THE INDUCTIVE BIAS OF RELU NETWORKS ON ORTHOGONALLY SEPARABLE DATA

Anonymous authors
Paper under double-blind review

In this paper, we study the inductive bias of two-layer ReLU networks trained by gradient flow. We find that on datasets that are simple ('orthogonally separable'), the ReLU network learns a function that is also simple, despite its ability to fit the training data in much more complex ways. The function learnt by the network can be characterised exactly, and we show (for binary classification) that it is a combination of two max-margin classifiers: one corresponding to the positive and one to the negative subset of the data.

1 INTRODUCTION

This paper is motivated by the problem of understanding the inductive bias of ReLU networks, or to put it plainly, understanding what it is that neural networks learn. This is a fundamental open question in neural network theory; it is also a crucial part of understanding how neural networks behave on previously unseen data (generalisation) and it could ultimately lead to rigorous a priori guarantees on neural nets' behaviour.

For a long time, the dominant way of thinking about machine learning systems was as minimisers of the empirical risk. However, this paradigm has turned out to be insufficient for understanding deep learning, where many empirical risk minimisers exist, often with vastly different generalisation properties. To understand deep networks, we therefore need a more fine-grained notion of 'what the model learns'.

This has motivated the study of the implicit bias of the training procedure – the ways in which the training algorithm influences which of the empirical risk minimisers is attained. The central thesis of this line of work is that the commonly used optimisation algorithms (gradient descent and its variants) bias learning towards 'simple' solutions, which tend to generalise well. Our work supports this thesis, by showing that when trained on 'simple' datasets, two-layer neural networks converge to 'simple' solutions, irrespective of the network's complexity (width).

Before describing our main result, we review the most relevant prior work. Most of it focused on characterising the inductive bias of linear classifiers trained by gradient descent: Soudry et al. (2018) considered the problem of learning a logistic regression classifier on linearly separable data, and show that the normalised weight vector converges to the max-margin direction. Ji & Telgarsky (2019a) considered deep linear networks trained on linearly separable data, and show that the normalised end-to-end weight vector converges to the max-margin direction. In fact, they show that all first-layer neurons converge to the same 'canonical neuron' (which points in the max-margin direction).

Our work can be seen as a continuation of this line of research into the nonlinear realm. We consider two-layer ReLU networks, and an appropriate stronger notion of linearly separable datasets. For such *orthogonally separable* datasets, we show that the network learns a function with just two canonical neurons, one corresponding to the positive data subset and one corresponding to the negative data subset. In other words, there are only two groups of neurons, within which all neurons behave asymptotically the same. We show that the direction of each canonical neuron is the max-margin direction for the respective data subset.

To understand how this simple solution comes about, it is helpful to think of training as proceeding in two phases: a very early phase, in which neurons are still close to the initialisation (which we assume to be small), and a later phase, in which neurons move away.

The clustering of neurons into two groups happens in the first phase. This 'specialisation' occurs because near the origin, neurons behave as if each was locally minimising (or maximising) the same function G independently of other neurons, so they all converge to the function's local minima

(or maxima). The simplicity of the data ensures that this function has a single minimum and a single maximum. Clustering thus arises as a consequence of independent optimisation of the same objective.

As training proceeds to the second phase, the neurons' directions are roughly preserved, even as inter-neuron dependence kicks in. Importantly, the neurons' activation patterns become fixed and equal for neurons in the same group. The network thus becomes an ensemble of two linear subnetworks, each of which converges to its respective max-margin classifier.

2 FORMAL SETTING

In this section, we introduce the learning scenario including the assumptions we make about the dataset, the model, and the training procedure. We consider binary classification. Denote the training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$ for all $i \in [n]$. We denote by $\mathbf{X} \in \mathbb{R}^{d \times n}$ the matrix with $\{\mathbf{x}_i\}$ as columns and by $\mathbf{y} \in \mathbb{R}^n$ the vector with $\{y_i\}$ as entries.

Orthogonally separable data. A binary classification dataset $(\mathbf{X}, \mathbf{y})$ is called *orthogonally separable* if for all $i, j \in [n]$,

$$\begin{aligned} \mathbf{x}_i^\intercal \mathbf{x}_j &> 0, \qquad && \text{if } y_i = y_j, \\ \mathbf{x}_i^\intercal \mathbf{x}_j &\leq 0, \qquad && \text{if } y_i \neq y_j. \end{aligned} \tag{1}$$

In other words, a dataset is orthogonally separable if and only if it is linearly separable, and any training example $\mathbf{x}_i$ can serve as a linear separator. Geometrically, this means that examples with $y_i = 1$ ('positive examples') and examples with $y_i = -1$ ('negative examples') lie in opposite orthants.

Two-layer ReLU networks. We define two-layer width-p fully-connected ReLU networks, parameterised by $\boldsymbol{\theta} \triangleq \{\mathbf{W}, \mathbf{a}\}$, as

$$\begin{aligned} f_{\boldsymbol{\theta}} &: \mathbb{R}^d \to \mathbb{R}, \\ f_{\boldsymbol{\theta}}(\mathbf{x}) &\triangleq \mathbf{a}^\intercal \rho(\mathbf{W}\mathbf{x}), \end{aligned} \tag{2}$$

where $\mathbf{W} \triangleq [\mathbf{w}_1, \dots \mathbf{w}_p]^\intercal \in \mathbb{R}^{p \times d}$ and $\mathbf{a} \triangleq [a_1, \dots, a_p]^\intercal \in \mathbb{R}^p$ are the first- and second-layer weights of the network, and ρ is the element-wise ReLU function, $\rho(\mathbf{z})_i = \max\{0, z_i\}$. We will often view the network as a collection of neurons, $\{(a_j, \mathbf{w}_j)\}_{j=1}^p$.

Cross-entropy loss. We assume a training loss of the form

$$\ell(\boldsymbol{\theta}) \triangleq \sum_{i=1}^n \ell_i(f_{\boldsymbol{\theta}}(\mathbf{x}_i)), \qquad \ell_i(u) \triangleq \log(1 + \exp(-y_i u)); \tag{3}$$

this is the standard empirical cross-entropy loss. More generally, our results hold when the loss is differentiable, ℓ_i' is bounded and Lipschitz continuous, and satisfies $-y_i \ell_i'(u) > 0$ for all $u \in \mathbb{R}$.

Gradient flow training. We assume the loss is optimised by gradient descent with infinitesimally small step size, also known as gradient flow. Under the gradient flow dynamics, the parameter trajectory is an absolutely continuous curve $\{\boldsymbol{\theta}(t) \mid t \geq 0\}$ satisfying the differential inclusion

$$\frac{\partial \boldsymbol{\theta}(t)}{\partial t} \in -\partial \ell(\boldsymbol{\theta}(t)), \qquad \text{for almost all } t \in [0, \infty), \tag{4}$$

where $\partial \ell$ denotes the Clarke subdifferential Clarke (1975); Clarke et al. (2008) of ℓ, an extension of the gradient to not-everywhere differentiable functions,

$$\partial \ell(\boldsymbol{\theta}) \triangleq \operatorname{conv}\left\{ \lim_{k \to \infty} \nabla \ell(\boldsymbol{\theta}_k) \,\middle|\, \boldsymbol{\theta}_k \to \boldsymbol{\theta} \right\}. \tag{5}$$

$\boldsymbol{\theta}(t)$ is the value of the parameters at time t, and we will use the suffix (t) more generally to denote the value of some function of $\boldsymbol{\theta}$ at time t.

Near-zero balanced initialisation. We assume that at the beginning of training, the weights satisfy $\mathbf{w}_j = \lambda \mathbf{v}_j$ and $a_j \in \{\pm\lambda\|\mathbf{v}_j\|\}$ for some $\mathbf{v}_1, \ldots, \mathbf{v}_p \in \mathbb{R}^d$ and small $\lambda > 0$. This technical condition ensures that the neurons are balanced and small in size, $\|\mathbf{w}_j\| = |a_j| \approx 0$, which simplifies the calculations involved in the analysis of gradient flow.

3 Main result

Under the assumptions of Section 2 (and one additional assumption), the network converges to a combination of two max-margin neurons. Specifically, given a dataset $(\mathbf{X}, \mathbf{y})$, define the *positive and the negative max-margin vectors* $\mathbf{w}_+, \mathbf{w}_- \in \mathbb{R}^d$ as

$$\mathbf{w}_+ = \arg\min_{\mathbf{w}} \|\mathbf{w}\|^2 \qquad \text{subject to} \qquad \mathbf{w}^\intercal \mathbf{x}_i \geq 1 \;\text{ for }\; i : y_i = 1, \tag{6}$$

$$\mathbf{w}_- = \arg\min_{\mathbf{w}} \|\mathbf{w}\|^2 \qquad \text{subject to} \qquad \mathbf{w}^\intercal \mathbf{x}_i \geq 1 \;\text{ for }\; i : y_i = -1. \tag{7}$$

We call examples which attain equality in eqs. (6) and (7) *positive support examples* and *negative support examples* respectively. We now state the main result.

Theorem 1. *Consider an orthogonally separable dataset $(\mathbf{X}, \mathbf{y})$ such that its positive support examples span $\mathbb{R}^d$, and its negative support examples also span $\mathbb{R}^d$. For almost all such datasets and almost all balanced initialisations $(\mathbf{v}_1, \ldots, \mathbf{v}_p)$ with λ small enough, a two-layer ReLU network $f_{\boldsymbol{\theta}}$ trained by gradient flow with the cross-entropy loss satisfies*

$$\left\| \frac{\mathbf{W}(t)}{\|\mathbf{W}(t)\|_F} - \left(\mathbf{u}\mathbf{w}_+^\intercal + \mathbf{z}\mathbf{w}_-^\intercal\right) \right\|_F \to 0, \qquad \text{as } t \to \infty, \tag{8}$$

for some $\mathbf{u}, \mathbf{z} \in \mathbb{R}_+^p$ such that either $u_i = 0$ or $z_i = 0$ for all $i \in [p]$.

The theorem says that each neuron (row of $\mathbf{W}$), properly normalised, converges either to a scalar multiple of the positive max-margin direction, $u_i \mathbf{w}_+$, or to a scalar multiple of the negative max-margin direction, $z_i \mathbf{w}_-$. In other words, there are asymptotically only two distinct types of neurons, and the network could in principle be pruned down to a width of just two. These two 'canonical neurons' moreover have an explicit characterisation, given by eqs. (6) and (7).

The following corollary expresses the above in terms of the function learnt by the network.

Corollary 1. *Under the conditions of Theorem 1, there exist constants $u, z \geq 0$ such that*

$$\frac{f_{\boldsymbol{\theta}(t)}(\mathbf{x})}{\|\boldsymbol{\theta}(t)\|^2} \to u\rho(\mathbf{w}_+^\intercal \mathbf{x}) - z\rho(\mathbf{w}_-^\intercal \mathbf{x}), \qquad \text{as } t \to \infty. \tag{9}$$

3.1 Discussion of assumptions

Many of our assumptions are technical, serving to simplify the analysis while detracting little from the result's relevance. We assume the training dynamics is driven by gradient flow rather than the discrete gradient descent, which typically requires a smooth objective to be analysed easily. The balancedness condition on the initialisation only serves to simplify calculations; an approximate version of it holds near the origin. See [CITE!!!] for the definition of approximate balancedness and its application in the context of training deep linear nets. The assumption that support vectors span $\mathbb{R}^d$ comes from Ji & Telgarsky (2019a). It seems to us that it could be lifted, though we have not investigated this possibility in depth.

Two assumptions that deserve more attention are near-zero initialisation and orthogonal separability; both are crucial for the result to hold. Near-zero initialisation grants neurons high directional mobility early in training, allowing them to cluster close to the canonical directions. Orthogonal separability ensures that the canonical directions are 'easy to find' by local descent. In prior work, which considered linear networks Soudry et al. (2018); Ji & Telgarsky (2019a), this role is fulfilled by linear separability. The reason we need a stronger condition is that ReLU updates are more local compared to linear updates: a linear neuron takes into account all examples in the training set, whereas a ReLU neuron updates only on examples in its positive half-plane (its active examples). It is therefore harder to ensure that updates point in the right direction.

4 PROOF SKETCH

In the analysis, we distinguish between two phases of training. The first phase takes place close to the origin, $\|\boldsymbol{\theta}\| \approx 0$. In this phase, while neurons move little in the absolute sense, they converge in direction to certain regions of the weight space called *extremal sectors*.

4.1 CONVERGENCE TO EXTREMAL SECTORS

(All definitions and results in this subsection are by Maennel et al. (2018). We need them later on.)

Sectors are regions in weight space corresponding to different activation patterns. They are important for understanding neuron dynamics: roughly speaking, neurons in the same sector move in the same direction.

Definition 1 (Sector)**.** *The* sector *associated to a sequence of signs $\boldsymbol{\sigma} \in \{-1, 0, 1\}^n$ is the region in input space defined as*

$$\mathcal{S}_{\boldsymbol{\sigma}} \triangleq \{\mathbf{w} \in \mathbb{R}^d \mid \operatorname{sign} \mathbf{w}^\intercal \mathbf{x}_i = \sigma_i,\ i \in [n]\}. \tag{10}$$

We may also refer to the sign sequence $\boldsymbol{\sigma}$ itself as a sector.

Extremal sectors are sectors where neurons tend to cluster early in training. To characterise them formally, we first need to introduce the function $G : \mathbb{S}^{d-1} \to \mathbb{R}$,

$$G(\mathbf{w}) \triangleq -\sum_{i=1}^{n} \ell_i'(0) \cdot \rho(\mathbf{w}^\intercal \mathbf{x}_i). \tag{11}$$

Definition 2 (Extremal directions and sectors)**.** *We say that $\mathbf{w} \in \mathbb{S}^{d-1}$ is a* positive extremal direction*, if it is a strict local maximum of G. We say that $\mathbf{w}$ is a* negative extremal direction *if it is a strict local minimum of G. A sector is called (positive/negative) extremal, if it contains a (positive/negative) extremal direction.*

The following lemma (Maennel et al., 2018, Lemma 5) shows that all neurons either turn off, i.e. become deactivated for all training examples and stop updating, or converge to extremal sectors.

Lemma 1. *Let a two-layer ReLU network $f_{\boldsymbol{\theta}}$ be balanced at initialisation and trained by gradient flow. Assume that the loss derivative ℓ_i' is Lipschitz continuous. Then, for almost all datasets and almost all initialisations with λ small enough, there exists a time T such that each neuron satisfies one of these three conditions:*

- $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$ *where $\boldsymbol{\sigma} \leq \mathbf{0}$ and so $\mathbf{w}_j$ remains constant for $t \geq T$, or*
- $a_j(T) > 0$ *and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$ where $\boldsymbol{\sigma}$ is a positive extremal sector, or*
- $a_j(T) < 0$ *and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$ where $\boldsymbol{\sigma}$ is a negative extremal sector.*

4.2 ORTHOGONAL SEPARABILITY: TWO ABSORBING EXTREMAL SECTORS

Lemma 1 shows that by the end of the early phase of training, neurons converge to extremal sectors. Although eq. (11) shows that the number of extremal sectors depends only on the data (i.e. is independent of model expressivity), it is a priori unclear how many extremal sectors there are for a given dataset, or what happens once neurons have converged to extremal sectors. We now answer both of these questions for orthogonally separable datasets.

First, we claim that for orthogonally separable datasets, there are only two extremal sectors, one corresponding to the positive data subset and one corresponding to the negative data subset. That is, by converging to an extremal sector, neurons 'choose' whether to activate for positive examples or for negative examples. They thus naturally form two groups of similar neurons.

Lemma 2. *In the setting of Theorem 1, there is exactly one positive extremal direction and exactly one negative extremal direction. The positive extremal sector $\boldsymbol{\sigma}^+$ is given by*

$$\sigma_j^+ = \begin{cases} 1, & \text{if } y_j = 1, \\ -1, & \text{if } y_j = -1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i < 0 \text{ for some } i \text{ with } y_i = 1, \\ 0, & \text{if } y_j = -1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i = 0 \text{ for all } i \text{ with } y_i = 1, \end{cases} \tag{12}$$

and the negative extremal sector $\boldsymbol{\sigma}^-$ is given by

$$\sigma_j^- = \begin{cases} 1, & \text{if } y_j = -1, \\ -1, & \text{if } y_j = 1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i < 0 \text{ for some } i \text{ with } y_i = -1, \\ 0, & \text{if } y_j = 1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i = 0 \text{ for all } i \text{ with } y_i = -1. \end{cases} \tag{13}$$

Second, we show that once a neuron reaches an extremal sector, it remains in the sector forever, i.e. its activation pattern remains fixed for the rest of training.

Lemma 3. *Assume the setting of Theorem 1. If at time T the neuron $(a_j, \mathbf{w}_j)$ satisfies $a_j(T) > 0$ and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$, where $\boldsymbol{\sigma}$ is the positive extremal sector (eq.* (12)*), then for $t \geq T$, $\mathbf{w}_j(t) \in \mathcal{S}_{\boldsymbol{\sigma}}$. The same holds if $a_j(T) < 0$ and $\boldsymbol{\sigma}$ is the negative extremal sector (eq.* (13)*).*

4.3 Proof of Theorem 1

Once neurons enter their respective absorbing sectors, the second phase of training begins. In this phase, the network's activation patterns are fixed: half of the neurons are active and update on the positive examples, while the other half are active and update on the negative examples. The network thus behaves like an ensemble of independent linear subnetworks trained on subsets of the data. Once this happens, it becomes possible to apply existing results for linear networks; in particular, each subnetwork converges to its respective max-margin classifier.

We give more details in the proof below.

Theorem 1. *Consider an orthogonally separable dataset $(\mathbf{X}, \mathbf{y})$ such that its positive support examples span $\mathbb{R}^d$, and its negative support examples also span $\mathbb{R}^d$. For almost all such datasets and almost all balanced initialisations $(\mathbf{v}_1, \ldots, \mathbf{v}_p)$ with λ small enough, a two-layer ReLU network $f_{\boldsymbol{\theta}}$ trained by gradient flow with the cross-entropy loss satisfies*

$$\left\| \frac{\mathbf{W}(t)}{\|\mathbf{W}(t)\|_F} - \left(\mathbf{u}\mathbf{w}_+^\intercal + \mathbf{z}\mathbf{w}_-^\intercal\right) \right\|_F \to 0, \qquad \text{as } t \to \infty, \tag{8}$$

for some $\mathbf{u}, \mathbf{z} \in \mathbb{R}_+^p$ such that either $u_i = 0$ or $z_i = 0$ for all $i \in [p]$.

Proof of Theorem 1. By Lemmas 1 and 2, there exists a time T such that each neuron satisfies either

- $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$ where $\boldsymbol{\sigma} \leq \mathbf{0}$ and $\mathbf{w}_j$ remains constant for $t \geq T$, or
- $a_j(T) > 0$ and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}^+}$, or
- $a_j(T) < 0$ and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}^-}$,

where $\boldsymbol{\sigma}^+, \boldsymbol{\sigma}^-$ are the unique positive and negative extremal sectors given by eqs. (12) and (13). Denote by $\mathcal{J}_0, \mathcal{J}_+, \mathcal{J}_-$, the sets of neurons satisfying the first, the second, and the third condition respectively. By Lemma 3, if $j \in \mathcal{J}_+$ then $\mathbf{w}_j(t) \in \mathcal{S}_{\boldsymbol{\sigma}^+}$ for all $t \geq T$, and if $j \in \mathcal{J}_-$ then $\mathbf{w}_j(t) \in \mathcal{S}_{\boldsymbol{\sigma}^-}$ for $t \geq T$. Hence, for $t \geq T$, if $\mathbf{x}_i$ is such that $y_i = 1$ then

$$f_{\boldsymbol{\theta}}(\mathbf{x}_i) \triangleq \sum_{j \in [p]} a_j \rho(\mathbf{w}_j^\intercal \mathbf{x}_i) = \sum_{j \in \mathcal{J}_+} a_j \mathbf{w}_j^\intercal \mathbf{x}_i. \tag{14}$$

Combined with Lemma A.3, this implies that for $k \in \mathcal{J}_+$,

$$\begin{aligned} \frac{\partial a_k}{\partial t} &= -\sum_{i: y_i = 1} \ell_i'\Big(\sum_{j \in \mathcal{J}_+} a_j \mathbf{w}_j^\intercal \mathbf{x}_i \Big) \cdot \mathbf{w}_k^\intercal \mathbf{x}_i, \\ \frac{\partial \mathbf{w}_k}{\partial t} &= -\sum_{i: y_i = 1} \ell_i'\Big(\sum_{j \in \mathcal{J}_+} a_j \mathbf{w}_j^\intercal \mathbf{x}_i \Big) \cdot a_k \mathbf{x}_i, \end{aligned} \tag{15}$$

where we have used that $\mathbf{P}_{\mathbf{w}} \mathbf{x}_i = \mathbf{x}_i$ for i with $y_i = 1$ due to positive extremality. From eq. (15) it follows that the evolution of neurons in $\mathcal{J}_+$ depends only on positive examples and other neurons in $\mathcal{J}_+$. Moreover, the neurons behave linearly on the positive data subset, while ignoring the negative subset. The same argument shows that the evolution of neurons in $\mathcal{J}_-$ depends only on other neurons

in $\mathcal{J}_-$ and the negative data subset, on which the neurons act linearly. In other words, from time T onwards the ReLU network decomposes into a constant part and two independent linear networks, one trained on the positive data subset and the other trained on the negative data subset.

We can therefore apply existing max-margin convergence results for linear networks to each of the linear subnetworks. Denote by $\mathbf{W}^\intercal = [\mathbf{W}_0^\intercal, \mathbf{W}_+^\intercal, \mathbf{W}_-^\intercal]$ the three parts of the weight matrix. Then by (Ji & Telgarsky, 2019a, Theorems 2.2 and 2.8) and (Ji & Telgarsky, 2020, Theorem 3.1), there exist vectors $\bar{\mathbf{u}}, \bar{\mathbf{z}}$, such that

$$\left\| \frac{\mathbf{W}_+(t)}{\|\mathbf{W}_+(t)\|_F} - \bar{\mathbf{u}}\mathbf{w}_+^\intercal \right\|_F \to 0, \qquad \text{as } t \to \infty, \tag{16}$$

$$\left\| \frac{\mathbf{W}_-(t)}{\|\mathbf{W}_-(t)\|_F} - \bar{\mathbf{z}}\mathbf{w}_-^\intercal \right\|_F \to 0, \qquad \text{as } t \to \infty. \tag{17}$$

We now need to relate $\|\mathbf{W}_+\|_F$ and $\|\mathbf{W}_-\|_F$ to $\|\mathbf{W}\|_F$. In particular, it will be useful to show that $\|\mathbf{W}_+(t)\|_F^2 / \log t$ has a limit as $t \to \infty$; the same is true for $\|\mathbf{W}_-(t)\|_F^2 / \log t$ (by the same argument). We consider the learning of the positive linear subnetwork, whose objective is effectively $\ell^+(\boldsymbol{\theta}) := \sum_{i:y_i=1} \ell_i(f_{\boldsymbol{\theta}}(\mathbf{x}_i))$. By (Ji & Telgarsky, 2019a, Theorem 2.2), we know that $\ell^+(t) \to 0$ as $t \to \infty$. Following (Lyu & Li, 2020, Definition A.3), define

$$\tilde{\gamma}(\boldsymbol{\theta}) \triangleq \frac{g(\log 1/\ell^+(\boldsymbol{\theta}))}{2\|\mathbf{W}_+\|_F^2}, \tag{18}$$

where $g(q) := -\log\left(\exp(\exp(-q)) - 1\right)$ for the cross-entropy loss. Then

$$\frac{\|\mathbf{W}_+(t)\|_F^2}{\log t} = \frac{g(\log 1/\ell^+(t))}{2\tilde{\gamma}(t)\log t} = \frac{-\log\left(\exp(\ell^+(t)) - 1\right)}{2\tilde{\gamma}(t)\log t}. \tag{19}$$

Using the Taylor expansion $\exp(u) = 1 + \Theta(u)$ for $u \to 0$ and (Lyu & Li, 2020, Corollary A.11), we obtain

$$\frac{\|\mathbf{W}_+(t)\|_F^2}{\log t} = \frac{-\log \Theta(\ell^+(t))}{2\tilde{\gamma}(t)\log t} = \frac{\log \Theta(t \log t)}{2\tilde{\gamma}(t)\log t} = \frac{1}{2\tilde{\gamma}(t)}\left(\frac{\Theta(1) + \log\log t}{\log t} + 1\right). \tag{20}$$

By (Lyu & Li, 2020, Theorem A.7:1), $\tilde{\gamma}$ is increasing in t and hence converges; it follows that $\|\mathbf{W}_+(t)\|_F^2 / \log t$ has a limit. By (Lyu & Li, 2020, Corollary A.11), $\|\mathbf{W}_+(t)\|_F^2 = \Theta(\log t)$, implying that the limit is finite and strictly positive. We will denote it by ν_+ and the analogous quantity for $\mathbf{W}_-$ by ν_-.

We now return to the main thread of the proof. We analyse the convergence of $\mathbf{W}(t)/\|\mathbf{W}(t)\|_F$ by analysing $\mathbf{W}_0/\|\mathbf{W}(t)\|_F$, $\mathbf{W}_+(t)/\|\mathbf{W}(t)\|_F$ and $\mathbf{W}_-(t)/\|\mathbf{W}(t)\|_F$ in turn. Since $\|\mathbf{W}(t)\|_F^2 = \|\mathbf{W}_0\|_F^2 + \|\mathbf{W}_+(t)\|_F^2 + \|\mathbf{W}_-(t)\|_F^2$,

$$\lim_{t\to\infty} \frac{\|\mathbf{W}(t)\|_F^2}{\log t} = \nu_+ + \nu_-. \tag{21}$$

It follows that

$$\frac{\mathbf{W}_0}{\|\mathbf{W}(t)\|_F} \to \mathbf{0}, \qquad \text{as } t \to \infty. \tag{22}$$

Also, by eqs. (16) and (21),

$$\frac{\mathbf{W}_+(t)}{\|\mathbf{W}(t)\|_F} = \frac{\mathbf{W}_+(t)}{\|\mathbf{W}_+(t)\|_F} \cdot \frac{\|\mathbf{W}_+(t)\|_F / \sqrt{\log t}}{\|\mathbf{W}(t)\|_F / \sqrt{\log t}} \to \frac{\sqrt{\nu_+}}{\sqrt{\nu_+ + \nu_-}}\, \bar{\mathbf{u}}\mathbf{w}_+^\intercal, \tag{23}$$

and similarly

$$\frac{\mathbf{W}_+(t)}{\|\mathbf{W}(t)\|_F} \to \frac{\sqrt{\nu_-}}{\sqrt{\nu_+ + \nu_-}}\, \bar{\mathbf{z}}\mathbf{w}_-^\intercal. \tag{24}$$

Combining the last three equations yields the result. □

5 RELATED WORK

There is a lot of prior work characterising the implicit bias of gradient descent for various linear models. For a logistic regression classifier, Soudry et al. (2018) show that assuming an exponentially-tailed loss and linearly separable data, the normalised weight vector converges to the max-margin direction. Ji & Telgarsky (2019b) extend this result to non-separable data, Nacson et al. (2019) extend it to super-polynomially-tailed losses, and Gunasekar et al. (2018a) considers different optimisation algorithms. For deep linear networks, Ji & Telgarsky (2019a) show that the end-to-end weight matrix converges to the max-margin solution and consecutive weight matrices align. Gunasekar et al. (2018b) consider linear convolutional networks and prove convergence to a predictor related to the $\ell_{2/L}$ bridge penalty in the frequency domain.

A few papers have started addressing the implicit bias problem for nonlinear (homogeneous or ReLU) networks. The problem is much harder and hence requires stronger assumptions. Lyu & Li (2020) and Ji & Telgarsky (2020) assume that at some point during training, the network attains perfect classification accuracy. Training from this point onward, Ji & Telgarsky (2020) show that the network parameters converge in direction. Lyu & Li (2020) show that this direction is a critical KKT point of the (nonlinear) max-margin problem. A complementary approach is taken by Maennel et al. (2018) who analyse the very early phase of training, when the weights are close to the origin. For two-layer networks, they show convergence of neurons to extremal sectors. Our work can be seen as a first step towards bridging the very early and the very late phase of training.

Zooming out a bit, there is also work motivated by similar questions, but taking a different approach. For example, Li & Liang (2018) show that two-layer ReLU nets trained on structured data converge to a solution that generalises well. Like ours, their analysis requires that the network's activation patterns change little, but they achieve it by containing training in the neighbourhood of the (relatively large) initialisation (this is the standard lazy training argument Chizat et al. (2019)). In contrast, we initialise much closer to zero, allowing the neurons to move more. Another related paper is Chizat & Bach (2020). Using a mean-field analysis, the authors show that infinite-width two-layer ReLU nets converge to max-margin classifiers in a certain non-Hilbertian function space.

6 CONCLUSION & FUTURE WORK

In this work, we prove that two-layer ReLU nets trained by gradient flow on orthogonally separable data converge to a combination of the positive and the negative max-margin classifier. This shows that when trained on simple datasets, neural networks learn simple functions, despite their ability to fit much more complex ones. We consider this a first step towards characterising the inductive bias of neural network training with ReLU nonlinearities.

The proof distinguishes between two phases of learning: an early phase, in which neurons specialise direction-wise though hardly move, and a late phase, in which the network's activation pattern is fixed and hence the network decomposes into an ensemble of linear subnetworks. Our hope is that a similar strategy could be applied to prove results for deeper networks and more complicated datasets as well.

ACKNOWLEDGMENTS

We thank Hartmut Maennel for help with understanding his work on the early phase of training.

REFERENCES

Lénaïc Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on Computational Learning Theory (COLT)*, 2020.

Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in supervised differentiable programming. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

Frank H Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

Frank H Clarke, Yuri S Ledyaev, Ronald J Stern, and Peter R Wolenski. *Nonsmooth analysis and control theory*, volume 178. Springer Science & Business Media, 2008.

Damek Davis, Dmitriy Drusvyatskiy, Sham Kakade, and Jason D Lee. Stochastic subgradient method converges on tame functions. *Foundations of computational mathematics*, 20(1):119–154, 2020.

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. 2018a.

Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018b.

Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. In *International Conference on Learning Representations (ICLR)*, 2019a.

Ziwei Ji and Matus Telgarsky. The implicit bias of gradient descent on nonseparable data. In *Conference on Computational Learning Theory (COLT)*, 2019b.

Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. In *arXiv:2006.06657*, 2020.

Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

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

Mor Shpigel Nacson, Jason Lee, Suriya Gunasekar, Pedro Henrique Pamplona Savarese, Nathan Srebro, and Daniel Soudry. Convergence of gradient descent on separable data. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2019.

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 (JMLR)*, 19(1):2822–2878, 2018.

A BASIC LEMMAS

This section collects a few lemmas useful for proofs. We assume the same setting and notation as Sections 2 and 4. In addition, we denote by $\mathbf{P}_{\mathbf{w}}$ the orthogonal projection onto $\operatorname{span}\{\mathbf{x}_i |\, \mathbf{w}^\intercal \mathbf{x}_i = 0\}^\perp$, and by $g : \mathbb{R}^d \to \mathbb{R}^d$,

$$g(\mathbf{w}) \triangleq -\sum_{i=1}^{n} \ell_i'(0) \cdot \mathbb{1}\{\mathbf{w}^\intercal \mathbf{x}_i > 0\}\, \mathbf{P}_{\mathbf{w}} \mathbf{x}_i. \tag{25}$$

A.1 LEMMAS ABOUT SECTORS

The following lemma gives a necessary condition for a vector to be an extremal direction.

Lemma A.1. *If $\mathbf{w} \in \mathbb{S}^{d-1}$ is an extremal direction, then $g(\mathbf{w}) = C\mathbf{w}$ for some constant C.*

Proof. Let $\hat{\mathbf{w}} \in \mathbb{S}^{d-1}$ be a positive extremal direction (the negative case is analogous), and let $\hat{\mathbf{w}} \in \mathcal{S}_{\hat{\boldsymbol{\sigma}}}$. A sector is called *open* if $\sigma_i \neq 0$ for all $i \in [n]$. Denote by $\mathcal{A}(\hat{\boldsymbol{\sigma}})$ the set of all open sectors adjacent to $\hat{\boldsymbol{\sigma}}$,

$$\mathcal{A}(\hat{\boldsymbol{\sigma}}) := \left\{ \boldsymbol{\sigma} \in \{\pm 1\}^n \,\middle|\, \max_{i \in [n]} |\sigma_i - \hat{\sigma}_i| \leq 1 \right\}. \tag{26}$$

Since $\hat{\mathbf{w}}$ is a local maximum of G and G is sector-wise linear, $\hat{\mathbf{w}}$ maximises G when constrained to (the closure of) any adjacent sector, i.e. for any $\boldsymbol{\sigma} \in \mathcal{A}(\hat{\boldsymbol{\sigma}})$,

$$\begin{aligned} \hat{\mathbf{w}} = \arg\max_{\mathbf{w}} G(\mathbf{w}), \text{ subject to } \|\mathbf{w}\|^2 = 1, \\ \sigma_i \mathbf{w}^\intercal \mathbf{x}_i \geq 0 \text{ for all } i \in [n]. \end{aligned} \tag{27}$$

For $\mathbf{w}$ in the feasible region, G can be treated as a linear function with $\nabla G(\mathbf{w}) = g(\mathbf{w}_{\boldsymbol{\sigma}})$ where $\mathbf{w}_{\boldsymbol{\sigma}}$ is any vector such that $\mathbf{w}_{\boldsymbol{\sigma}} \in \mathcal{S}_{\boldsymbol{\sigma}}$. Hence, the necessary first-order KKT conditions for the problem (27) are

$$g(\mathbf{w}_{\boldsymbol{\sigma}}) = C\hat{\mathbf{w}} - \sum_{i=1}^{n} \lambda_i \sigma_i \mathbf{x}_i, \tag{28}$$

where $\lambda_i \geq 0$ for all i, but $\lambda_i \neq 0$ requires that the corresponding constraint is tight, $\sigma_i \hat{\mathbf{w}}^\intercal \mathbf{x}_i = 0$. It follows that $\mathbf{P}_{\hat{\mathbf{w}}} \lambda_i \sigma_i \mathbf{x}_i = \mathbf{0}$. Multiplying eq. (28) from the left by $\mathbf{P}_{\hat{\mathbf{w}}}$ therefore yields

$$C\hat{\mathbf{w}} = \mathbf{P}_{\hat{\mathbf{w}}}\, g(\mathbf{w}_{\boldsymbol{\sigma}}) = -\sum_{i=1}^{n} \ell_i'(0) \cdot \mathbb{1}\{\sigma_i = 1\}\, \mathbf{P}_{\hat{\mathbf{w}}} \mathbf{x}_i. \tag{29}$$

By adjacency, $\sigma_i = \hat{\sigma}_i$ whenever $\hat{\sigma}_i \in \{\pm 1\}$, so they can differ only when $\hat{\sigma}_i = 0$, i.e. when $\mathbf{P}_{\hat{\mathbf{w}}} \mathbf{x}_i = \mathbf{0}$. It follows that

$$C\hat{\mathbf{w}} = -\sum_{i=1}^{n} \ell_i'(0) \cdot \mathbb{1}\{\hat{\sigma}_i = 1\}\, \mathbf{P}_{\hat{\mathbf{w}}} \mathbf{x}_i = g(\hat{\mathbf{w}}). \tag{30}$$

□

The following lemma describes the local behaviour of the function G (defined in eq. (11)).

Lemma A.2. *For $\mathbf{w} \in \mathbb{S}^{d-1}$ and $\mathbf{v} \in \mathbb{R}^d$, there exists $\epsilon_{\max} > 0$ such that for $\epsilon \in [0, \epsilon_{\max}]$,*

$$G\left(\frac{\mathbf{w} + \epsilon \mathbf{v}}{\|\mathbf{w} + \epsilon \mathbf{v}\|} \right) = \frac{1}{\|\mathbf{w} + \epsilon \mathbf{v}\|} \left(G(\mathbf{w}) - \epsilon \sum_{i=1}^{n} \ell_i'(0)\, \mathbb{1}\{(\mathbf{w} + \epsilon \mathbf{v})^\intercal \mathbf{x}_i > 0\}\, \mathbf{v}^\intercal \mathbf{x}_i \right). \tag{31}$$

Proof. Let g be defined as in eq. (25); then

$$G\left(\frac{\mathbf{w} + \epsilon \mathbf{v}}{\|\mathbf{w} + \epsilon \mathbf{v}\|} \right) = \frac{1}{\|\mathbf{w} + \epsilon \mathbf{v}\|} (\mathbf{w} + \epsilon \mathbf{v})^\intercal g(\mathbf{w} + \epsilon \mathbf{v}). \tag{32}$$

We now analyse $\mathbf{w}^\intercal g(\mathbf{w}+\epsilon\mathbf{v})$ and $\epsilon\mathbf{v}^\intercal g(\mathbf{w}+\epsilon\mathbf{v})$ separately, starting with the former. Denote $I_i^\epsilon := \mathbb{1}\{(\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_i > 0\}$. Then $g(\mathbf{w}+\epsilon\mathbf{v})$ can be written as

$$g(\mathbf{w}+\epsilon\mathbf{v}) = -\sum_{i=1}^{n}\ell_i'(0)\cdot I_i^\epsilon I_i^0\,\mathbf{P}_\mathbf{w}\mathbf{x}_i - \sum_{i=1}^{n}\ell_i'(0)\cdot I_i^\epsilon(1-I_i^0)\,\mathbf{P}_\mathbf{w}\mathbf{x}_i - \sum_{i=1}^{n}\ell_i'(0)\cdot I_i^\epsilon\,(\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}}-\mathbf{P}_\mathbf{w})\mathbf{x}_i. \tag{33}$$

Define

$$\epsilon_{\max} := \frac{1}{2}\max_{\epsilon>0}\epsilon, \qquad \text{subject to: } \operatorname{sign}\{(\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_i\}\operatorname{sign}\{\mathbf{w}^\intercal\mathbf{x}_i\} \geq 0 \quad \forall i. \tag{34}$$

For $\epsilon \in [0,\epsilon_{\max}]$, $I_i^0 = 1$ implies $I_i^\epsilon = 1$, so the first term in eq. (41) equals $g(\mathbf{w})$. Regarding the second term, $I_i^\epsilon(1-I_i^0)$ is nonzero only if $I_i^\epsilon = 1, I_i^0 = 0$. For $\epsilon \in [0,\epsilon_{\max}]$, this can only happen if $(\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_i > 0$ and $\mathbf{w}^\intercal\mathbf{x}_i = 0$. In this scenario however, $\mathbf{P}_\mathbf{w}\mathbf{x}_i = \mathbf{0}$, so the second term in eq. (41) is zero. Regarding the third term, as long as $\epsilon \in [0,\epsilon_{\max}]$, $(\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_i = 0$ implies $\mathbf{w}^\intercal\mathbf{x}_i = 0$, so

$$\mathbf{w} \in \operatorname{span}\{\mathbf{x}_j |\, \mathbf{w}^\intercal\mathbf{x}_j = 0\}^\perp \subseteq \operatorname{span}\{\mathbf{x}_j |\, (\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_j = 0\}^\perp \tag{35}$$

and $\mathbf{w}^\intercal(\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}} - \mathbf{P}_\mathbf{w}) = \mathbf{w}^\intercal - \mathbf{w}^\intercal = \mathbf{0}^\intercal$. It follows that

$$\mathbf{w}^\intercal g(\mathbf{w}+\epsilon\mathbf{v}) = \mathbf{w}^\intercal g(\mathbf{w}) = G(\mathbf{w}). \tag{36}$$

Turning to $\epsilon\mathbf{v}^\intercal g(\mathbf{w}+\epsilon\mathbf{v})$, we have that

$$\epsilon\mathbf{v}^\intercal g(\mathbf{w}+\epsilon\mathbf{v}) = -\epsilon\mathbf{v}^\intercal\sum_{i=1}^{n}\ell_i'(0)\cdot\mathbb{1}\{(\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{x}_i > 0\}\,\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}}\mathbf{x}_i, \tag{37}$$

where

$$\epsilon\mathbf{v}^\intercal\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}} = (\mathbf{w}+\epsilon\mathbf{v})^\intercal\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}} - \mathbf{w}^\intercal\mathbf{P}_{\mathbf{w}+\epsilon\mathbf{v}} = (\mathbf{w}+\epsilon\mathbf{v})^\intercal - \mathbf{w}^\intercal = \epsilon\mathbf{v}^\intercal. \tag{38}$$

Plugging eq. (36) and eq. (37) into eq. (32) yields the result. □

A.2 Training dynamics

In the following lemma, we prove a formula for the evolution of the parameters of a two-layer network trained by gradient flow (eq. (4)). The formula has appeared in Maennel et al. (2018) before (but without a proof).

Lemma A.3. *Assume that the training inputs with the zero vector $\{\mathbf{x}_i\}_i \cup \{\mathbf{0}\}$ are in general position. Then a two-layer ReLU network trained by gradient flow on $(\mathbf{X},\mathbf{y})$ satisfies for all $j \in [p]$ and almost all $t \geq 0$,*

$$\frac{\partial a_j}{\partial t} = -\sum_{i=1}^{n}\ell_i'(t)\cdot\rho(\mathbf{w}_j^\intercal\mathbf{x}_i), \tag{39}$$

$$\frac{\partial \mathbf{w}_j}{\partial t} = -\sum_{i=1}^{n}\ell_i'(t)\,\mathbb{1}\{\mathbf{w}_j^\intercal\mathbf{x}_i > 0\}\,a_j\mathbf{P}_{\mathbf{w}_j}\mathbf{x}_i. \tag{40}$$

Proof. Fix $\boldsymbol{\theta}$, and denote by $\boldsymbol{\Sigma}_{\boldsymbol{\theta}} \in \{-1,0,1\}^{p\times n}$ the activation matrix for $f_{\boldsymbol{\theta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}[j,i] \triangleq \operatorname{sign}\mathbf{w}_j^\intercal\mathbf{x}_i$. Then for any sequence $\boldsymbol{\theta}_k \to \boldsymbol{\theta}$ such that $\{\nabla\ell(\boldsymbol{\theta}_k)\}$ exists and has a limit,

$$\lim_{k\to\infty}\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k} \in \Big\{\boldsymbol{\Sigma} \in \{\pm1\}^{p\times n} \,\Big|\, \boldsymbol{\Sigma}[i,j] = \boldsymbol{\Sigma}_{\boldsymbol{\theta}}[i,j] \text{ if } \boldsymbol{\Sigma}_{\boldsymbol{\theta}}[i,j] \neq 0\Big\}. \tag{41}$$

Conversely, for any $\boldsymbol{\Sigma}$ in the set above, there exists a sequence $\boldsymbol{\theta}_k \to \boldsymbol{\theta}$ such that $\lim_{k\to\infty}\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k} = \boldsymbol{\Sigma}$. To see this, observe that each $\mathbf{w}_j$ can be approached separately. Let $\mathbf{A}$ be the matrix whose rows are formed by those $\mathbf{x}_i$ for which $\mathbf{w}_j^\intercal\mathbf{x}_i = 0$. Then by the general position of inputs, $\mathbf{A}$ is a wide full-rank matrix and $\mathbf{A}\mathbf{w}_j = \mathbf{0}$. It follows that for any $\boldsymbol{\epsilon}$, $\mathbf{A}\mathbf{w} = \boldsymbol{\epsilon}$ has a solution, which can be chosen convergent to $\mathbf{w}_j$ as $\boldsymbol{\epsilon} \to \mathbf{0}$.

We deduce that

$$\partial\ell(\boldsymbol{\theta}(t)) = \operatorname{conv}\big\{\mathbf{g}(\boldsymbol{\Sigma}) \,|\, \boldsymbol{\Sigma}[i,j] \in \{\pm1\},\, \boldsymbol{\Sigma}[i,j] = \boldsymbol{\Sigma}_{\boldsymbol{\theta}(t)}[i,j] \text{ if } \boldsymbol{\Sigma}_{\boldsymbol{\theta}(t)}[i,j] \neq 0\big\}, \tag{42}$$

where we define $\mathbf{g}(\boldsymbol{\Sigma})$ coordinate-wise as

$$\begin{aligned} \mathbf{g}(\boldsymbol{\Sigma})[a_j] &:= \sum_{i=1}^{n} \ell_i'(t)\, \mathbb{1}\{\boldsymbol{\Sigma}[j,i]=1\}\, \mathbf{w}_j^\intercal \mathbf{x}_i, \\ \mathbf{g}(\boldsymbol{\Sigma})[\mathbf{w}_j] &:= \sum_{i=1}^{n} \ell_i'(t)\, \mathbb{1}\{\boldsymbol{\Sigma}[j,i]=1\}\, a_j \mathbf{x}_i. \end{aligned} \tag{43}$$

Since the value of $g(\boldsymbol{\Sigma})[a_j]$ is independent of $\boldsymbol{\Sigma}$, this proves eq. (39).

The proof of eq. (40) is slightly more complicated, as we need to pin down a single member of $\partial \ell(\boldsymbol{\theta}(t))$. To do that, we recall a result by Davis et al. (2020), who show that for a large class of deep learning scenarios (which includes ours), the objective ℓ admits a chain rule, i.e.

$$\ell'(t) = \left\langle \partial \ell(\boldsymbol{\theta}(t)), \frac{\partial \boldsymbol{\theta}}{\partial t} \right\rangle \qquad \text{for almost all } t \geq 0, \tag{44}$$

where the right-hand side above should be interpreted as the only element of the set $\{\mathbf{h}^\intercal \partial \boldsymbol{\theta}/\partial t \,|\, \mathbf{h} \in \partial \ell(\boldsymbol{\theta}(t))\}$. For t such that both eq. (4) and eq. (44) hold,

$$0 = \left\langle \partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)), \frac{\partial \boldsymbol{\theta}}{\partial t} \right\rangle, \tag{45}$$

implying that

$$\frac{\partial \boldsymbol{\theta}}{\partial t} \in \operatorname{span}\{\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)\}^{\perp}. \tag{46}$$

Suppose $\mathbf{h}_1, \mathbf{h}_2$ satisfy both eq. (4) and eq. (46) (taking the role of $\partial \boldsymbol{\theta}/\partial t$). Then

$$\mathbf{h}_1 - \mathbf{h}_2 \in (\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t))) \cap \operatorname{span}\{\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)\}^{\perp} = \{\mathbf{0}\}. \tag{47}$$

It follows that $\partial \boldsymbol{\theta}/\partial t$ is the unique member of both $\operatorname{span}\{\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)\}^{\perp}$ and $-\partial \ell(\boldsymbol{\theta}(t))$. By eqs. (42) and (43),

$$\operatorname{span}\{\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)\} \supseteq \operatorname{span}\left\{\boldsymbol{\xi}_{ij} \,|\, \mathbf{w}_j^\intercal \mathbf{x}_i = 0\right\}, \tag{48}$$

where $\boldsymbol{\xi}_{ij}[\mathbf{w}_j] = \mathbf{x}_i$ and all other elements of $\boldsymbol{\xi}_{ij}$ are zero. Since

$$\frac{\partial \boldsymbol{\theta}}{\partial t} \in \operatorname{span}\{\partial \ell(\boldsymbol{\theta}(t)) - \partial \ell(\boldsymbol{\theta}(t)\}^{\perp} \subseteq \operatorname{span}\left\{\boldsymbol{\xi}_{ij} \,|\, \mathbf{w}_j^\intercal \mathbf{x}_i = 0\right\}^{\perp}, \tag{49}$$

we obtain that for all (i,j) with $\mathbf{w}_j^\intercal \mathbf{x}_i = 0$, $\partial \mathbf{w}_j/\partial t \perp \mathbf{x}_i$. In other words,

$$\frac{\partial \mathbf{w}_j}{\partial t} = \mathbf{P}_{\mathbf{w}_j} \frac{\partial \mathbf{w}_j}{\partial t} \in \operatorname*{conv}_{\boldsymbol{\Sigma}} \left\{ -\sum_{i=1}^{n} \ell_i'(t)\, \mathbb{1}\{\boldsymbol{\Sigma}[j,i]=1\}\, a_j \mathbf{P}_{\mathbf{w}_j} \mathbf{x}_i \right\}, \tag{50}$$

where the inclusion follows from $\partial \boldsymbol{\theta}/\partial t \in -\partial \ell(\boldsymbol{\theta}(t))$ and eqs. (42) and (43). Now observe that by definition, $\mathbf{P}_{\mathbf{w}_j} \mathbf{x}_i = \mathbf{0}$ for all (i,j) with $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}[j,i] = 0$, hence the set in eq. (50) is a singleton whose only element equals eq. (40). □

The following lemma shows that a balanced two-layer network remains balanced and neurons keep their signs.

Lemma A.4. *If a two-layer neural network is balanced at initialisation and trained by gradient flow with a loss whose derivative is bounded, then for $t \geq 0$,*

$$a_j(t) = \operatorname{sign} a_j(0) \cdot \|\mathbf{w}_j(t)\|. \tag{51}$$

Proof. By Lemma A.3, for almost all $t \geq 0$,

$$\frac{\partial \|\mathbf{w}_j\|^2}{\partial t} = -\sum_{i=1}^{n} \ell_i'(t)\, \mathbb{1}\left\{\mathbf{w}_j^\intercal \mathbf{x}_i > 0\right\} a_j \mathbf{w}_j^\intercal \mathbf{x}_i = \frac{\partial a_j^2}{\partial t}, \tag{52}$$

i.e. a_j and $\mathbf{w}_j$ grow equally fast. Since $|a_j(0)| = \|\mathbf{w}_j(0)\|$ at initialisation, $|a_j(t)| = \|\mathbf{w}_j(t)\|$ throughout training.

Next denote by $B, V > 0$ some scalars such that $|\ell_i'(u)| \leq B$ for all $i \in [n]$ and $u \in \mathbb{R}$, and $\|\mathbf{x}_i\| \leq V$ for all $i \in [n]$. Then $\left|\partial a_j^2/\partial t\right| \leq nBa_j^2 V$, or equivalently $\left|\partial \log a_j^2/\partial t\right| \leq nBV$. It follows that $a_j^2(t)$ lies between $a_j^2(0)\exp(-nBVt)$ and $a_j^2(0)\exp(nBVt)$, and hence a_j cannot cross zero in finite time, proving eq. (51). □

B PROOFS OF MAIN RESULTS

Lemma 2. *In the setting of Theorem 1, there is exactly one positive extremal direction and exactly one negative extremal direction. The positive extremal sector $\boldsymbol{\sigma}^+$ is given by*

$$\sigma_j^+ = \begin{cases} 1, & \text{if } y_j = 1, \\ -1, & \text{if } y_j = -1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i < 0 \text{ for some } i \text{ with } y_i = 1, \\ 0, & \text{if } y_j = -1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i = 0 \text{ for all } i \text{ with } y_i = 1, \end{cases} \tag{12}$$

and the negative extremal sector $\boldsymbol{\sigma}^-$ is given by

$$\sigma_j^- = \begin{cases} 1, & \text{if } y_j = -1, \\ -1, & \text{if } y_j = 1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i < 0 \text{ for some } i \text{ with } y_i = -1, \\ 0, & \text{if } y_j = 1 \text{ and } \mathbf{x}_j^\intercal \mathbf{x}_i = 0 \text{ for all } i \text{ with } y_i = -1. \end{cases} \tag{13}$$

Proof. We will prove the positive case; the negative case follows by inverting all labels. Because G is a continuous function on a compact domain, it has a maximum. At least one maximum must moreover be strict, or otherwise G would have to be constant. This shows that a positive extremal direction exists; we now show there is no more than one such direction.

By Lemma A.1, there cannot be more than one extremal direction per sector; it therefore suffices to show that no sector except one, $\boldsymbol{\sigma}^+$, admits a positive extremal direction. We will show that if $\mathbf{w} \in \mathbb{S}^{d-1}$ lies in any sector other than $\boldsymbol{\sigma}^+$, then $\mathbf{w}$ is not positive extremal; in particular we show that $G(\mathbf{w})$ can be locally increased.

Let $\boldsymbol{\sigma} \neq \boldsymbol{\sigma}^+$ and let $\mathbf{w} \in \mathcal{S}_{\boldsymbol{\sigma}} \cap \mathbb{S}^{d-1}$. By Lemma A.2, for any $\mathbf{v} \in \mathbb{R}^d$ there exists $\epsilon_{\max} > 0$ such that for $\epsilon \in (0, \epsilon_{\max}]$,

$$G\left(\frac{\mathbf{w} + \epsilon \mathbf{v}}{\|\mathbf{w} + \epsilon \mathbf{v}\|}\right) = \frac{G(\mathbf{w}) + \epsilon\alpha}{\|\mathbf{w} + \epsilon \mathbf{v}\|}, \tag{53}$$

where

$$\alpha := -\sum_{i=1}^n \ell_i'(0)\, \mathbb{1}\{(\mathbf{w} + \epsilon\mathbf{v})^\intercal \mathbf{x}_i > 0\} \mathbf{v}^\intercal \mathbf{x}_i. \tag{54}$$

We now analyse the different possible realisations of $\boldsymbol{\sigma}$, and for each we find $\mathbf{v} \in \mathbb{R}^d$ such that $(G(\mathbf{w}) + \epsilon\alpha)/\|\mathbf{w} + \epsilon\mathbf{v}\| > G(\mathbf{w})$ for small ϵ.

Suppose first that $\sigma_j = -1$ for some example with $y_j = 1$, or that $\sigma_j = 1$ for some example with $y_j = -1$. Then set $\mathbf{v} := y_j \mathbf{x}_j / \|\mathbf{x}_j\|$. By orthogonal separability, we have that $\alpha \geq 0$. Also, $\sigma_j \triangleq \operatorname{sign} \mathbf{w}^\intercal \mathbf{x}_j = -y_j$ implies $\mathbf{w}^\intercal \mathbf{v} < 0$, therefore $\|\mathbf{w} + \epsilon\mathbf{v}\| < \|\mathbf{w}\| = 1$ for ϵ small enough. It follows that $(G(\mathbf{w}) + \epsilon\alpha)/\|\mathbf{w} + \epsilon\mathbf{v}\| > G(\mathbf{w})$.

Next suppose that $\sigma_j = 0$ for some example with $y_j = 1$, and set $\mathbf{v} := \mathbf{x}_j / \|\mathbf{x}_j\|$. Then $\alpha > 0$, because each term in eq. (54) is non-negative by orthogonal separability, and the term corresponding to $i = j$ is strictly positive:

$$-\ell_j'(0)\, \mathbb{1}\{(\mathbf{w} + \epsilon\mathbf{v})^\intercal \mathbf{x}_j > 0\}\, \mathbf{v}^\intercal \mathbf{x}_j = -\ell_j'(0)\, \mathbb{1}\{\epsilon \|\mathbf{x}_j\| > 0\} \|\mathbf{x}_j\| > 0. \tag{55}$$

From $\sigma_j \triangleq \operatorname{sign} \mathbf{w}^\intercal \mathbf{x}_j = 0$ it further follows that $\|\mathbf{w} + \epsilon\mathbf{v}\| = \sqrt{1 + \epsilon^2}$. Hence, $(G(\mathbf{w}) + \epsilon\alpha)/\|\mathbf{w} + \epsilon\mathbf{v}\| = (G(\mathbf{w}) + \epsilon\alpha)/(1 + O(\epsilon^2))$, which strictly exceeds $G(\mathbf{w})$ for ϵ small enough.

We have thus shown that if $\boldsymbol{\sigma}$ is positive extremal, then necessarily $\sigma_i = 1$ for all examples with $y_i = 1$, and $\sigma_i \in \{0, -1\}$ for examples with $y_i = -1$. Suppose now that $\sigma_j = 0$ for an example with $y_j = -1$ that satisfies $\mathbf{x}_j^\intercal \mathbf{x}_k < 0$ for some k with $y_k = 1$. Taking $\mathbf{v} := -\mathbf{x}_j / \|\mathbf{x}_j\|$ will make α strictly positive, as the term corresponding to $i = k$ in eq. (54) will be strictly positive (this term's indicator equals 1, as we know from the above that $\sigma_k = 1$). Like in the previous paragraph, $\|\mathbf{w} + \epsilon\mathbf{v}\| = 1 + O(\epsilon^2)$, which suffices to show $G(\mathbf{w})$ is locally submaximal.

Finally, let $\mathbf{x}_j$ be such that $y_j = -1$ and $\mathbf{x}_j^\intercal \mathbf{x}_i = 0$ for all i with $y_i = 1$, and suppose that $\sigma_j = -1$. With $\mathbf{v} := \mathbf{P}_{\mathbf{w}} \mathbf{x}_j / \|\mathbf{P}_{\mathbf{w}} \mathbf{x}_j\|$ (we know that $\mathbf{P}_{\mathbf{w}} \mathbf{x}_j \neq \mathbf{0}$ because $\mathbf{w}^\intercal \mathbf{x}_j \neq 0$ by $\sigma_j = -1$), we have

$$\alpha = -\frac{1}{\|\mathbf{P}_{\mathbf{w}} \mathbf{x}_j\|} \sum_{i=1}^n \ell_i'(0)\, \mathbb{1}\{(\mathbf{w} + \epsilon\mathbf{v})^\intercal \mathbf{x}_i > 0\}\, \mathbf{x}_j^\intercal \mathbf{P}_{\mathbf{w}} \mathbf{x}_i. \tag{56}$$

We claim that each term in eq. (56) is zero: For terms with $\sigma_i = -1$, the indicator $\mathbb{1}\{(\mathbf{w}+\epsilon\mathbf{v})^\intercal \mathbf{x}_i > 0\}$ is zero. For terms with $\sigma_i = 0$, $\mathbf{P}_\mathbf{w}\mathbf{x}_i = \mathbf{0}$. (Also notice that such terms necessarily satisfy $y_i = -1$ and $\mathbf{x}_i^\intercal \mathbf{x}_k = 0$ for all k with $y_k = 1$, which we will need shortly.) Lastly, for terms with $\sigma_i = 1$, we know $y_i = 1$, and hence $\mathbf{x}_i^\intercal \mathbf{x}_l = 0$ for all l with $\sigma_l = 0$. In other words, $\mathbf{x}_i \perp \operatorname{span}\{\mathbf{x}_l | \mathbf{w}^\intercal \mathbf{x}_l = 0\}$, implying $\mathbf{P}_\mathbf{w}\mathbf{x}_i = \mathbf{x}_i$. In the context of eq. (56), we obtain $\mathbf{x}_j^\intercal \mathbf{P}_\mathbf{w}\mathbf{x}_i = \mathbf{x}_j^\intercal \mathbf{x}_i = 0$, concluding the proof that $\alpha = 0$. Since $\sigma_j = -1$, $\|\mathbf{w}+\epsilon\mathbf{v}\| < \|\mathbf{w}\| = 1$ for small enough ϵ, and $(G(\mathbf{w}) + \epsilon\alpha)/\|\mathbf{w}+\epsilon\mathbf{v}\| > G(\mathbf{w})$. We have thus ruled out all sectors except $\boldsymbol{\sigma}^+$, proving that for orthogonally separable datasets there is a unique positive extremal sector. □

Lemma 3. *Assume the setting of Theorem 1. If at time T the neuron $(a_j, \mathbf{w}_j)$ satisfies $a_j(T) > 0$ and $\mathbf{w}_j(T) \in \mathcal{S}_{\boldsymbol{\sigma}}$, where $\boldsymbol{\sigma}$ is the positive extremal sector (eq.* (12)*), then for $t \geq T$, $\mathbf{w}_j(t) \in \mathcal{S}_{\boldsymbol{\sigma}}$. The same holds if $a_j(T) < 0$ and $\boldsymbol{\sigma}$ is the negative extremal sector (eq.* (13)*).*

Proof. We omit the neuron index, and only prove the positive case; the negative case is analogous. Denote $\boldsymbol{\sigma}(t) := \operatorname{sign}(\mathbf{X}^\intercal \mathbf{w}(t))$. We proceed by contradiction. Suppose there exists a time $T_1 > T$ such that $\boldsymbol{\sigma}(T_1) \neq \boldsymbol{\sigma}(T)$. Wlog, take T_1 such that $\boldsymbol{\sigma}(t)$ is constant on (T, T_1) and denote this constant sector $\bar{\boldsymbol{\sigma}}$; by continuity $\bar{\sigma}_k = \sigma_k(T)$ if $\sigma_k(T) \neq 0$.

Now consider $\sigma_k(T) = 0$. By the gradient flow differential inclusion, for almost all $t \in (T, T_1)$,

$$\frac{\partial \mathbf{w}^\intercal \mathbf{x}_k}{\partial t} \in \operatorname*{conv}_{\boldsymbol{\sigma}'} \left\{ -\sum_{i=1}^n \ell_i'(t)\, \mathbb{1}\{\sigma_i' = 1\}\, a\mathbf{x}_i^\intercal \mathbf{x}_k \right\}, \tag{57}$$

where each $\boldsymbol{\sigma}'$ in the definition of the convex hull satisfies $\sigma_i' = \sigma_i(T)$ if $\sigma_i(T) \neq 0$, implying

$$\begin{aligned} \{i \,|\, \sigma_i' = 1\} &\subseteq \{i \,|\, \sigma_i(T) = 1\} \cup \{i \,|\, \sigma_i(T) = 0\} \\ &= \{i \,|\, y_i = 1\} \cup \{i \,|\, y_i = -1 \text{ and } \mathbf{x}_i^\intercal \mathbf{x}_j = 0 \text{ for all } j \text{ with } y_j = 1\}. \end{aligned} \tag{58}$$

Denote the two sets in the last expression $\mathcal{I}^+$ and $\mathcal{I}^0$, and consider the gradient corresponding to some $\boldsymbol{\sigma}'$ in eq. (57). The gradient terms corresponding to $i \in \mathcal{I}^+$ are zero (because $k \in \mathcal{I}^0$ and so $\mathbf{x}_i^\intercal \mathbf{x}_k = 0$) and the terms corresponding to $i \in \mathcal{I}^0$ (if there are any) are negative. The total gradient for $\boldsymbol{\sigma}'$ is therefore non-positive, which is preserved under taking convex hulls, and so we obtain $\frac{\partial}{\partial t}\mathbf{w}^\intercal \mathbf{x}_k \leq 0$. It follows that $\bar{\sigma}_k \neq 1$.

By Lemma A.3, for almost all $t \in (T, T_1)$ and any $k \in [n]$,

$$\frac{\partial \mathbf{w}^\intercal \mathbf{x}_k}{\partial t} = -\sum_{i=1}^n \ell_i'(t)\, \mathbb{1}\{\bar{\sigma}_i = 1\}\, a\mathbf{x}_i^\intercal \mathbf{P}_\mathbf{w}\mathbf{x}_k, \tag{59}$$

where $\mathbb{1}\{\bar{\sigma}_i = 1\} = \mathbb{1}\{y_i = 1\}$ as we have shown above. Observe that for $\mathbf{x}_i$ with $y_i = 1$, we have $\mathbf{P}_\mathbf{w}\mathbf{x}_i = \mathbf{x}_i$. This is because $\mathbf{P}_\mathbf{w}$ projects onto

$$\operatorname{span}\{\mathbf{x}_i \,|\, \bar{\sigma}_i = 0\}^\perp \supseteq \operatorname{span}\{\mathbf{x}_i \,|\, \sigma_i(T) = 0\}^\perp \tag{60}$$

and $\mathbf{x}_i$ lies in the right-hand side by the definition of positive extremal sector (eq. (12)). Therefore

$$\frac{\partial \mathbf{w}^\intercal \mathbf{x}_k}{\partial t} = -\sum_{i=1}^n \ell_i'(t)\, \mathbb{1}\{y_i = 1\}\, a\mathbf{x}_i^\intercal \mathbf{x}_k. \tag{61}$$

One can easily check that if $\sigma_k(T) = 1$ then $\frac{\partial}{\partial t}\mathbf{w}^\intercal \mathbf{x}_k > 0$, if $\sigma_k(T) = -1$ then $\frac{\partial}{\partial t}\mathbf{w}^\intercal \mathbf{x}_k < 0$, and if $\sigma_k(T) = 0$ then $\frac{\partial}{\partial t}\mathbf{w}^\intercal \mathbf{x}_k = 0$. It follows that $\boldsymbol{\sigma}(T_1) = \boldsymbol{\sigma}(T)$, which is a contradiction. □

Corollary 1. *Under the conditions of Theorem 1, there exist constants $u, z \geq 0$ such that*

$$\frac{f_{\boldsymbol{\theta}(t)}(\mathbf{x})}{\|\boldsymbol{\theta}(t)\|^2} \to u\rho(\mathbf{w}_+^\intercal \mathbf{x}) - z\rho(\mathbf{w}_-^\intercal \mathbf{x}), \qquad \text{as } t \to \infty. \tag{9}$$

Proof. By Lemma A.4, $\|\boldsymbol{\theta}\|^2 = \|\mathbf{a}\|^2 + \|\mathbf{W}\|_F^2 = 2\|\mathbf{a}\|^2 = 2\|\mathbf{W}\|_F^2$. Then for any $\mathbf{x} \in \mathbb{R}^d$,

$$\frac{2f_{\boldsymbol{\theta}(t)}(\mathbf{x})}{\|\boldsymbol{\theta}(t)\|^2} = \frac{\mathbf{a}(t)^\intercal}{\|\mathbf{a}(t)\|}\, \rho\left(\frac{\mathbf{W}(t)\mathbf{x}}{\|\mathbf{W}(t)\|_F}\right). \tag{62}$$

Denote $\tilde{\mathbf{a}} := \lim_{t\to\infty} \mathbf{a}(t)/\|\mathbf{a}(t)\|$; this limit exists by (Ji & Telgarsky, 2020, Theorem 3.1). Then by Theorem 1, as $t \to \infty$,

$$\frac{2f_{\boldsymbol{\theta}(t)}(\mathbf{x})}{\|\boldsymbol{\theta}(t)\|^2} \to \tilde{\mathbf{a}}^\intercal \rho\big(\mathbf{u}\mathbf{w}_+^\intercal \mathbf{x} + \mathbf{z}\mathbf{w}_-^\intercal \mathbf{x}\big) \tag{63}$$

$$= \tilde{\mathbf{a}}^\intercal \rho(\mathbf{u}\mathbf{w}_+^\intercal \mathbf{x}) + \tilde{\mathbf{a}}^\intercal \rho(\mathbf{z}\mathbf{w}_-^\intercal \mathbf{x}) \tag{64}$$

$$= \tilde{\mathbf{a}}^\intercal \mathbf{u}\, \rho(\mathbf{w}_+^\intercal \mathbf{x}) + \tilde{\mathbf{a}}^\intercal \mathbf{z}\, \rho(\mathbf{w}_-^\intercal \mathbf{x}), \tag{65}$$

where in eq. (64) we used the fact that for all $i \in [p]$, either $u_i = 0$ or $z_i = 0$, and in eq. (65) we used $\mathbf{u}, \mathbf{z} \geq \mathbf{0}$. Finally, in the notation of **??**,

$$\tilde{\mathbf{a}}^\intercal \mathbf{u} = \lim_{t\to\infty} \frac{\mathbf{a}(t)^\intercal \mathbf{u}}{\|\mathbf{a}(t)\|} = \lim_{t\to\infty} \frac{\mathbf{a}_+(t)^\intercal \bar{\mathbf{u}}}{\|\mathbf{a}(t)\|} \geq 0, \tag{66}$$

where the positivity follows from $\mathbf{a}_+(t), \bar{\mathbf{u}} \geq \mathbf{0}$. By similar considerations we obtain that $\tilde{\mathbf{a}}^\intercal \mathbf{z} \leq 0$, completing the proof. □