step size η. This, along with the above displayed equation, implies that $b_1^{(t)} = \Omega(\ln(t))$. Namely, after a constant number of iterations t_1, we have $b_1^{(t_1)} > 0$, and therefore $\beta_1^{(t_1)} < 0$.

Likewise, by Equations (26,27), we have

$$b_2^{(t+1)} - b_2^{(t)} = \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right) = -w_2^{(t+1)} + w_2^{(t)}.$$

Summing over the above from 0 to $t-1$ and rearranging yields

$$b_2^{(t)} = -w_2^{(t)} + w_2^{(0)} + b_2^{(0)}.$$

By Lemma D.5, we have $-w_2^{(t)} = \Omega(\ln(t))$, where the Omega notation hides constants that depend only on the initialization and the fixed step size η. This, along with the above displayed equation, implies that $b_2^{(t)} = \Omega(\ln(t))$. Namely, after a constant number of iterations t_2, we have $b_2^{(t_2)} > 0$, and therefore $\beta_2^{(t_2)} > 0$.

Taking $T = \max\{t_1, t_2\}$, the proposition follows after at most T iterations. □

D.2 ONLY ONE NEURON HAS SPECIALIZED

In this section, we analyze all the cases where one neuron has specialized on its instance, while the other neuron is active on both. We limit ourselves to cases where $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$, since these are the cases of interest in our analysis, in which a perfect classification error is obtained.

D.2.1 RIGHT POINTING NEURON ACTIVE ON BOTH, LEFT POINTING NEURON SPECIALIZED

Assumption D.9. *Given the training set $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$, consider the network $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$. We assume that at initialization we have $w_1^{(0)} > 0$, $w_2^{(0)} < 0$, $\beta_1^{(0)} < -1$ and $\beta_2^{(0)} \in (-1, 1)$, and that $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$.*

Proposition D.10. *Under Assumption D.9, consider the optimization dynamics of GD on the exponential loss with a fixed step size $\eta > 0$. Then, there exists a constant $T \geq 0$ that depends only on $\boldsymbol{\theta}^{(0)}$ and η, such that after at most T iterations, the network satisfies Assumption D.1 or Assumption D.2.*

Proof. For a network that satisfies Assumption D.9, the outputs of the network on the training points are

$$\Phi(\boldsymbol{\theta}, x_1) = [-w_1 + b_1]_+ - [-w_2 + b_2]_+ = -w_1 + b_1 + w_2 - b_2,$$
$$\Phi(\boldsymbol{\theta}; x_2) = [w_1 + b_1]_+ - [w_2 + b_2]_+ = w_1 + b_1,$$

and the training loss is therefore

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\exp\left(-w_1 + b_1 + w_2 - b_2\right) + \frac{1}{2}\exp\left(-w_1 - b_1\right). \tag{36}$$

With the above, we can compute the partial derivatives to obtain

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_1} = -\frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2) - \frac{1}{2}\exp(-w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_1} = \frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2) - \frac{1}{2}\exp(-w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_2} = \frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_2} = -\frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2).$$

Thus, the GD updates are given by

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{37}$$

$$b_1^{(t+1)} = b_1^{(t)} - \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right), \tag{38}$$

$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right), \tag{39}$$

$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right). \tag{40}$$

By the above update rules and Lemma B.2, we have that the signs of the weights are invariant, as well as the breakpoint $\beta_2^{(t)}$. Since $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$, the first neuron cannot turn inactive on x_2, and we thus deduce that we must remain under the conditions of Assumption D.9, or transition to Assumption D.1 or Assumption D.2. We will now show that we cannot remain in Assumption D.9 for more than a constant number of iterations.

From Equations (37,38), we get

$$v_1^{(t+1)} = w_1^{(t+1)} - b_1^{(t+1)} = w_1^{(t)} - b_1^{(t)} + \eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) = v_1^{(t)} + \eta\exp\left(v_2^{(t)} - v_1^{(t)}\right), \tag{41}$$

and from Equations (39,40) we obtain

$$v_2^{(t+1)} = w_2^{(t+1)} - b_2^{(t+1)} = w_2^{(t)} - b_2^{(t)} - \eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) = v_2^{(t)} - \eta\exp\left(v_2^{(t)} - v_1^{(t)}\right). \tag{42}$$

Using the shorthand $S \coloneqq v_1^{(0)} + v_2^{(0)}$, we observe that

$$v_1^{(t+1)} + v_2^{(t+1)} = v_1^{(t)} + \eta\exp\left(v_2^{(t)} - v_1^{(t)}\right) + v_2^{(t)} - \eta\exp\left(v_2^{(t)} - v_1^{(t)}\right) = v_1^{(t)} + v_2^{(t)}.$$

By induction, we get that $v_1^{(t)} + v_2^{(t)} = S$ for all t. Substituting $v_2^{(t)} = S - v_1^{(t)}$ into the update rule for $v_1^{(t)}$ in Equation (41), we have

$$v_1^{(t+1)} = v_1^{(t)} + \eta\exp\left(S - 2v_1^{(t)}\right) = v_1^{(t)} + \eta\exp(S)\exp\left(-2v_1^{(t)}\right).$$

Applying Lemma B.4 to the above recursion, we have that $v_1^{(t)}$ increases at a logarithmic rate. In particular, there exists an initialization-dependent constant T such that $v_1^{(T)} > 0$, implying $w_1^{(T)} > b_1^{(T)}$. Since $w_1^{(T)} > 0$, we divide by $w_1^{(T)}$ and multiply by -1 to obtain

$$\beta_1^{(T)} > -1,$$

implying that the first neuron is no longer active on x_1. □

D.2.2 LEFT POINTING NEURON ACTIVE ON BOTH, RIGHT POINTING NEURON SPECIALIZED

Assumption D.11. *Given the training set* $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$, *consider the network* $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$. *We assume that at initialization we have* $w_1^{(0)} > 0$, $w_2^{(0)} < 0$, $\beta_1^{(0)} \in (-1, 1)$ *and* $\beta_2^{(0)} > 1$, *and that* $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ *for all* $t \geq 0$.

Proposition D.12. *Under Assumption D.11, consider the optimization dynamics of GD on the exponential loss with a fixed step size* $\eta > 0$. *Then, there exists a constant* $T \geq 0$ *that depends only on* $\boldsymbol{\theta}^{(0)}$ *and* η, *such that after at most* T *iterations, the network satisfies Assumption D.1 or Assumption D.2.*

Proof. For a network that satisfies Assumption D.11, the values of the network on the training points are

$$\Phi(\boldsymbol{\theta}, x_1) = [-w_1 + b_1]_+ - [-w_2 + b_2]_+ = w_2 - b_2,$$
$$\Phi(\boldsymbol{\theta}; x_2) = [w_1 + b_1]_+ - [w_2 + b_2]_+ = w_1 + b_1 - w_2 - b_2,$$

and the training loss is therefore

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\exp(w_2 - b_2) + \frac{1}{2}\exp(w_2 + b_2 - w_1 - b_1).$$

With the above, we can compute the partial derivatives to obtain

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_1} = -\frac{1}{2}\exp(w_2 + b_2 - w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_1} = -\frac{1}{2}\exp(w_2 + b_2 - w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_2} = \frac{1}{2}\exp(w_2 - b_2) + \frac{1}{2}\exp(w_2 + b_2 - w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_2} = -\frac{1}{2}\exp(w_2 - b_2) + \frac{1}{2}\exp(w_2 + b_2 - w_1 - b_1).$$

Thus, the GD updates are given by

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta \exp\left(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}\right), \tag{43}$$
$$b_1^{(t+1)} = b_1^{(t)} + \frac{1}{2}\eta \exp\left(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}\right), \tag{44}$$
$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right) - \frac{1}{2}\eta \exp\left(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}\right), \tag{45}$$
$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta \exp\left(w_2^{(t)} - b_2^{(t)}\right) - \frac{1}{2}\eta \exp\left(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}\right). \tag{46}$$

By the above update rules and Lemma B.2, we have that the signs of the weights are invariant, as well as the breakpoint $\beta_1^{(t)}$. Since $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$, the second neuron cannot turn inactive on x_1, and we thus deduce that we must remain under the conditions of Assumption D.11, or transition to Assumption D.1 or Assumption D.2. We will now show that we cannot remain in Assumption D.11 for more than a constant number of iterations.

From Equations (43,44) we get

$$u_1^{(t+1)} = w_1^{(t+1)} + b_1^{(t+1)} = w_1^{(t)} + b_1^{(t)} + \eta \exp(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}) = u_1^{(t)} + \eta \exp(u_2^{(t)} - u_1^{(t)}), \tag{47}$$

and from Equations (45,46) we obtain

$$u_2^{(t+1)} = w_2^{(t+1)} + b_2^{(t+1)} = w_2^{(t)} + b_2^{(t)} - \eta \exp(w_2^{(t)} + b_2^{(t)} - w_1^{(t)} - b_1^{(t)}) = u_2^{(t)} - \eta \exp(u_2^{(t)} - u_1^{(t)}). \tag{48}$$

Using the shorthand $S \coloneqq u_1^{(0)} + u_2^{(0)}$, we observe that

$$u_1^{(t+1)} + u_2^{(t+1)} = u_1^{(t)} + \eta \exp\left(u_2^{(t)} - u_1^{(t)}\right) + u_2^{(t)} - \eta \exp\left(u_2^{(t)} - u_1^{(t)}\right) = u_1^{(t)} + u_2^{(t)}.$$

By induction we get that $u_1^{(t)} + u_2^{(t)} = S$ for all t. Substituting $u_1^{(t)} = S - u_2^{(t)}$ into the update rule for u_2 in Equation (48), we have

$$u_2^{(t+1)} = u_2^{(t)} - \eta \exp\left(2u_2^{(t)} - S\right) = u_2^{(t)} - \eta \exp(-S) \exp\left(2u_2^{(t)}\right).$$

Applying Lemma B.4 to the above recursion, we have that $u_2^{(t)}$ decreases at a logarithmic rate. In particular, there exists an initialization-dependent constant T such that $u_2^{(T)} < 0$, implying $w_2^{(T)} < -b_2^{(T)}$. Since $w_2^{(T)} < 0$, we divide by $w_2^{(T)}$ to obtain

$$\beta_2^{(T)} < 1,$$

implying that the second neuron is no longer active on x_2. □

D.2.3 Both neurons are right-active

Assumption D.13 (Both neurons are right-active)**.** *Given the training set $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$, consider the network $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$. We assume that at initialization we have $w_1^{(0)} > 0$, $w_2^{(0)} > 0$, $\beta_1^{(0)} \in (-1, 1)$ and $\beta_2^{(0)} < -1$, and that $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$.*

Proposition D.14. *Under Assumption D.13, consider the optimization dynamics of GD on the exponential loss with a fixed step size $\eta > 0$. Then, there exists a constant $T \geq 0$ that depends only on $\boldsymbol{\theta}^{(0)}$ and η, such that after at most T iterations, the network satisfies either Assumption D.1, Assumption D.2 or Assumption D.11.*

Proof. For a network that satisfies Assumption D.13, the training dynamics are identical to those analyzed in a previous case (see Appendix D.2.2).

By the update rules in Equations (43–46) and by Lemma B.2, we have that the sign of $w_1^{(t)}$ is invariant, as well as the breakpoint $\beta_1^{(t)}$. Since $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$, the second neuron cannot turn inactive on x_1, and we thus deduce that we must remain under the conditions of Assumption D.13, or transition to either one of Assumption D.1, Assumption D.2 or Assumption D.11. We will now show that we cannot remain in Assumption D.13 for more than a constant number of iterations.

Recall the shorthand $S \coloneqq u_1^{(0)} + u_2^{(0)}$, we plug $u_2^{(t)} = S - u_1^{(t)}$ in Equation (47), to obtain

$$u_1^{(t+1)} = u_1^{(t)} + \eta \exp\left(S - 2u_1^{(t)}\right) = u_1^{(t)} + \eta \exp(S) \exp\left(-2u_1^{(t)}\right).$$

Applying Lemma B.4 to the above recursion, we have that

$$u_1^{(t)} - u_1^{(0)} = \sum_{i=0}^{t-1} \eta \exp(S) \exp\left(-2u_1^{(i)}\right) \geq C \ln(t),$$

for some initialization-dependent constant $C > 0$. Let T be large enough such that $C \ln(T) > 2w_2^{(0)}$, then by Equation (45) we have

$$\begin{aligned} w_2^{(T)} &= w_2^{(0)} - \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp\left(v_2^{(i)}\right) - \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp(S) \exp\left(-2u_1^{(i)}\right) \\ &\leq w_2^{(0)} - \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp(S) \exp\left(-2u_1^{(i)}\right) < w_2^{(0)} - w_2^{(0)} = 0, \end{aligned}$$

and therefore $w_2^{(t)}$ becomes negative after at most T iterations, so we are no longer under Assumption D.13. □

D.2.4 BOTH NEURONS ARE LEFT-ACTIVE

Assumption D.15 (Both neurons are left-active)**.** *Given the training set* $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$*, consider the network* $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$*. We assume that at initialization we have* $w_1^{(0)} < 0$*,* $w_2^{(0)} < 0$*,* $\beta_1^{(0)} > 1$ *and* $\beta_2^{(0)} \in (-1, 1)$*, and that* $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ *for all* $t \geq 0$*.*

Proposition D.16. *Under Assumption D.15, consider the optimization dynamics of GD on the exponential loss with a fixed step size* $\eta > 0$*. Then, there exists a constant* $T \geq 0$ *that depends only on* $\boldsymbol{\theta}^{(0)}$ *and* η*, such that after at most* T *iterations, the network satisfies either Assumption D.1, Assumption D.2 or Assumption D.9.*

Proof. For a network that satisfies Assumption D.15, the training dynamics are identical to those analyzed in a previous case (see Appendix D.2.1).

By the update rules in Equations (37–40) and by Lemma B.2, we have that the sign of $w_2^{(t)}$ is invariant, as well as the breakpoint $\beta_2^{(t)}$. Since $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$ for all $t \geq 0$, the first neuron cannot turn inactive on x_2, and we thus deduce that we must remain under the conditions of Assumption D.15, or transition to either one of Assumption D.1, Assumption D.2 or Assumption D.9. We will now show that we cannot remain in Assumption D.15 for more than a constant number of iterations.

Recall the shorthand $S \coloneqq v_1^{(0)} + v_2^{(0)}$, we plug $v_1^{(t)} = S - v_2^{(t)}$ in Equation (42), to obtain

$$v_2^{(t+1)} = v_2^{(t)} - \eta \exp\left(2v_2^{(t)} - S\right) = v_2^{(t)} - \eta \exp(-S) \exp\left(2v_2^{(t)}\right).$$

Applying Lemma B.4 to the above recursion, we have that

$$v_2^{(t)} - v_2^{(0)} = -\sum_{i=0}^{t-1} \eta \exp(-S) \exp\left(2v_2^{(i)}\right) \leq -C \ln(t),$$

for some initialization-dependent constant $C > 0$. Let T be large enough such that $C \ln(T) > -2w_1^{(0)}$, then by Equation (37) we have

$$\begin{aligned} w_1^{(T)} &= w_1^{(0)} + \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp\left(-u_1^{(i)}\right) + \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp(-S) \exp\left(2v_2^{(i)}\right) \\ &\geq w_1^{(0)} + \frac{1}{2} \sum_{i=0}^{T-1} \eta \exp(-S) \exp\left(2v_2^{(i)}\right) > w_1^{(0)} - w_1^{(0)} = 0, \end{aligned}$$

and therefore $w_1^{(t)}$ changes sign becomes positive after at most T iterations, so we are no longer under Assumption D.15. □

D.3 BOTH NEURONS ARE ACTIVE ON BOTH TRAINING POINTS

Assumption D.17 (Both neurons are both active)**.** *Given the training set* $(x_1, y_1) = (-1, -1)$, $(x_2, y_2) = (1, 1)$*, suppose that the network* $\Phi(\boldsymbol{\theta}^{(0)}; x) = \left[w_1^{(0)} x + b_1^{(0)}\right]_+ - \left[w_2^{(0)} x + b_2^{(0)}\right]_+$ *satisfies* $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$*, and that both neurons are active on both instances.*

Proposition D.18. *Under Assumption D.17, consider GD on the exponential loss with a fixed step size* $\eta > 0$*. Then, there exists* $T \geq 0$*, depending only on* $\boldsymbol{\theta}^{(0)}$ *and* η*, such that after at most* T *iterations, at least one neuron is inactive on at least one training instance.*

Proof. We begin with evaluating $\Phi(\boldsymbol{\theta}^{(t)}; x)$ at the data points $x_1 = -1$ and at $x_2 = 1$. Since both neurons are active on both instances, we have

$$\begin{aligned} \Phi\left(\boldsymbol{\theta}^{(t)}; x_1\right) &= \left[-w_1^{(t)} + b_1^{(t)}\right]_+ - \left[-w_2^{(t)} + b_2^{(t)}\right]_+ = -w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}, \\ \Phi\left(\boldsymbol{\theta}^{(t)}; x_2\right) &= \left[w_1^{(t)} + b_1^{(t)}\right]_+ - \left[w_2^{(t)} + b_2^{(t)}\right]_+ = w_1^{(t)} + b_1^{(t)} - w_2^{(t)} - b_2^{(t)}. \end{aligned}$$

The loss at iteration t is thus given by

$$\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) = \frac{1}{2}\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)}\right). \quad (49)$$

Using the shorthands $A_t \coloneqq \exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right)$ and $B_t \coloneqq \exp\left(-w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)}\right)$, the partial derivatives can be written compactly as

$$\begin{aligned}
\frac{\partial\mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_1^{(t)}} &= -\frac{1}{2}A_t - \frac{1}{2}B_t,\\
\frac{\partial\mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_1^{(t)}} &= \frac{1}{2}A_t - \frac{1}{2}B_t,\\
\frac{\partial\mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_2^{(t)}} &= \frac{1}{2}A_t + \frac{1}{2}B_t,\\
\frac{\partial\mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_2^{(t)}} &= -\frac{1}{2}A_t + \frac{1}{2}B_t.
\end{aligned}$$

Using the above to compute the gradient step update for each parameter, we get

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta A_t + \frac{1}{2}\eta B_t, \quad (50)$$

$$b_1^{(t+1)} = b_1^{(t)} - \frac{1}{2}\eta A_t + \frac{1}{2}\eta B_t, \quad (51)$$

$$\begin{aligned}
w_2^{(t+1)} &= w_2^{(t)} - \frac{1}{2}\eta A_t - \frac{1}{2}\eta B_t,\\
b_2^{(t+1)} &= b_2^{(t)} + \frac{1}{2}\eta A_t - \frac{1}{2}\eta B_t.
\end{aligned}$$

We now bound the difference $v_1^{(t)} \coloneqq w_1^{(t)} - b_1^{(t)}$, by subtracting Equation (51) from Equation (50) to obtain

$$v_1^{(t+1)} = v_1^{(t)} + \eta A_t. \quad (52)$$

Additionally, summing the gradient updates of the weights and the biases separately, we get

$$\begin{aligned}
w_1^{(t+1)} + w_2^{(t+1)} &= w_1^{(t)} + w_2^{(t)},\\
b_1^{(t+1)} + b_2^{(t+1)} &= b_1^{(t)} + b_2^{(t)}.
\end{aligned}$$

Recursively applying the above, we arrive at

$$\begin{aligned}
w_1^{(t)} + w_2^{(t)} &= w_1^{(0)} + w_2^{(0)},\\
b_1^{(t)} + b_2^{(t)} &= b_1^{(0)} + b_2^{(0)}.
\end{aligned}$$

By adding the right-hand-side and subtracting the left-hand-side of the above, we can simplify A_t as follows

$$\begin{aligned}
A_t &= \exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) = \exp\left(-2w_1^{(t)} + 2b_1^{(t)} + w_1^{(0)} - b_1^{(0)} + w_2^{(0)} - b_2^{(0)}\right)\\
&= \exp\left(v_1^{(0)} + v_2^{(0)}\right)\exp\left(-2v_1^{(t)}\right).
\end{aligned}$$

Plugging the above in Equation (52), we obtain the recursion

$$v_1^{(t+1)} = v_1^{(t)} + \eta\exp\left(v_1^{(0)} + v_2^{(0)}\right)\exp\left(-2v_1^{(t)}\right). \quad (53)$$

By our assumption that both neurons are active on both instances, we have that $b_1^{(0)} > 0$ (since otherwise at least one of $w_1^{(0)}x_1 + b_1^{(0)}$ and $w_1^{(0)}x_2 + b_1^{(0)}$ is negative). Moreover, we have that $|w_1^{(0)}| < b_1^{(0)}$ (since otherwise the breakpoint of the first neuron which is $-b_1^{(0)}/w_1^{(0)}$ will be in

$[-1, 1]$). We therefore have that $v_1^{(0)} < 0$, and by Equation (53) we get that it is monotonically increasing.

Using Lemma B.4, we have that $v_1^{(t)} - v_1^{(0)} \geq C \ln(t)$ for all $t \geq 1$ and some $C > 0$ that depends only on $\boldsymbol{\theta}^{(0)}$ and η. In particular, for some T large enough such that $C \ln(T) \geq -v_1^{(0)}$, it must hold that $v_1^{(T)} \geq 0$, implying $|w_1^{(T)}| \geq b_1^{(T)}$. Namely, it holds that $-b_1^{(T)}/w_1^{(T)} \in [-1, 1]$, and that the first neuron has turned inactive on one of the instances, concluding the proof of the proposition. □

E Proofs for Section 5

E.1 Proof of Theorem 5.1

Before we prove the theorem, we will state and prove the following proposition, which proves that for almost all initializations $\boldsymbol{\theta}^{(0)}$ such that $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$, the limiting equilibrium point to which GD converges is nonzero. Additionally, this proposition also allows us to ignore certain pathological edge cases of Lebesgue measure zero.

Proposition E.1. *Consider optimizing the objective in Equation (2) using GD with step size $\eta \in (0, 0.5]$. Then, for almost all $\boldsymbol{\theta}^{(0)}$ such that $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$ and for all $t \geq 0$, the GD iterates $\boldsymbol{\theta}^{(t)} = (w_1^{(t)}, b_1^{(t)}, w_2^{(t)}, b_2^{(t)})$ satisfy*

1. $|w_1^{(t)}| \neq |b_1^{(t)}|$,
2. $|w_2^{(t)}| \neq |b_2^{(t)}|$,
3. $w_1^{(t)} \neq 0$ *and* $w_2^{(t)} \neq 0$.
4. $w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)} \neq 0$,
5. $\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) < 0.5$.

Proof. We will first prove that if all items hold at an iteration t, then the last item also holds at the next iteration $t + 1$.

The assumption $\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) < 0.5$ and Lemma B.3 force the first neuron to be active on x_2 and the second neuron to be active on x_1. Thus, the only possible activation patterns for the network at such an iteration are when the first neuron is active only on x_2 or on both instances, and that the second neuron is active on x_1 or on both instances, for a total of four possible patterns (ignoring degenerate cases where breakpoints are ill-defined or coincide with data points due to Items (1–3)).

- Starting from the case where each neuron is active on one instance, we compute $\Phi(\boldsymbol{\theta}^{(t)}; x)$ at $x_1 = -1$ and $x_2 = 1$ to obtain

$$\Phi(\boldsymbol{\theta}^{(t)}; x_1) = \left[-w_1^{(t)} + b_1^{(t)}\right]_+ - \left[-w_2^{(t)} + b_2^{(t)}\right]_+ = w_2^{(t)} - b_2^{(t)},$$
$$\Phi(\boldsymbol{\theta}^{(t)}, x_2) = \left[w_1^{(t)} + b_1^{(t)}\right]_+ - \left[w_2^{(t)} + b_2^{(t)}\right]_+ = w_1^{(t)} + b_1^{(t)}.$$

  The loss at iteration t is thus given by

$$\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) = \frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right). \tag{54}$$

With the above, we compute the partial derivatives as follows

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_1^{(t)}} = -\frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_1^{(t)}} = -\frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_2^{(t)}} = \frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_2^{(t)}} = -\frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)}\right),$$

which imply the following gradient descent updates

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$b_1^{(t+1)} = b_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta\exp\left(w_2^{(t)} - b_2^{(t)}\right),$$
$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta\exp\left(w_2^{(t)} - b_2^{(t)}\right).$$

Plugging the above in Equation (54), we get the update rule for the loss

$$\mathcal{L}\left(\boldsymbol{\theta}^{(t+1)}\right) = \frac{1}{2}\exp\left(w_2^{(t)} - b_2^{(t)} - \eta\exp\left(w_2^{(t)} - b_2^{(t)}\right)\right) + \frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)} - \eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right)\right),$$

namely, the loss over each instance strictly decreases, and therefore $\mathcal{L}(\boldsymbol{\theta}^{(t+1)}) < \mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$.

- Now suppose that the first neuron is active on both instances, and the second neuron is only active on x_1. Then the outputs of the network on the training points are given by

$$\Phi(\boldsymbol{\theta}, x_1) = [-w_1 + b_1]_+ - [-w_2 + b_2]_+ = -w_1 + b_1 + w_2 - b_2,$$
$$\Phi(\boldsymbol{\theta}; x_2) = [w_1 + b_1]_+ - [w_2 + b_2]_+ = w_1 + b_1,$$

and the training loss is therefore

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\exp\left(-w_1 + b_1 + w_2 - b_2\right) + \frac{1}{2}\exp\left(-w_1 - b_1\right). \tag{55}$$

With the above, we can compute the partial derivatives to obtain

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_1} = -\frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2) - \frac{1}{2}\exp(-w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_1} = \frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2) - \frac{1}{2}\exp(-w_1 - b_1),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial w_2} = \frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2),$$
$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial b_2} = -\frac{1}{2}\exp(-w_1 + b_1 + w_2 - b_2).$$

Thus, the GD updates are given by

$$w_1^{(t+1)} = w_1^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$b_1^{(t+1)} = b_1^{(t)} - \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\eta\exp\left(-w_1^{(t)} - b_1^{(t)}\right),$$
$$w_2^{(t+1)} = w_2^{(t)} - \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right),$$
$$b_2^{(t+1)} = b_2^{(t)} + \frac{1}{2}\eta\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right).$$

Plugging the above in Equation (55), we get the update rule for the loss over the first instance

$$\ell\left(-\Phi\left(\boldsymbol{\theta}^{(t+1)};x_1\right)\right)=\exp\left(-w_1^{(t)}+b_1^{(t)}+w_2^{(t)}-b_2^{(t)}-2\eta\exp\left(-w_1^{(t)}+b_1^{(t)}+w_2^{(t)}-b_2^{(t)}\right)\right),$$

and for the second instance

$$\ell\left(\Phi\left(\boldsymbol{\theta}^{(t+1)};x_2\right)\right)=\exp\left(-w_1^{(t)}-b_1^{(t)}-\eta\exp\left(-w_1^{(t)}-b_1^{(t)}\right)\right).$$

Therefore the training loss is strictly decreasing, and we have $\mathcal{L}(\boldsymbol{\theta}^{(t+1)})<\mathcal{L}(\boldsymbol{\theta}^{(t)})<0.5$.

- Suppose that the first neuron is active on x_2, and the second neuron is active on both instances. Then the outputs of the network on the training points are given by

$$\Phi(\boldsymbol{\theta},x_1)=[-w_1+b_1]_+-[-w_2+b_2]_+=w_2-b_2,$$
$$\Phi(\boldsymbol{\theta};x_2)=[w_1+b_1]_+-[w_2+b_2]_+=w_1+b_1-w_2-b_2,$$

and the training loss is therefore

$$\mathcal{L}(\boldsymbol{\theta})=\frac{1}{2}\exp(w_2-b_2)+\frac{1}{2}\exp(w_2+b_2-w_1-b_1). \tag{56}$$

With the above, we can compute the partial derivatives to obtain

$$\frac{\partial\mathcal{L}(\boldsymbol{\theta})}{\partial w_1}=-\frac{1}{2}\exp(w_2+b_2-w_1-b_1),$$
$$\frac{\partial\mathcal{L}(\boldsymbol{\theta})}{\partial b_1}=-\frac{1}{2}\exp(w_2+b_2-w_1-b_1),$$
$$\frac{\partial\mathcal{L}(\boldsymbol{\theta})}{\partial w_2}=\frac{1}{2}\exp(w_2-b_2)+\frac{1}{2}\exp(w_2+b_2-w_1-b_1),$$
$$\frac{\partial\mathcal{L}(\boldsymbol{\theta})}{\partial b_2}=-\frac{1}{2}\exp(w_2-b_2)+\frac{1}{2}\exp(w_2+b_2-w_1-b_1).$$

Thus, the GD updates are given by

$$w_1^{(t+1)}=w_1^{(t)}+\frac{1}{2}\eta\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}\right),$$
$$b_1^{(t+1)}=b_1^{(t)}+\frac{1}{2}\eta\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}\right),$$
$$w_2^{(t+1)}=w_2^{(t)}-\frac{1}{2}\eta\exp\left(w_2^{(t)}-b_2^{(t)}\right)-\frac{1}{2}\eta\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}\right),$$
$$b_2^{(t+1)}=b_2^{(t)}+\frac{1}{2}\eta\exp\left(w_2^{(t)}-b_2^{(t)}\right)-\frac{1}{2}\eta\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}\right).$$

Plugging the above in Equation (56), we get the update rule for the loss over the first instance

$$\ell\left(-\Phi\left(\boldsymbol{\theta}^{(t+1)};x_1\right)\right)=\exp\left(w_2^{(t)}-b_2^{(t)}-\eta\exp\left(w_2^{(t)}-b_2^{(t)}\right)\right),$$

and for the second instance

$$\ell\left(\Phi\left(\boldsymbol{\theta}^{(t+1)};x_2\right)\right)=\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}-2\eta\exp\left(w_2^{(t)}+b_2^{(t)}-w_1^{(t)}-b_1^{(t)}\right)\right).$$

Therefore the training loss is strictly decreasing, and we have $\mathcal{L}(\boldsymbol{\theta}^{(t+1)})<\mathcal{L}(\boldsymbol{\theta}^{(t)})<0.5$.

- Suppose that both neurons are active on both instances. Then the outputs of the network on the training points are given by

$$\Phi\left(\boldsymbol{\theta}^{(t)};x_1\right)=\left[-w_1^{(t)}+b_1^{(t)}\right]_+-\left[-w_2^{(t)}+b_2^{(t)}\right]_+=-w_1^{(t)}+b_1^{(t)}+w_2^{(t)}-b_2^{(t)},$$
$$\Phi\left(\boldsymbol{\theta}^{(t)};x_2\right)=\left[w_1^{(t)}+b_1^{(t)}\right]_+-\left[w_2^{(t)}+b_2^{(t)}\right]_+=w_1^{(t)}+b_1^{(t)}-w_2^{(t)}-b_2^{(t)}.$$

The loss at iteration t is thus given by

$$\mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right) = \frac{1}{2}\exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right) + \frac{1}{2}\exp\left(-w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)}\right). \tag{57}$$

Using the shorthands $A_t := \exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)}\right)$ and $B_t := \exp\left(-w_1^{(t)} - b_1^{(t)} + w_2^{(t)} + b_2^{(t)}\right)$, the partial derivatives can be written compactly as

$$\begin{aligned}
\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_1^{(t)}} &= -\frac{1}{2}A_t - \frac{1}{2}B_t, \\
\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_1^{(t)}} &= \frac{1}{2}A_t - \frac{1}{2}B_t, \\
\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial w_2^{(t)}} &= \frac{1}{2}A_t + \frac{1}{2}B_t, \\
\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)})}{\partial b_2^{(t)}} &= -\frac{1}{2}A_t + \frac{1}{2}B_t.
\end{aligned}$$

Using the above to compute the gradient step update for each parameter, we get

$$\begin{aligned}
w_1^{(t+1)} &= w_1^{(t)} + \frac{1}{2}\eta A_t + \frac{1}{2}\eta B_t, \\
b_1^{(t+1)} &= b_1^{(t)} - \frac{1}{2}\eta A_t + \frac{1}{2}\eta B_t, \\
w_2^{(t+1)} &= w_2^{(t)} - \frac{1}{2}\eta A_t - \frac{1}{2}\eta B_t, \\
b_2^{(t+1)} &= b_2^{(t)} + \frac{1}{2}\eta A_t - \frac{1}{2}\eta B_t.
\end{aligned}$$

Substituting the above in Equation (57), we can write the loss update rule for the first instance at iteration $t+1$ as

$$\begin{aligned}
\ell\left(y_1\Phi\left(\boldsymbol{\theta}^{(t+1)}; x_1\right)\right) &= \exp\left(-w_1^{(t+1)} + b_1^{(t+1)} + w_2^{(t+1)} - b_2^{(t+1)}\right) \\
&= \exp\left(-w_1^{(t)} + b_1^{(t)} + w_2^{(t)} - b_2^{(t)} - 2\eta A_t\right) \\
&= \exp\left(-2\eta A_t\right) \cdot \ell\left(y_1\Phi\left(\boldsymbol{\theta}^{(t)}; x_1\right)\right),
\end{aligned}$$

and since $A_t > 0$, we obtain that the loss on x_1 strictly decreases with t. Likewise, we can verify that

$$\ell\left(y_2\Phi\left(\boldsymbol{\theta}^{(t+1)}; x_2\right)\right) = \exp\left(-2\eta B_t\right) \cdot \ell\left(y_2\Phi\left(\boldsymbol{\theta}^{(t)}; x_2\right)\right),$$

and therefore $\mathcal{L}(\boldsymbol{\theta}^{(t+1)}) < \mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$.

Fix any $\eta \in (0, 0.5]$, and define the GD map $G : \mathbb{R}^4 \to \mathbb{R}^4$ by $G(\boldsymbol{\theta}) := \boldsymbol{\theta} - \eta\nabla\mathcal{L}(\boldsymbol{\theta})$. We will show that if $\mathcal{L}(\boldsymbol{\theta}) < 0.5$ then this is an analytic open map by analyzing its Jacobian, J_G. We have

$$J_G(\boldsymbol{\theta}) = I_4 - \eta\nabla^2\mathcal{L}(\boldsymbol{\theta}),$$

where I_4 is the four-dimensional identity matrix. For ease of notation, let $\theta_1 := w_1$, $\theta_2 := b_1$, $\theta_3 := w_2$ and $\theta_4 := b_2$. We now bound the entries of the Hessian as follows: For any activation pattern and any $i \in \{1, 2\}$, $\Phi(\boldsymbol{\theta}; x_i)$ computes an affine transformation of the form $\sum_{j=1}^{4} \xi_{i,j}\theta_j$ for some $\xi_{i,j} \in \{-1, 0, 1\}$, $j \in [4]$. From this, we obtain that G is analytic (it is the difference between the identity map and a composition between an affine transformation and an exponent) wherever it is differentiable, and we also have that for all $j_1, j_2 \in [4]$,

$$\frac{\partial^2}{\partial\theta_{j_1}\partial\theta_{j_2}}\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\xi_{1,j_1}\xi_{1,j_2}\exp\left(\Phi\left(\boldsymbol{\theta}; x_1\right)\right) + \frac{1}{2}\xi_{2,j_1}\xi_{2,j_2}\exp\left(-\Phi\left(\boldsymbol{\theta}; x_2\right)\right).$$

Bounding the above in absolute value yields

$$\left|\frac{\partial^2}{\partial\theta_{j_1}\partial\theta_{j_2}}\mathcal{L}(\boldsymbol{\theta})\right| \leq \frac{1}{2}\exp\left(\Phi\left(\boldsymbol{\theta};x_1\right)\right) + \frac{1}{2}\exp\left(-\Phi\left(\boldsymbol{\theta};x_2\right)\right) = \mathcal{L}(\boldsymbol{\theta}) < 0.5.$$

Thus, for $\eta \in (0, 0.5]$ we have that $J_G(\boldsymbol{\theta})$ is strictly diagonally dominant, and hence non-singular. This implies that G is an open map on any domain where it is differentiable.

We now prove Items (1–5) by induction. First, we have that $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$ holds by our assumption, and that if Item 5 holds for t then it also holds for $t+1$ by the previous case analysis.

By removing the measure zero set where Items (1–4) are violated, we have that the induction base holds for $t = 0$. For the induction step, assume that these items hold for t. Since $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$, we have that G is an analytic open map, and its composition with itself $t+1$ times, $G^{t+1} \coloneqq G \circ \ldots \circ G$, is also an analytic open map. Thus, the maps

- $\beta_{1,\pm}^{t+1}(\boldsymbol{\theta}^{(0)}) \coloneqq G_1^{t+1}(\boldsymbol{\theta}^{(0)}) \pm G_2^{t+1}(\boldsymbol{\theta}^{(0)})$,
- $\beta_{2,\pm}^{t+1}(\boldsymbol{\theta}^{(0)}) \coloneqq G_3^{t+1}(\boldsymbol{\theta}^{(0)}) \pm G_4^{t+1}(\boldsymbol{\theta}^{(0)})$,
- $S^{t+1} \coloneqq G_1^{t+1}(\boldsymbol{\theta}^{(0)}) - G_2^{t+1}(\boldsymbol{\theta}^{(0)}) + G_3^{t+1}(\boldsymbol{\theta}^{(0)}) + G_4^{t+1}(\boldsymbol{\theta}^{(0)})$,
- $G_1^{t+1}(\boldsymbol{\theta}^{(0)})$ and $G_3^{t+1}(\boldsymbol{\theta}^{(0)})$,

where G_j^{k+1} returns the jth coordinate of G^{t+1}, are all analytic. Moreover, none of these can be the zero map, since G^{t+1} is an open map, and thus any open set is mapped by G^{t+1} to an open set U. Let $\boldsymbol{\theta} = (w_1, b_1, w_2, b_2) \in U$ be arbitrary, we can choose $\varepsilon > 0$ small enough such that $(w_1 + \varepsilon, b_1, w_2, b_2) \in U$, and thus either one of $w_1 - b_1 + w_2 + b_2 \neq 0$ or $w_1 + \varepsilon - b_1 + w_2 + b_2 \neq 0$ hold, and thus S^{t+1} cannot collapse an open set to zero, and it is not the zero map. A similar argument proves that the remaining maps are not the zero map either, and therefore it holds that the zero-set (the preimage of zero) of these maps has measure zero (Mityagin, 2015). By removing these measure-zero sets from the set of valid initializations for $\boldsymbol{\theta}^{(0)}$, we have that items (1–4) hold. Finally, by taking an infinite yet countable union over the measure-zero sets removed at each iteration of GD, we are removing a set of measure zero from the set $\{\boldsymbol{\theta}^{(0)} : \mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5\}$, which concludes the proof of the proposition. □

With the above proposition, we turn to prove our main theorem.

Proof of Theorem 5.1. By Proposition E.1, we have for almost all $\boldsymbol{\theta}^{(0)}$ such that $\mathcal{L}(\boldsymbol{\theta}^{(0)}) < 0.5$, that the GD iterates $\boldsymbol{\theta}^{(t)}$ satisfy Items (1–5). In particular, the breakpoints $\beta_1^{(t)}, \beta_2^{(t)}$ are well-defined and never overlap a data instance for all $t \geq 0$, the loss at any iteration $t \geq 0$ satisfies $\mathcal{L}(\boldsymbol{\theta}^{(t)}) < 0.5$, and $v_1^{(t)} + u_2^{(t)} \neq 0$.

By the previous conditions and Lemma B.3, the first neuron must be active on x_2 and the second neuron must be active on x_1. In addition, each neuron may or may not be active on the remaining instance. We split our analysis according to the activation pattern as follows:

1. If both neurons are active on both instances, then Assumption D.17 holds. By Proposition D.18, after a constant number of iterations, we transition to the setting of Assumption D.13 or Assumption D.15, which is analyzed in Item 2; to the setting of Assumption D.9 or Assumption D.11, which is analyzed in Item 3; to the setting of Assumption D.2, which is analyzed in Item 4; or to the setting of Assumption D.1, which is analyzed in Item 5.
2. If both neurons point to the same direction ($\operatorname{sign}(w_1^{(t)}) = \operatorname{sign}(w_2^{(t)})$), then Assumption D.13 or Assumption D.15 holds. By Propositions D.14,D.16, after a constant number of iterations, we transition to the setting of Assumption D.9 or Assumption D.11, which is analyzed in Item 3; to the setting of Assumption D.2, which is analyzed in Item 4; or to the setting of Assumption D.1, which is analyzed in Item 5.

3. If one neuron has specialized and the other points to the opposite direction and is active on both instances, then Assumption D.9 or Assumption D.11 holds. By Propositions D.10,D.12, after a constant number of iterations, we transition to the setting of Assumption D.2, which is analyzed in Item 4; or to the setting of Assumption D.1, which is analyzed in Item 5.

4. If both neurons have specialized and are inactive on a subinterval of $(-1, 1)$, Assumption D.2 holds. By Proposition D.8, after a constant number of iterations, we transition to the setting of Assumption D.1, which is analyzed in Item 5.

5. Finally, if both neurons have specialized and are active on a subinterval of $(-1, 1)$, Assumption D.1 holds. By Proposition D.6, we have

$$x^\star(t) = \Theta\left(\frac{1}{\ln(t)}\right).$$

Since the settings in Items (1–4) can only hold for a constant number of iterations before transitioning to one of the next items, after a constant number of iterations, we transition to the final setting in Item 5, from which the theorem follows. □

E.2 Proof of Theorem 5.2

Before we prove the theorem, we will need a few auxiliary lemmas. The following lemma provides explicit constants for bounding the growth rate of the weights.

Lemma E.2. *Suppose that $w_1^{(0)} \in (0, 3)$, $w_2^{(0)} \in (-2, 0)$, and $b_1^{(0)} = b_2^{(0)} = 0$. This is the boundary case $\beta_1^{(0)} = \beta_2^{(0)} = 0$ of Assumption D.1. Then*

$$w_1^{(t)} - w_2^{(t)} \le 5 + \ln\left(1 + 2\exp\left(\eta\right)\eta t\right).$$

Proof. Using Lemma D.4 and Lemma D.5, and slightly loosening their upper bounds, we get

$$w_1^{(t)} - w_2^{(t)} \le \frac{1}{2}\ln\left(\exp\left(2w_1^{(0)}\right) + 2\eta C_1 \exp\left(v_1^{(0)}\right)\cdot t\right) + \frac{1}{2}\ln\left(\exp\left(-2w_2^{(0)}\right) + 2\eta\exp\left(-u_2^{(0)}\right) C_2 \cdot t\right)$$

where $C_1 = \exp\left(\eta\exp\left(v_1^{(0)}\right)\exp\left(-2w_1^{(0)}\right)\right)$ and $C_2 = \exp\left(\eta\exp\left(-u_2^{(0)}\right)\exp\left(2w_2^{(0)}\right)\right)$.
Since $b_1^{(0)} = b_2^{(0)} = 0$ we can simplify the constant to get

$$w_1^{(t)} - w_2^{(t)} \le \frac{1}{2}\ln\left(\exp\left(2w_1^{(0)}\right) + 2\eta C_1 \exp\left(w_1^{(0)}\right)\cdot t\right) + \frac{1}{2}\ln\left(\exp\left(-2w_2^{(0)}\right) + 2\eta\exp\left(-w_2^{(0)}\right) C_2 \cdot t\right)$$

where $C_1 = \exp\left(\eta\exp\left(-w_1^{(0)}\right)\right)$ and $C_2 = \exp\left(\eta\exp\left(w_2^{(0)}\right)\right)$. Merging the $\ln$ functions we get

$$w_1^{(t)} - w_2^{(t)} \le \frac{1}{2}\ln\left(\left[\exp\left(2w_1^{(0)}\right) + 2\eta C_1 \exp\left(w_1^{(0)}\right)\cdot t\right]\left[\exp\left(-2w_2^{(0)}\right) + 2\eta\exp\left(-w_2^{(0)}\right) C_2 \cdot t\right]\right)$$

Factor the leading exponential in each term:

$$\exp\left(2w_1^{(0)}\right) + 2\eta C_1 \exp\left(w_1^{(0)}\right)\cdot t = \exp\left(2w_1^{(0)}\right)\left(1 + 2\eta C_1 \exp\left(-w_1^{(0)}\right)\cdot t\right)$$
$$\exp\left(-2w_2^{(0)}\right) + 2\eta\exp\left(-w_2^{(0)}\right) C_2 \cdot t = \exp\left(-2w_2^{(0)}\right)\left(1 + 2\eta\exp\left(w_2^{(0)}\right) C_2 \cdot t\right)$$

Plugging it in the bound we get

$$w_1^{(t)} - w_2^{(t)} \le \frac{1}{2}\ln\left(\left[\exp\left(2w_1^{(0)} - 2w_2^{(0)}\right)\right]\left[1 + 2\eta C_1 \exp\left(-w_1^{(0)}\right)\cdot t\right]\left[1 + 2\eta\exp\left(w_2^{(0)}\right) C_2 \cdot t\right]\right).$$

Once again by using logarithm rules we get

$$w_1^{(t)} - w_2^{(t)} \le w_1^{(0)} - w_2^{(0)} + \frac{1}{2}\ln\left(\left[1 + 2\eta C_1 \exp\left(-w_1^{(0)}\right)\cdot t\right]\left[1 + 2\eta\exp\left(w_2^{(0)}\right) C_2 \cdot t\right]\right).$$

Using the inequality $(1+A)(1+B) \leq (1+\max\{A,B\})^2$ for all $A, B > 0$ we get

$$w_1^{(t)} - w_2^{(t)} \leq w_1^{(0)} - w_2^{(0)} + \ln\left(1 + \eta A t\right).$$

where $A = \max\{2C_1 \exp\left(-w_1^{(0)}\right), 2C_2 \exp\left(w_2^{(0)}\right)\}$. Using the assumptions that $w_1^{(0)} \in (0,3)$ and $w_2^{(0)} \in (-2,0)$ we get $w_1^{(0)} - w_2^{(0)} < 5$ and $A < 2\exp(\eta)$ and thus

$$w_1^{(t)} - w_2^{(t)} \leq 5 + \ln\left(1 + 2\exp(\eta)\,\eta t\right).$$

□

The following lemma shows that under a zero biases assumption and a tamed step size, the convergence rate of the system to a balanced GD step for both neurons is monotone.

Lemma E.3. *Suppose that $w_2^{(0)} < 0 < w_1^{(0)}$ and $b_1^{(0)} = b_2^{(0)} = 0$, i.e., the boundary case $\beta_1^{(0)} = \beta_2^{(0)} = 0$ of Assumption D.1. If we optimize Equation (2) using GD with step size $\eta \in (0, 0.5]$, then*

$$\lim_{t\to\infty} b_2^{(t)} - b_1^{(t)} = \lim_{t\to\infty} w_1^{(t)} + w_2^{(t)} = \frac{v_1^{(0)} + u_2^{(0)}}{2}.$$

where the convergence is monotone.

Proof. The specialized update rules hold in this boundary case, so the proof of Lemma D.7 applies unchanged and gives both limits. It remains to prove monotonicity. Since $w_1^{(t)} + w_2^{(t)} - b_1^{(t)} + b_2^{(t)} = v_1^{(0)} + u_2^{(0)}$ by Equations (24–27), it suffices to prove monotonicity for just the sum of the weights.

For the sake of brevity, we define $S_t \coloneqq w_1^{(t)} + w_2^{(t)}$. Compute

$$\begin{aligned}
S_{t+1} &= w_1^{(t+1)} + w_2^{(t+1)} \\
&= w_1^{(t)} + \frac{1}{2}\eta\exp(-2w_1^{(t)} + v_1^{(0)}) + w_2^{(t)} - \frac{1}{2}\eta\exp(2w_2^{(t)} - u_2^{(0)}) \\
&= S_t + \frac{1}{2}\eta\left[\exp(-2w_1^{(t)} + v_1^{(0)}) - \exp(2w_2^{(t)} - u_2^{(0)})\right]
\end{aligned}$$

Define $D_t \coloneqq w_1^{(t)} - w_2^{(t)} > 0$. We can rewrite $w_1^{(t)} = \frac{1}{2}(S_t + D_t),\ w_2^{(t)} = \frac{1}{2}(S_t - D_t)$, and thus we can also rewrite $S_{t+1} - S_t$ using $v_1^{(0)}, u_2^{(0)}, S_t$ and D_t as follows

$$\begin{aligned}
S_{t+1} - S_t &= \frac{1}{2}\eta\left[\exp(-2w_1^{(t)} + v_1^{(0)}) - \exp(2w_2^{(t)} - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\left[\exp(-S_t - D_t + v_1^{(0)}) - \exp(S_t - D_t - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\exp(-D_t)\left[\exp(v_1^{(0)} - S_t) - \exp(S_t - u_2^{(0)})\right] \\
&= \frac{1}{2}\eta\exp(-D_t)g(S_t)
\end{aligned}$$

where $g(s) \coloneqq \exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)})$. We notice that if $S_t = S^\star = \frac{v_1^{(0)} + u_2^{(0)}}{2}$, then $S_{t+1} = S_t$, implying that this is a fixed point. Since $\frac{1}{2}\eta\exp(-D_t) > 0$, the only term that affects the sign of $S_{t+1} - S_t$ is $g(S_t)$. If $S_t < S^\star$ then $g(S_t) > 0$ which implies that $S_{t+1} > S_t$; and if $S_t > S^\star$, then $g(S_t) < 0$, which implies that $S_{t+1} < S_t$. Assume without loss of generality that $S_t < S^\star$ (the other direction is symmetric). We have that $g(s)$ is decreasing since

$$g'(s) = -\exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)}) < 0.$$

Denote $L \coloneqq \sup_{s\in[S_0, S^\star]} |g'(s)|$. We will now compute L. We have $g''(s) = \exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)})$. Equating this to zero to find the extrema points, we get that $S^\star$ is an extremum point of $g'(s)$. Since

$$g^{(3)}(s) = -\exp(v_1^{(0)} - s) - \exp(s - u_2^{(0)}) < 0,$$

then $S^\star$ is a maximum point of $g'(s)$. But since $g'(s) < 0$, then $S^\star$ is a minimum point of $|g'(s)|$. This implies that the maximum point in the interval must be attained at the boundary, thus

$$L = \max\{\exp(v_1^{(0)} - S_0) + \exp(S_0 - u_2^{(0)}), \exp(v_1^{(0)} - S^\star) + \exp(S^\star - u_2^{(0)})\}.$$

Since $g(S)$ is L-Lipschitz, we get

$$|g(S_t) - g(S^\star)| \leq L|S_t - S^\star|.$$

But $g(S)$ is also decreasing, and $g(S^\star) = 0$ and $g(S_t) \geq g(S^\star) = 0$, so we get that

$$g(S_t) = g(S_t) - g(S^\star) \leq L(S^\star - S_t),$$

and thus by the above equation we compute

$$S_{t+1} - S_t = \frac{1}{2}\eta \exp(-D_t) g(S_t) \leq \frac{1}{2}\eta L(S^\star - S_t)\exp(-D_t).$$

Rearranging yields

$$S_{t+1} < \frac{1}{2}\eta L \exp(-D_t) S^\star + (1 - \frac{1}{2}\eta L \exp(-D_t)) S_t,$$

and further simplifying, we have

$$S_{t+1} - S^\star < (1 - \frac{1}{2}\exp(-D_t)\eta L)(S_t - S^\star).$$

Combining Assumption D.1 and Proposition D.3, we have that $w_1^{(t)} > 0$ and increases, and that $w_2^{(t)} < 0$ and decreases, hence $\exp(-D_t)$ decreases, and we can upper bound it by $\exp(-D_0) = \exp(w_2^{(0)} - w_1^{(0)})$. Plugging this upper bound in the above displayed equation, we get

$$S_{t+1} - S^\star < (1 - \frac{1}{2}\eta L \exp(w_2^{(0)} - w_1^{(0)}))(S_t - S^\star). \tag{58}$$

Next, we upper bound L as follows. The assumption that $b_1^{(0)} = b_2^{(0)} = 0$ yields

$$\exp\left(v_1^{(0)} - S^\star\right) + \exp(S^\star - u_2^{(0)}) = 2\exp\left(\frac{w_1^{(0)} - w_2^{(0)}}{2}\right).$$

Similarly, we have

$$\exp\left(v_1^{(0)} - S_0\right) + \exp\left(S_0 - u_2^{(0)}\right) = \exp\left(-w_2^{(0)}\right) + \exp\left(w_1^{(0)}\right).$$

Thus, when the above is the maximum in L, we have

$$\frac{\exp(w_1^{(0)} - w_2^{(0)})}{L} = \left(\exp\left(w_2^{(0)}\right) + \exp\left(-w_1^{(0)}\right)\right)^{-1} \geq \frac{1}{2},$$

and if the maximum in L is attained by the second argument, we have

$$\frac{\exp(w_1^{(0)} - w_2^{(0)})}{L} = \frac{1}{2}\exp\left(\frac{w_1^{(0)} - w_2^{(0)}}{2}\right) > \frac{1}{2}.$$

Therefore, in both cases, we have by our assumption that $\eta \leq 0.5 \leq \frac{\exp(w_1^{(0)} - w_2^{(0)})}{L}$, implying that $1 - \frac{1}{2}\eta L \exp(w_2^{(0)} - w_1^{(0)}) \geq 0$, and thus by Equation (58) and by using the assumption that $S_t < S^\star$, we conclude that

$$S_{t+1} \in (S_t, S^\star].$$

□

The following proposition shows that under a zero biases assumption, the difference between the biases throughout optimization increases at a linear rate to a constant that depends on the sum of the weights.

Proposition E.4. *Suppose that $w_2^{(0)} < 0 < w_1^{(0)}$, $w_1^{(0)} + w_2^{(0)} > 1$, and $b_1^{(0)} = b_2^{(0)} = 0$, i.e., the boundary case $\beta_1^{(0)} = \beta_2^{(0)} = 0$ of Assumption D.1. If we optimize Equation (2) using GD with step size $\eta \in (0, 0.3]$, then*

$$b_2^{(t)} - b_1^{(t)} \geq \frac{w_1^{(0)} + w_2^{(0)}}{11} - \frac{1}{10}$$

for all $t \geq t_0$, where

$$t_0 \coloneqq \left\lfloor \frac{w_1^{(0)} + w_2^{(0)}}{2\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right)} \right\rfloor.$$

Proof. Consider $\frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right)$. By the monotonicity of the weights in Proposition D.3, $\frac{1}{2}\eta \exp\left(-w_1^{(t)} - b_1^{(t)}\right) \leq \frac{1}{2}\eta \exp\left(-w_1^{(0)} - b_1^{(0)}\right) = \frac{1}{2}\eta \exp\left(-w_1^{(0)}\right)$ for all t, so

$$b_1^{(t)} \leq b_1^{(0)} + t \cdot \frac{1}{2}\eta \exp\left(-w_1^{(0)}\right) = t \cdot \frac{1}{2}\eta \exp\left(-w_1^{(0)}\right). \tag{59}$$

On the other hand, as long as $w_2^{(t)} - b_2^{(t)} \geq \frac{-w_1^{(0)} + w_2^{(0)}}{2}$, then the growth rate of $b_2^{(t)}$ can be lower bounded as follows

$$b_2^{(t)} = b_2^{(0)} + \frac{1}{2}\sum_{i=0}^{t-1} \eta \exp\left(w_2^{(i)} - b_2^{(i)}\right) \geq t \cdot \frac{1}{2}\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right), \tag{60}$$

and we also have by subtracting Equation (27) from Equation (26) that

$$w_2^{(t+1)} - b_2^{(t+1)} = w_2^{(t)} - b_2^{(t)} - \eta \exp\left(w_2^{(t)} - b_2^{(t)}\right) \leq w_2^{(t)} - b_2^{(t)} - \eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right),$$

which recursively implies that

$$w_2^{(t)} - b_2^{(t)} \leq w_2^{(0)} - b_2^{(0)} - t\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right) = w_2^{(0)} - t\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right).$$

By the above, $w_2^{(t)} - b_2^{(t)}$ decreases at a linear rate if it is lower bounded, which entails that a necessary condition for $w_2^{(t)} - b_2^{(t)} \geq \frac{-w_1^{(0)} + w_2^{(0)}}{2}$ to hold is

$$\frac{-w_1^{(0)} + w_2^{(0)}}{2} \leq w_2^{(0)} - t\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right).$$

Solving for t, we have

$$t \leq \frac{w_1^{(0)} + w_2^{(0)}}{2\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right)}. \tag{61}$$

The assumptions $w_2^{(0)} < 0 < w_1^{(0)}$ and $w_1^{(0)} + w_2^{(0)} > 1$ imply $-w_1^{(0)} < w_2^{(0)}$, yielding that at the first iteration we have

$$w_2^{(0)} - b_2^{(0)} = w_2^{(0)} \geq \frac{-w_1^{(0)} + w_2^{(0)}}{2},$$

thus Equation (61) holds for at least one iteration. Moreover, since our assumptions also entail

$$\frac{w_1^{(0)} + w_2^{(0)}}{2\eta \exp\left(\frac{-w_1^{(0)} + w_2^{(0)}}{2}\right)} > \frac{1}{2\eta \exp\left(\frac{-1+0}{2}\right)} > \frac{0.3}{\eta} \geq 1,$$

hence we can take the floor function of the right-hand side in Equation (61),

$$\frac{w_1^{(0)}+w_2^{(0)}}{2\eta\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)}-1\le t_0\coloneqq\left\lfloor\frac{w_1^{(0)}+w_2^{(0)}}{2\eta\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)}\right\rfloor\le\frac{w_1^{(0)}+w_2^{(0)}}{2\eta\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)}. \tag{62}$$

We now address the possibility that the threshold is crossed before t_0. Let $\tau\le t_0$ be the first iteration such that

$$w_2^{(\tau)}-b_2^{(\tau)}<\frac{-w_1^{(0)}+w_2^{(0)}}{2}.$$

Since $w_2^{(t)}+b_2^{(t)}=w_2^{(0)}$, we obtain

$$b_2^{(\tau)}=\frac{w_2^{(0)}-\left(w_2^{(\tau)}-b_2^{(\tau)}\right)}{2}>\frac{w_1^{(0)}+w_2^{(0)}}{4}.$$

Together with the monotonicity of $b_2^{(t)}$, Equation (59), and the upper bound on t_0 in Equation (62), this gives

$$b_2^{(t_0)}-b_1^{(t_0)}>\frac{w_1^{(0)}+w_2^{(0)}}{4}\left(1-\exp\left(-\frac{w_1^{(0)}+w_2^{(0)}}{2}\right)\right)\ge\frac{w_1^{(0)}+w_2^{(0)}}{11},$$

which is stronger than the desired bound. We may therefore assume that the threshold is not crossed before t_0, in which case Equation (60) applies at $t=t_0$. Subtracting Equation (59) from Equation (60) yields

$$b_2^{(t)}-b_1^{(t)}\ge t\cdot\frac{1}{2}\eta\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)-t\cdot\frac{1}{2}\eta\exp\left(-w_1^{(0)}\right).$$

Substituting t with t_0, we have

$$b_2^{(t_0)}-b_1^{(t_0)}\ge\frac{1}{2}\eta t_0\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)-\frac{1}{2}\eta t_0\exp\left(-w_1^{(0)}\right). \tag{63}$$

Lower bounding the first summand in the right-hand side of the above using Equation (62), we have

$$\frac{1}{2}\eta t_0\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)\ge\frac{w_1^{(0)}+w_2^{(0)}}{4}-\frac{1}{2}\eta\exp\left(\frac{-w_1^{(0)}+w_2^{(0)}}{2}\right)\ge\frac{w_1^{(0)}+w_2^{(0)}}{4}-0.1, \tag{64}$$

where in the last inequality, we used $\eta\le0.3$ and $-w_1^{(0)}+w_2^{(0)}<1$ which hold by our assumptions.

We now lower bound the second summand in Equation (63). We have

$$-\frac{1}{2}\eta t_0\exp\left(-w_1^{(0)}\right)\ge-\frac{w_1^{(0)}+w_2^{(0)}}{4}\exp\left(-\frac{w_1^{(0)}+w_2^{(0)}}{2}\right)\ge-\frac{w_1^{(0)}+w_2^{(0)}}{4}\exp\left(-\frac{1}{2}\right).$$

Plugging the above and Equation (64) in Equation (63) yields the lower bound

$$b_2^{(t_0)}-b_1^{(t_0)}\ge\frac{w_1^{(0)}+w_2^{(0)}}{4}\left(1-\exp\left(-\frac{1}{2}\right)\right)-0.1\ge\frac{w_1^{(0)}+w_2^{(0)}}{11}-\frac{1}{10}.$$

Finally, we use the monotonicity of $b_2^{(t)}-b_1^{(t)}$ which holds by our assumption on the step size η and Lemma E.3, to conclude the proof of the proposition for all $t\ge t_0$.

□

Proof of Theorem 5.2. The theorem follows in a straightforward manner, by utilizing Proposition E.4 to lower bound the numerator of $x^\star(t)$, using Lemma E.2 to upper bound the denominator, and plugging the bounds in the theorem statement to obtain explicit constants. □

F Additional Experiments

In this section, we extend the experiments conducted in Section 6 and provide empirical evidence that slow margin evolution persists beyond GD and beyond the simple, idealized architecture analyzed in our theory. In particular, we study a network with more than two neurons on a larger, higher-dimensional dataset and also explore adaptive optimization with Adam.

F.1 Wide Networks

In this experiment, we test whether the slow margin evolution of the two-neuron model also appears in a wider, higher-dimensional setting. The comparison is empirical and is not intended to identify an exact asymptotic rate for the wider network; rather, it visually compares the wide network's trajectory with that of the two-neuron model, for which our theory proves a logarithmic convergence rate under GD.

As mentioned in Section 6.2, we train a width-10 ReLU network on 20 points in $\mathbb{R}^5$. This is still far smaller than a practical architecture, but the scale is deliberate: it is a scale at which we can still enumerate every activation pattern and compute the network's Euclidean robustness margin exactly (up to numerical solver tolerance) within a tractable time frame, rather than resorting to attack-based approximations, while still probing whether increased width, input dimension, and sample size erode the bottleneck identified by our theory.

Collapsing the wide network into a two-neuron model. Our theoretical results are proved for a specific architecture and a two-point dataset. To examine whether the same behavior persists in the wide network, we first train the wide network for 10^6 iterations, retaining only runs whose loss is below $1/20$ at iteration 10^4. We then collapse the trained network into a two-neuron network on the synthetic points $x = 1$ and $x = -1$, initializing its four hidden-layer parameters w_1, b_1, w_2, b_2 according to the following four conditions. First, we give the neurons opposite orientations and a common breakpoint $x_0 \in (-1, 1)$, so that each neuron is active on only its corresponding synthetic point and the network has a unique zero at x_0. Because the two breakpoints initially coincide, this initialization is a boundary case of the regime specified by Assumption D.1. After the first balanced update, the first neuron's breakpoint moves to the left of the second neuron's breakpoint, and the network enters that regime. Second, we place x_0 so that its distance to the closer synthetic point exactly reproduces the wide network's margin, transferring the geometric quantity of interest to the reduced model. Third, we scale the neuron active on that side so that the output of the two-neuron network at the corresponding synthetic point matches the output of the wide network at its margin-defining training point. Under the exponential loss, this also matches the scalar loss derivative with respect to the network output at the two corresponding points. Fourth, we determine the scale of the remaining neuron by imposing the equilibrium condition

$$w_1 + w_2 + b_1 - b_2 = 0,$$

or, equivalently, $\Phi(1) = -\Phi(-1)$. Lemma D.7 shows that GD asymptotically approaches this balance, so the collapsed network is initialized directly on the balanced manifold within the regime governed by our $\Theta(1/\ln(t))$ result. We use the same parameter initialization for Adam, but initialize the collapsed network's iteration counter and first- and second-moment estimates at zero, while the wide network retains the optimizer state accumulated during its initial training. Although Lemma D.7 is specific to GD, the imposed balance is also preserved by Adam in this construction: the two synthetic points generate gradients of equal magnitude, and the zero-initialized moment estimates consequently produce matched adaptive updates. This justifies using the same collapse as an empirical comparator under Adam, but does not extend our GD convergence theorem to Adam; the slow margin evolution observed under Adam remains an empirical finding. With these initializations, we continue training both networks for a further 10^6 iterations under the same optimizer and learning rate, recomputing their margins at regular checkpoints. As shown in Figures 4 and 5 for GD and Adam, respectively, the two margin trajectories closely track on average. The wide network improves slightly faster than its collapsed counterpart, but their close agreement provides evidence that the same slow margin evolution persists in this wider, higher-dimensional setting.

Data. We consider a binary classification task in $\mathbb{R}^5$ defined by two disks, one per class. A point in the positive class ($y = +1$) has first coordinate fixed at $x_1 = 1$, while its remaining 4 coordinates

are drawn uniformly at random from the 4-dimensional ball of radius $\sqrt{5}$ centered at the origin. Symmetrically, a point in the negative class ($y = -1$) has first coordinate fixed at $x_1 = -1$, with the remaining $d - 1$ coordinates drawn uniformly from the same ball of radius $\sqrt{5}$. Thus each class occupies a 4-dimensional disk of radius $\sqrt{5}$ embedded in the hyperplane $x_1 = \pm 1$ of $\mathbb{R}^5$. We use $n = 20$ training points in total, split evenly as $n_+ = n_- = 10$, yielding a training set $\{(x_i, y_i)\}_{i=1}^{20}$ with $x_i \in \mathbb{R}^5$ and $y_i \in \{+1, -1\}$. Our data structure allows a simple linear predictor that thresholds at $x_1 = 0$ to obtain a robustness margin of 1, which aligns the reference robustness margin with our theoretical setting.

Network architecture and initialization. We train a single-hidden-layer ReLU network

$$\Phi(\boldsymbol{\theta}; \mathbf{x}) = \sum_{k=1}^{10} v_k \left[\mathbf{w}_k^\top \mathbf{x} + b_k\right]_+, \qquad \mathbf{w}_k \in \mathbb{R}^5, \quad b_k \in \mathbb{R}, \quad v_k \in \mathbb{R}.$$

The first-layer weights are initialized as $\mathbf{w}_k \sim \mathcal{N}(0, 0.4)$ entrywise (He-style scaling by the input dimension), the biases are initialized at zero, $b_k = 0$, and the output layer is initialized to a fixed balanced sign pattern, $(v_1, \ldots, v_{10}) = (\underbrace{+1, \ldots, +1}_{5}, \underbrace{-1, \ldots, -1}_{5})$. Throughout training, only $\mathbf{w}_k$ and b_k are updated; v_k are held fixed at their initial values.

Loss. We minimize the empirical exponential loss

$$L(\boldsymbol{\theta}) = \frac{1}{20} \sum_{i=1}^{20} \exp\big(-y_i \Phi(\boldsymbol{\theta}; \mathbf{x}_i)\big).$$

Full-batch training and epochs. Training is full-batch: every gradient iteration uses all 20 training points, with no mini-batching or data shuffling. Consequently, one *epoch* coincides exactly with one gradient-update iteration, and we use the terms interchangeably below.

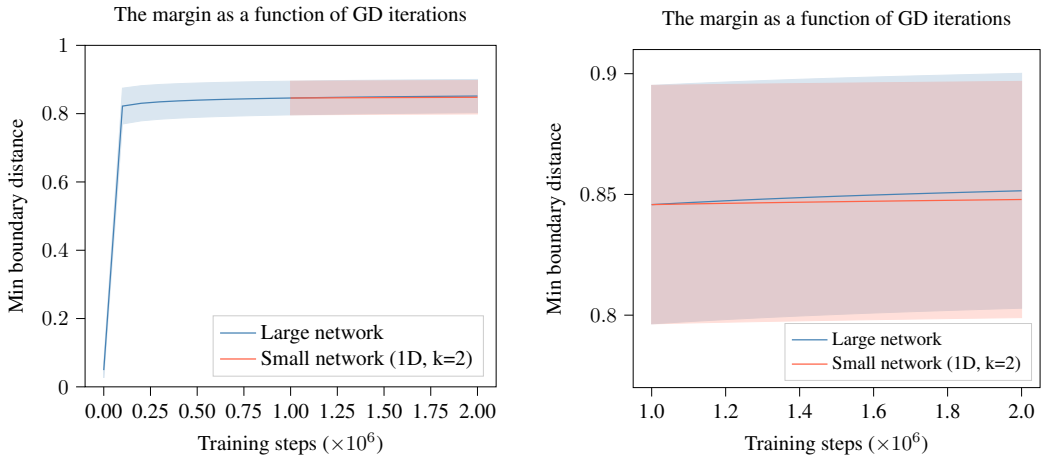
Optimizer and hyperparameters. For both full-batch GD and Adam, the figures report the learning rate $\eta = 0.1$. Additional runs for $\eta \in \{0.05, 0.5\}$ exhibited the same qualitative agreement between the wide and collapsed margin trajectories for each optimizer. For Adam, the momentum parameters are $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and the numerical-stability constant is $\varepsilon = 10^{-8}$. No weight decay (or any other explicit regularization) is applied. We conduct 200 independent trials for each optimizer, with independent data draws and initializations, and average the results over the successful trials (160 for GD and 200 for Adam).

Margin computation. Because Φ is piecewise affine, we compute its robustness margin exactly (up to numerical solver tolerance) by enumerating all 2^{10} activation patterns. For each $\boldsymbol{\sigma} \in \{0, 1\}^{10}$, let

$$R_{\boldsymbol{\sigma}} := \{\mathbf{x} \in \mathbb{R}^5 : (2\sigma_k - 1)(\mathbf{w}_k^\top \mathbf{x} + b_k) \geq 0 \text{ for every } k = 1, \ldots, 10\}$$

denote the corresponding closed polyhedral region. On $R_{\boldsymbol{\sigma}}$, the network agrees with the affine function $\Phi_{\boldsymbol{\sigma}}(\mathbf{x}) := \sum_{k:\sigma_k=1} v_k(\mathbf{w}_k^\top \mathbf{x} + b_k)$. These regions cover $\mathbb{R}^5$ and may overlap along activation boundaries, where the corresponding affine expressions agree. Consequently, the decision boundary decomposes exactly as $\{\mathbf{x} : \Phi(\boldsymbol{\theta}; \mathbf{x}) = 0\} = \bigcup_{\boldsymbol{\sigma} \in \{0,1\}^{10}} (H_{\boldsymbol{\sigma}} \cap R_{\boldsymbol{\sigma}})$, where $H_{\boldsymbol{\sigma}} := \{\mathbf{x} : \Phi_{\boldsymbol{\sigma}}(\mathbf{x}) = 0\}$. For each training point, we compute its distance to every nonempty boundary piece $H_{\boldsymbol{\sigma}} \cap R_{\boldsymbol{\sigma}}$ and take the minimum over all activation patterns. This distance is available in closed form whenever the orthogonal projection onto $H_{\boldsymbol{\sigma}}$ lies in $R_{\boldsymbol{\sigma}}$; otherwise, it is obtained by solving a small convex quadratic program that enforces the region constraints. This procedure gives the Euclidean distance from each training point to the decision boundary up to numerical solver tolerance. Finally, we minimize these distances separately over the positive- and negative-class examples to obtain the two class margins tracked throughout the experiment.

Experiment flow and results. After initializing the wide network as described above, we train it alone for one million full-batch updates, keeping the output layer $\mathbf{v}$ fixed and updating only the hidden weights and biases $\{\mathbf{w}_k, b_k\}_{k=1}^{10}$. We retain only runs whose loss is below the target threshold $1/n$ at iteration 10,000; this excludes 40 runs under GD and none under Adam. This



(a) The full process: the wide network's margin from initialization through both phases. The collapsed two-neuron network's margin appears when it is initialized at the end of the first phase and matches the wide network's margin at that point.

(b) A zoom-in on the second phase, once both networks train jointly. The iteration count retains the global training index and therefore begins at 10^6.

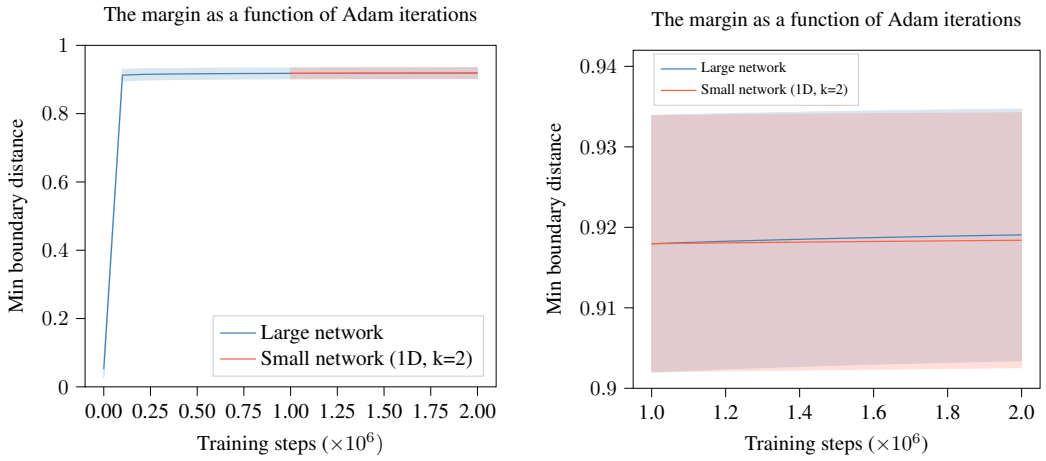
Figure 4: The margin as a function of training iterations for GD at $\eta = 0.1$. The two networks' average margin trajectories closely track, although the wider network improves slightly faster. The wider network has width 10 and input dimension 5. Results are averaged over the 160 runs that reached the small-loss regime using GD, with error bands representing half a standard deviation; the remaining 40 runs did not reach this regime and were excluded.

checkpoint guarantees correct classification of every training point and provides a common post-separation regime for comparing the runs. We then continue training the wide network from its current state while simultaneously training the freshly initialized two-neuron model under the same optimizer and learning rate for an additional one million iterations, recomputing both margins at regular checkpoints. Because the reduced model is initialized to match the geometric and loss-related quantities specified above, its subsequent trajectory provides a theory-amenable prediction of how the margin would evolve if its dynamics were governed primarily by the closest point in each class. Comparing this prediction with the wide network's actual trajectory tests whether the additional width, off-axis data variation, and optimizer history materially alter the margin dynamics. Under both GD and Adam, the two trajectories remain close throughout the comparison period, although the wide network improves slightly faster. Under GD, our theory guarantees that the margin gap of the two-neuron model decays as $\Theta(1/\ln t)$; under Adam, both networks exhibit similarly slow empirical margin evolution. Their close tracking suggests that the same finite-time bottleneck persists in the wide network under both optimizers, without identifying an asymptotic rate outside the two-neuron GD setting. The left panels of Figures 4 and 5 show the wide network's trajectory from initialization through both training phases, whereas the right panels isolate the post-collapse phase.

F.2 Adam optimizer in the two-neuron setting

In this experiment, we examine whether slow margin convergence persists under a widely used adaptive optimizer. We repeat the experimental setup described in Section 6, replacing GD with Adam and using the standard parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, numerical-stability constant $\varepsilon = 10^{-8}$, step size $\eta = 0.05$, and no weight decay.

The stopping rule used in the GD experiment is not theoretically justified for Adam. In particular, even if Adam drives $|w_1^{(t)} + w_2^{(t)} + b_1^{(t)} - b_2^{(t)}|$ below 0.001, we have no analogue of our GD analysis showing that this event marks entry into a regime with predictable subsequent margin dynamics. In the reported experiment, every run that passed the loss checkpoint completed the full budget of 10^7 iterations. We assess stability directly through the numerator $|b_2^{(t)} - b_1^{(t)}|$ in Equation (14). Across



(a) The full process: the wide network's margin from initialization through both phases. The collapsed two-neuron network's margin appears when it is initialized at the end of the first phase and matches the wide network's margin at that point.

(b) A zoom-in on the second phase, once both networks train jointly. The iteration count retains the global training index and therefore begins at 10^6.

Figure 5: The margin as a function of training iterations for Adam at $\eta = 0.1$. The two networks' average margin trajectories closely track, although the wider network improves slightly faster. The wider network has width 10 and input dimension 5. Results are averaged over all 200 independent runs, with error bands representing half a standard deviation.

all successful runs, the absolute difference between its terminal value and its value 10,000 iterations earlier was at most 4×10^{-7}, indicating that it had effectively stabilized by the end of training.

For GD, the quantity $\min\{|b_2^{(t)} - b_1^{(t)}|, \frac{1}{2}|w_1^{(0)} + w_2^{(0)}|\}$ is a finite-time lower bound on the numerator whose relevance follows from the monotonicity established in Lemma E.3. Because no corresponding result is available for Adam, the term $\frac{1}{2}|w_1^{(0)} + w_2^{(0)}|$ has no established finite-time predictive role in this experiment. We therefore report the directly observed value $|b_2^{(t)} - b_1^{(t)}|$ after verifying its numerical stability. Out of 1,000 independent initializations, 233 had training loss below 0.5 at iteration 10,000, and all 233 completed the full budget. As in the GD experiment, all 233 successful runs satisfied $w_2^{(0)} < 0 < w_1^{(0)}$, providing empirical evidence that the same sign condition governs successful specialization under Adam.

As shown in Figure 6, the terminal numerator distribution is broadly consistent with the density f_B in Equation (15), which was derived for GD, and exhibits a qualitatively similar shape to the corresponding GD distribution in Figure 2. In particular, 7.3% of the successful Adam runs end with a numerator of at least 1, consistent with the prediction $\Pr[B > 1] \approx 0.10069$ obtained from F_B. Thus, for a non-negligible fraction of successful initializations, attaining a margin gap of at most $\varepsilon = 0.01$ requires the denominator in Equation (14) to reach a magnitude of approximately 100. At the end of 10^7 iterations, the denominator remains below 35 in every successful run, so the runs with numerator at least 1 remain well above a margin gap of 0.01. Thus, Adam does not eliminate the slow finite-time margin evolution observed in this setting.

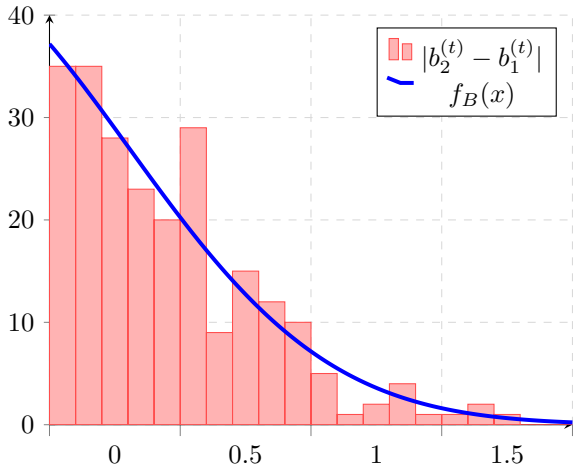


Figure 6: Empirical distribution of the numerator in Equation (14) under Adam. Among $1,000$ initializations, 233 had training loss below 0.5 at iteration $10{,}000$ and completed the full budget of 10^7 iterations. Because no Adam analogue of the GD monotonicity result in Lemma E.3 is available, we plot the directly observed terminal numerator $|b_2^{(t)} - b_1^{(t)}|$ rather than the GD-specific lower bound used in Figure 2. The numerator differed from its value $10{,}000$ iterations earlier by at most 4×10^{-7} across all successful runs, indicating that the reported values had effectively stabilized. The blue curve is the theoretical density of $\frac{1}{2}|w_1^{(0)} + w_2^{(0)}|$ derived for GD in Equation (15), scaled to match the histogram counts. The close agreement indicates that Adam produces a terminal numerator distribution similar to the GD prediction despite its adaptive step sizes.