CONVERGENCE ANALYSIS AND IMPLICIT REGULARIZATION OF FEEDBACK ALIGNMENT FOR DEEP LINEAR NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

We theoretically analyze the Feedback Alignment (FA) algorithm, an efficient alternative to backpropagation for training neural networks. We provide convergence guarantees with rates for deep linear networks for both continuous and discrete dynamics. Additionally, we study incremental learning phenomena for shallow linear networks. Interestingly, certain specific initializations imply that negligible components are learned before the principal ones, thus potentially negatively affecting the effectiveness of such a learning algorithm; a phenomenon we classify as implicit anti-regularization. We also provide initialization schemes where the components of the problem are approximately learned by decreasing order of importance, thus providing a form of implicit regularization.

1 INTRODUCTION

In order to train a Machine Learning (ML) architecture in the context of supervised learning, it is a consolidated practice to resort to the Gradient Descent (GD) algorithm and its variants (e.g. stochastic GD) and to calculate the gradient of the loss function via backpropagation (Rumelhart et al., 1986). However, one of its limitations is the so-called weight transport problem (Grossberg, 1987): the update of each neuron during the learning phase relies on the downstream weights (which are therefore "transported" along the backward pass). This implies that 1) the synaptic weights are the same, up to transposition, for both the forward (inference) and backward (learning) pass and 2) each neuron has a precise knowledge of all of its forward synapses. It is known that such a model does not accurately reflect the behaviour of a human brain, thus it is biologically implausible (Crick, 1989). Feedback Alignment (FA), originally proposed by Lillicrap (Lillicrap et al., 2016), has been an attempt to mitigate this discrepancy between the neuroscience side and the engineering side of AI and to tackle the weight transport problem in a systematic way. The simple, yet powerful idea is to replace the transposed weight matrices in the backpropagation algorithm with arbitrary, fixed matrices: via this strategy, each neuron during training receives a random projection of the error vector.

In order to illustrate the algorithm and for the purpose of our study, let us consider a fully-connected L-layer neural network (NN): given the input, $\boldsymbol{x} = \boldsymbol{h}_0 \in \mathbb{R}^d$, the predicted label $\hat{\boldsymbol{y}} \in \mathbb{R}^o$ is computed as

$$\boldsymbol{a}_\ell = \boldsymbol{W}_\ell \boldsymbol{h}_{\ell-1}\,, \quad \boldsymbol{h}_\ell = \sigma(\boldsymbol{a}_\ell)\,, \quad \ell \in \{1, \dots, L\} := [L], \tag{1}$$

σ being a (potentially) nonlinear function applied entry-wise to the vector $\boldsymbol{a}_\ell$, and finally $\hat{\boldsymbol{y}} = f(\boldsymbol{a}_L)$ for some output nonlinear function f. We consider a regression task with loss function $\mathcal{L}$ being the standard Mean Square Error (MSE). The backpropagation strategy for GD prescribes the following updates for each layer in the network: given a step size $\eta > 0$ and the error vector $\boldsymbol{e} = \delta \boldsymbol{a}_L = \frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} = \hat{\boldsymbol{y}} - \boldsymbol{y}$ where $\boldsymbol{y}$ is the true label, we have

$$\delta \boldsymbol{W}_L = -\eta \boldsymbol{e} \boldsymbol{h}_{L-1}^\top, \;\; \delta \boldsymbol{W}_\ell = -\eta \delta \boldsymbol{a}_\ell \boldsymbol{h}_{\ell-1}^\top \;\text{ and }\; \delta \boldsymbol{a}_\ell = \left(\boldsymbol{W}_{\ell+1}^\top \delta \boldsymbol{a}_{\ell+1}\right) \odot \sigma'(\boldsymbol{a}_\ell), \; \ell \in [L-1] \tag{GD}$$

where $\odot$ denotes the Hadamard product. The FA recipe prescribes the following:

$$\textbf{FeedBack Alignement:} \quad \text{in (GD) use} \quad \delta \boldsymbol{a}_\ell = (\boldsymbol{M}_\ell \delta \boldsymbol{a}_{\ell+1}) \odot \sigma'(\boldsymbol{a}_\ell)\,, \quad \ell \in [L-1]\,. \tag{2}$$

where $\{\boldsymbol{M}_\ell\}$ are a collection of arbitrarily chosen matrices, which do not evolve during training.

Related and previous work. Inspired by Lillicrap et al. (2016) and shortly after its publication, Nøkland (2016) proposed two new algorithms called Direct and Inverse FA. In particular, Direct FA reduces to the FA algorithm in the case of a 2-layer NN or multi-layer linear NN; the difference between these two algorithms in the general multi-layer nonlinear NN resides in injecting the error vector directly into the update of each hidden layer of the network: $\delta \boldsymbol{a}_\ell = (\boldsymbol{M}_\ell \boldsymbol{e}) \odot \sigma'(\boldsymbol{a}_\ell)$, $\ell \in [1, L-1]$. Such a modification of the FA algorithm can be interpreted as a noisy version of layer-wise training (Gilmer et al., 2017).

Since its first formulation, there has been an extensive numerical analysis on testing whether FA and Direct FA algorithms can be successfully applied to ML problems, in particular in the presence of complex datasets and deep architectures (Bartunov et al., 2018; Moskovitz et al., 2018; Launay et al., 2019). We mention the recent survey paper Launay et al. (2020) where the focus is on Direct FA and it is shown that the algorithm performs well in modern deep architectures (Transformers and Graph NN, among others) applied to problems like neural view synthesis or natural language processing. Furthermore, more on the applied (biologically-inspired) side, there have been recent pushes in furthering the idea of FA: spike-train level direct feedback alignment (Lee et al., 2020), memory-efficient direct feedback alignment (Chu et al., 2020) and direct random target projection (Frenkel et al., 2021) to mention a few examples.

On the other hand, rigorous theoretical results are scarce. Preliminary asymptotic results can be found in the original papers by Lillicrap et al. (2016) and Nøkland (2016). More recently, Refinetti et al. (2020) presented the derivation of a system of differential equations that governs the Direct FA dynamics for a 2-layer non-linear NN in the regime where the input dimension goes to infinity. There is no explicit formula for the solution of the system, but via a careful local analysis it is possible to identify two separate phases of learning (alignment and memorization) and a "degeneracy breaking" effect: at the end of the FA training, the selected solution maximizes the overlap between the second weight matrix $\boldsymbol{W}_2$ and the FA matrix $\boldsymbol{M}$. Such an "alignment" dynamics is also evident in our analysis (Section 2). Convergence guarantees for deep networks are still missing from the literature and one of the goals of this paper is indeed to provide rigorous proofs of convergence, together with convergence rates.

Our analysis is in a similar vein as some recent works on implicit regularization in linear neural networks Saxe et al. (2018); Gidel et al. (2019); Arora et al. (2018). Even though our setting is very similar, our analysis is different: because of the feedback alignment matrices, the autonomous differential equation obtained significantly differ and lead to different dynamics. Moreover our focus is slightly different from Saxe et al. (2018); Gidel et al. (2019); Arora et al. (2018) which are fully dedicated to implicit regularization of GD while we also aim at proving the convergence of FA.

Our contributions. We aim to provide a theoretical understanding of how and when FA works. We focus on two aspects: 1) proving convergence of the algorithm and 2) analyzing implicit regularization phenomena. We will extensively study the case of *deep linear* NNs. Despite being sometimes regarded as too simplistic, the study of linear networks is a crucial starting point for a systematic analysis of the FA algorithm: it will allow us to rigorously analyze a tractable model and it will give us insights and intuition on the general nonlinear setting (Section 7).

Our main results are the following.

- We prove convergence of the FA continuous dynamics, with rates, for L-layer NNs with $L \geq 2$ (Sections 2 and 3). Such results hold for any input dimension and number of neurons: the network is not necessarily overparametrized.
- We prove that, for certain initialization schemes, the continuous FA dynamics sequentially learn the solutions of a reduced-rank regression problem, but in a backward fashion: smaller components first, dominant components later (Section 4). Additionally, we provide guidelines for an initialization scheme that avoids such phenomena.
- Finally (Section 5), we analyze the corresponding discrete FA dynamics and we prove convergence to the true labels with linear convergence rates.

We assumed some conditions on the data and on the model in order to render the exposition and the reading more clear. However, some of the "strong" assumptions that will appear in the paper can be easily relaxed (see Section 2).

2 WARM-UP: SHALLOW NETWORKS

In this section, we consider a 2-layer *linear* NN with vector output $\hat{\boldsymbol{y}} = \boldsymbol{W}_2\boldsymbol{W}_1\boldsymbol{x} \in \mathbb{R}^o$ and input vector $\boldsymbol{x} \in \mathbb{R}^d$ such that $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}$ for some distribution $\mathcal{D}$. Define $\boldsymbol{\Sigma}_{xx} := \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top] \in \mathbb{R}^{d\times d}$, that we assume to be positive definite, and $\boldsymbol{\Sigma}_{xy} := \mathbb{E}[\boldsymbol{y}\boldsymbol{x}^\top] \in \mathbb{R}^{o\times d}$, then the following result holds:

Proposition 1. *For any distribution $\mathcal{D}$, the FA dynamics that dictates the learning process is*

$$\dot{\boldsymbol{W}}_1 = \boldsymbol{M}\left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2\boldsymbol{W}_1\boldsymbol{\Sigma}_{xx}\right), \qquad \dot{\boldsymbol{W}}_2 = \left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2\boldsymbol{W}_1\boldsymbol{\Sigma}_{xx}\right)\boldsymbol{W}_1^\top, \tag{3}$$

where the dot refers to the time derivative.

Note that this result particularly holds for empirical distributions corresponding to finite datasets.

Classical results on comparative spectral decomposition (van Loan, 1976; Paige & Saunders, 1981) guarantee that the two matrices $\boldsymbol{\Sigma}_{xx}, \boldsymbol{\Sigma}_{xy}$ can be simultaneously decomposed as $\boldsymbol{\Sigma}_{xx} = \boldsymbol{V}\boldsymbol{\Lambda}_{xx}\boldsymbol{V}^\top$ and $\boldsymbol{\Sigma}_{xy} = \boldsymbol{U}\boldsymbol{\Lambda}_{xy}\boldsymbol{V}^\top$, where the latter corresponds to the singular value decomposition (SVD) of $\boldsymbol{\Sigma}_{xy}$. Next, we introduce the change of variables $\boldsymbol{W}_1 = \boldsymbol{R}\tilde{\boldsymbol{W}}_1\boldsymbol{V}^\top$ and $\boldsymbol{W}_2 = \boldsymbol{U}\tilde{\boldsymbol{W}}_2\boldsymbol{R}^\top$, for some arbitrary left-orthogonal matrix $\boldsymbol{R}$, and we choose the FA matrix $\boldsymbol{M}$ to be decomposable in a similar fashion: $\boldsymbol{M} = \boldsymbol{R}\boldsymbol{D}\boldsymbol{U}^\top$, for some matrix $\boldsymbol{D}$ with positive diagonal entries $\boldsymbol{D}_{ii} > 0$ and zero entries otherwise. In particular, this implies that $\boldsymbol{M}$ is full rank, therefore guaranteeing convergence to a global minimum of the dynamics. The equations for the FA flow will then become

$$\dot{\tilde{\boldsymbol{W}}}_1 = \boldsymbol{D}\left(\boldsymbol{\Lambda}_{xy} - \tilde{\boldsymbol{W}}_2\tilde{\boldsymbol{W}}_1\boldsymbol{\Lambda}_{xx}\right) \qquad \dot{\tilde{\boldsymbol{W}}}_2 = \left(\boldsymbol{\Lambda}_{xy} - \tilde{\boldsymbol{W}}_2\tilde{\boldsymbol{W}}_1\boldsymbol{\Lambda}_{xx}\right)\tilde{\boldsymbol{W}}_1^\top. \tag{4}$$

Moreover, by the change of variable $\boldsymbol{W}_1' := \tilde{\boldsymbol{W}}_1\boldsymbol{\Lambda}_{xx}^{-1/2}$, $\boldsymbol{W}_2' := \boldsymbol{\Lambda}_{xx}^{-1/2}\tilde{\boldsymbol{W}}_2$, $\boldsymbol{D}' := \boldsymbol{D}\boldsymbol{\Lambda}_{xx}^{-1/2}$ where $\boldsymbol{\Lambda}_{xx}^{1/2}$ is the coordinate-wise square root of $\boldsymbol{\Lambda}_{xx}$,[1] we get after some simple calculations that

$$\dot{\boldsymbol{W}}_1' = \boldsymbol{D}'\left(\boldsymbol{\Lambda}_{xx}^{1/2}\boldsymbol{\Lambda}_{xy}\boldsymbol{\Lambda}_{xx}^{-1/2} - \boldsymbol{W}_2'\boldsymbol{W}_1'\right), \quad \dot{\boldsymbol{W}}_2' = \left(\boldsymbol{\Lambda}_{xx}^{1/2}\boldsymbol{\Lambda}_{xy}\boldsymbol{\Lambda}_{xx}^{-1/2} - \boldsymbol{W}_2'\boldsymbol{W}_1'\right)\boldsymbol{W}_1'^\top. \tag{5}$$

Therefore, by considering $\boldsymbol{\Lambda}_{xy}' := \boldsymbol{\Lambda}_{xx}^{1/2}\boldsymbol{\Lambda}_{xy}\boldsymbol{\Lambda}_{xx}^{-1/2}$, we may equivalently assume $\boldsymbol{\Sigma}_{xx} = \boldsymbol{I}_d$ (isotropic features), without any loss of generality.

It is now crucial to observe that if $\boldsymbol{W}_1' \in \mathbb{R}^{h\times d}, \boldsymbol{W}_2' \in \mathbb{R}^{o\times h}$ are diagonal matrices, then the dynamics decouples into k independent equations with $k = \min\{d, h, o\}$. In particular, let $\theta_1^i = (\boldsymbol{W}_1')_{ii}$ and $\theta_2^i = (\boldsymbol{W}_2')_{ii}$ and assume that at time $t = 0$ the matrices $\boldsymbol{W}_1'(0)$ and $\boldsymbol{W}_2'(0)$ are diagonal. Then, for each $i = 1, \dots, k$ we obtain the following scalar system:

$$\dot{\theta}_1^i = d^i\left(\lambda^i - \theta_2^i\theta_1^i\right) \qquad \dot{\theta}_2^i = \left(\lambda^i - \theta_2^i\theta_1^i\right)\theta_1^i, \tag{6}$$

where $\lambda^i = (\boldsymbol{\Lambda}_{xy})_{ii}$ and $d^i = \boldsymbol{D}_{ii}$. Note that for $i > k$ one could define diagonal coefficients for the matrices $\boldsymbol{W}_1'$ or $\boldsymbol{W}_2'$, but their derivative will be 0, thus non-trivial dynamics only occur for $i \in [k]$.

2.1 THE SCALAR DYNAMICS

We now turn our attention to analyzing a nonlinear system of the form (6) for the scalar functions $\theta_1, \theta_2 \in C^1(\mathbb{R})$ (we suppressed the indices).

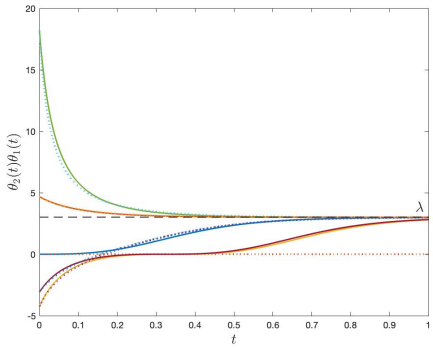
Theorem 2. *For any FA constant $d \in \mathbb{R}_{>0}$, and $\forall\ \theta_0 \in \mathbb{R}$ initial value with initialization scheme*

$$\theta_1(0) = \theta_0 \qquad \textit{and} \qquad \theta_2(0) = \frac{{\theta_0}^2}{2d}, \tag{7}$$

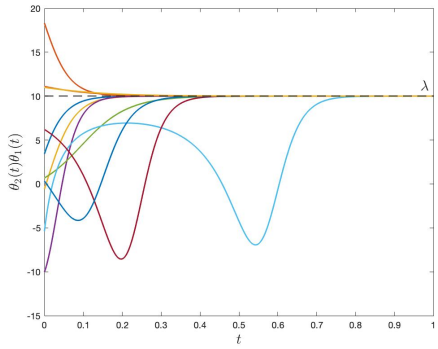
$\exists\ C > 0$ such that the product $\theta_2(t)\theta_1(t)$ of the solution of the system (6) converges exponentially to the signal λ and in a monotonic fashion: $|\theta_2(t)\theta_1(t) - \lambda| < Ce^{-\frac{3}{2}(d\lambda)^{\frac{2}{3}}t}$ as $t \to +\infty$.

The proof can be found in Appendix A.2: along with the proof, it appears that the FA weights drive the learning of the forward weights for the first layers to the point that their limit is "aligned" to the

[1] Note that we used a slight abuse of notation: depending on the matrix it is multiplied to $\boldsymbol{\Lambda}_{xx}^{-1/2}$ corresponds to the diagonal matrix either of size $d \times d$ or $o \times o$ with the diagonal coefficients $[\boldsymbol{\Lambda}_{xx}^{-1/2}]_{ii}$, $1 \le i \le \text{rank}(\boldsymbol{\Lambda}_{xx})$.



(a) FA (continuous lines) and GD (dotted lines), with initial conditions (7): $\theta_0 = 0$ and $\theta_0 \sim \mathcal{U}([-5, 5])$.



(b) FA dynamics with i.i.d initial conditions $(\theta_1(0), \theta_2(0)) \sim \mathcal{U}([-5, 5]) \times \mathcal{U}([-5, 5])$.

Figure 1: Numerical simulations of the solution $\theta_2(t)\theta_1(t)$ of the FA dynamics with FA constant $d = 2$ and true signal $\lambda = 3$ (left) and $\lambda = 10$ (right), with different initializations.

value of the FA constants (in a nonlinear fashion). Similarly, this holds for the deep network case (Section 3 and Appendix B.2). One can also observe that components with a large FA constant d are learned faster (see (35)–(36) in Appendix A.2), boosting the alignment.

From Theorem 2 it is clear that the choice of the FA constant will force the optimization to select *one* particular pair of solutions $(\theta_1(t), \theta_2(t))$ among the infinitely many possible solutions (the equation $\theta_1\theta_2 = \lambda$ is a 1-dimensional manifold on the $\mathbb{R}^2$-plane). However, all of these solutions represent global minima (those are the only stationary points of FA), and therefore via the FA dynamics, we can find the global minima of the loss function.

We can easily notice that zero-initialization does guarantee convergence for FA, unlike in the case of GD dynamics (Figure 1a). Furthermore, we want to highlight the ideological novelty of the proposed initialization scheme (7): unlike considering all weight matrices as independent players at the beginning of the optimization process and initializing them at random, we are imposing that once the weight matrix $\boldsymbol{W}_1$ of the first layers is initialized, the initial value of the weight matrix $\boldsymbol{W}_2$ of the second layer is deterministic, and it depends directly from the first layer.

More generally, it is easy (though lengthy) to prove that the product of the solutions $(\theta_1(t), \theta_2(t))$ of the system (6) always converges to the signal λ, regardless of the initialization scheme considered.

Theorem 3. *For any FA constant $d \in \mathbb{R}_{>0}$, and for any initial values $\theta_1(0)$ and $\theta_2(0)$, the system (6) admits a unique solution $(\theta_1(t), \theta_2(t))$ such that $\theta_1(t), \theta_2(t)$ are bounded for all times $t \geq 0$ and $\theta_2(t)\theta_1(t) \to \lambda$ as $t \to +\infty$. The convergence rate is exponential for any pair of initial values $(\theta_1(0), \theta_2(0))$ except for a set of (Lebesgue) measure zero in $\mathbb{R}^2$.*

For the proof, see Appendix A.3. The main consequence of Theorem 3 is that even by initializing all layers randomly and independently (as it is common practice), the FA dynamics still converges to the true signal. However, we want to stress that the initialization scheme (7) is more suitable for ML applications: for arbitrary initial conditions, the product of the solutions $\theta_2(t)\theta_1(t)$ may not be monotonic (compare with Theorem 2). Thus it may trigger inefficient early stopping practices (see Figure 1b). Furthermore, for a randomly and independently sampled pair of initial conditions, we may observe a highly undesirable "reversed" incremental learning phenomenon (Section 4), which is avoided if (7) is put into place.

Relaxing some assumptions. Considering the the assumption that the weight matrices $\boldsymbol{W}_1, \boldsymbol{W}_2$ share the left/right singular vectors with $\boldsymbol{\Sigma}_{xy}$ is not strictly needed: with more work, the more general case can be analyzed via a perturbation analysis, along the same lines as in Gidel et al. (2019) (Assumption 1). Furthermore, even though the initialization of the matrices $\tilde{\boldsymbol{W}}_1, \tilde{\boldsymbol{W}}_2$ are diagonal Saxe et al. (2018) argues that the diagonal solutions provide good approximations to the full solutions when $\tilde{\boldsymbol{W}}_1(0), \tilde{\boldsymbol{W}}_2(0)$ are full matrices, initialized with small random weights.

3 DEEP NETWORKS

In this section we extend the results of Section 2 to deep linear NNs: given a distribution of input-output pairs $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}$, the predicted output vector is now $\hat{\boldsymbol{y}} = \boldsymbol{W}_L \dots \boldsymbol{W}_2 \boldsymbol{W}_1 \boldsymbol{x}$ with weight matrices $\boldsymbol{W}_\ell \in \mathbb{R}^{h_\ell \times h_{\ell-1}}$ ($h_0 = d$, $h_L = o$) and we note $\boldsymbol{\Sigma}_{xx} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top] \in \mathbb{R}^{d\times d}$ and $\boldsymbol{\Sigma}_{xy} = \mathbb{E}[\boldsymbol{y}\boldsymbol{x}^\top] \in \mathbb{R}^{o\times d}$.

Proposition 4. *For any distribution $\mathcal{D}$, the FA equations of motion are*

$$\dot{\boldsymbol{W}}_\ell = \boldsymbol{M}_\ell(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_{[1:L]}\Sigma_{xx})\boldsymbol{W}_{[1:\ell-1]}^\top \qquad \ell \in [L], \tag{8}$$

for some FA weight matrices $\{\boldsymbol{M}_\ell\}_{\ell\in[L]}$, with $\boldsymbol{M}_L = \boldsymbol{I}$, and with $\boldsymbol{W}_{[1:\ell]} = \boldsymbol{W}_\ell \dots \boldsymbol{W}_1$.

Notice that the legitimate FA update would be $\dot{\boldsymbol{W}}_\ell = \tilde{\boldsymbol{M}}_{[\ell:L-1]}(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_{[1:L]}\Sigma_{xx})\boldsymbol{W}_{[1:\ell-1]}^\top$, however since we are requiring the FA matrices to be full rank, we can redefine the FA matrices as $\boldsymbol{M}_\ell = \tilde{\boldsymbol{M}}_{[\ell:L-1]}$. We now perform a similar change of variables as in Section 2: $\boldsymbol{W}_\ell = \boldsymbol{R}_\ell \tilde{\boldsymbol{W}}_\ell \boldsymbol{R}_{\ell-1}^\top$ with $\boldsymbol{R}_0 = \boldsymbol{V}$, $\boldsymbol{R}_L = \boldsymbol{U}$ and $\{\boldsymbol{R}_\ell\}_{\ell=1,\dots,L-1}$ arbitrary left-orthogonal matrices. We additionally assume that the FA matrices $\boldsymbol{M}_\ell$ have the following prescribed form $\boldsymbol{M}_\ell = \boldsymbol{R}_\ell \boldsymbol{D}_\ell \boldsymbol{U}^\top$ (with $\boldsymbol{D}_\ell$ a matrix with zero entries except for entries on the diagonal, which are positive). Then, the system becomes

$$\dot{\tilde{\boldsymbol{W}}}_\ell = \boldsymbol{D}_\ell(\boldsymbol{\Lambda}_{xy} - \tilde{\boldsymbol{W}}_{[1:L]}\boldsymbol{\Lambda}_{xx})\tilde{\boldsymbol{W}}_{[1:\ell-1]}^\top \qquad \ell \in [L]. \tag{9}$$

Considering the change of variable $\boldsymbol{W}_1' := \tilde{\boldsymbol{W}}_1 \boldsymbol{\Lambda}_{xx}^{-1/2}$, $\boldsymbol{W}_L' := \boldsymbol{\Lambda}_{xx}^{-1/2}\tilde{\boldsymbol{W}}_L$, $\boldsymbol{D}_\ell' := \boldsymbol{D}_\ell \boldsymbol{\Lambda}_{xx}^{-1/2}$, $\boldsymbol{\Lambda}_{xy}' := \boldsymbol{\Lambda}_{xx}^{1/2}\boldsymbol{\Lambda}_{xy}\boldsymbol{\Lambda}_{xx}^{-1/2}$ and assuming as in §2 that $\boldsymbol{W}_j'(0)$ are diagonal, the system decouples:

$$\dot{\theta}_\ell^i = d_\ell^i(\lambda^i - \theta_L^i \dots \theta_1^i)\theta_{\ell-1}^i \dots \theta_1^i, \qquad \ell \in [L], \tag{10}$$

and for each entry $i = 1, \dots, k$, with $k = \min\{d, h_1, \dots, h_{L-1}, o\}$, the analysis reduces to studying a system of scalar functions $\{\theta_1^i, \dots, \theta_L^i\}$.

Theorem 5. *For any set of FA constants $\{d_\ell\}_{\ell\in[L-1]}$, $d_\ell \in \mathbb{R}_{>0}$ and for any $\theta_0 \in \mathbb{R}$, there exists an initialization scheme of the form*

$$\theta_1(0) = \theta_0, \qquad \theta_\ell(0) = C_\ell \theta_0^{2^{\ell-1}}, \qquad \ell = 2, \dots, L, \tag{11}$$

where $C_\ell \in \mathbb{R}_{>0}$ depends on $\{d_1, \dots, d_\ell\}$, such that the system of differential equations (10) has a unique solution $(\theta_1(t), \dots, \theta_L(t))$. Furthermore, the following convergence result holds: $\theta_L(t)\theta_{L-1}(t)\dots\theta_1(t) \to \lambda$ as $t \to +\infty$.

As in the 2-layer case, the proposed initialization scheme is characterized by the fact that only the first layer is initialized at random, while all the other layers depend deterministically on it. Note that, along the same lines as Theorem 3, it is also possible to prove convergence to the true signal for any initialization set $\{\theta_\ell(0)\}_{\ell\in[L]}$. However, some desirable properties of the dynamics (monotonicity, exponential convergence) may be lost. For the proofs and the explicit construction of the initialization scheme (11) see Appendix B.2.

4 IMPLICIT REGULARIZATION PHENOMENA

In this section, we analyze the phenomenon of hierarchical learning (or implicit regularization) for FA, as described in Gidel et al. (2019) and Gissin et al. (2019) in the case of GD: along the optimization path, features of the model are learned in a sequential way, so that one feature is fully learned before the dynamics starts to learn the next feature. For FA, we may observe a similar behavior as GD, however, in certain settings the dynamics seems to favor the learning of secondary component prior than the learning of principal components, i.e. signals λ^i's with small magnitude are learned earlier than those of big magnitude. This phenomenon is potentially highly problematic since smaller singular values λ^i's are normally tied to noisy measurements and do not represent nor give meaningful information on the structure of the data. Privileging negligible features goes against the standard principles of reasoned data analysis (Principal Component Analysis, for example). We will see that the occurrence of this behavior depends on particular initializations of the algorithm and it can be therefore successfully avoided.

Theorem 6 (Informal)**.** *Let $\lambda_1 > \ldots > \lambda^k$ be the principal components of $\mathbf{\Lambda}_{xy}$. There exists initialization schemes such that the component λ_i is incrementally learned at time T_i. Moreover, there exists two different initialization schemes such that*

$$\textbf{\textit{Anti-Regularization(§4.1):}}\ T_1 > \ldots > T_k \quad \textit{and} \quad \textbf{\textit{Regularization(§4.2):}}\ T_1 < \ldots < T_k \tag{12}$$

To illustrate such a phenomenon and rigorously describe it, consider the 2-layer case in the SVD-setting as in Section 2: $\forall\, i = 1, \ldots, k$, the system (6) can be equivalently rewritten as

$$\dot{\theta}_1^i = -\frac{1}{2}\left((\theta_1^i)^3 + 2dK^i\theta_1^i - 2d\lambda^i\right), \qquad \theta_1^i(0) = \theta_0^i, \tag{13}$$

and $\theta_2^i(t) = \frac{1}{2d}\theta_1^i(t)^2 + K^i$, with $K^i = \theta_2^i(0) - \frac{1}{2d^i}(\theta_1^i(0))^2 \in \mathbb{R}$ and $d, \lambda^i > 0$ (for simplicity we assumed $d^i = d, \forall\, i$). Depending on the value of K^i (i.e. the initial condition of the i-th system), the cubic polynomial on the right-hand-side of (13) may have between 1 and 3 distinct real roots. The value of the discriminant $\Delta = -4d^2\left(8d(K^i)^3 + 27(\lambda^i)^2\right)$ will dictate which regime we fall into.

We will be considering the case where $\Delta > 0$ and the case $K^i = 0$ (i.e. the initialization scheme (7)), which is a special instance of the case $\Delta < 0$. The case $\Delta = 0$ can be equally analyzed, but it will not be considered in this paper, as it has zero probability of occurring.[2] The sequential learning phenomenon becomes evident when we perform a double scaling limit of the time variable $t \mapsto \delta t$ and the initial condition $\theta_0^i \mapsto f(\delta, \theta_0^i)$, $\delta \to +\infty$, as it will be clear below. Note that the rescaling is uniform across the components i. Proof of Theorem 7 and lengthier discussion can be found in Appendix C.

4.1 Case $\Delta > 0$: backward learning

If $K^i < -\frac{3}{2}(\lambda^i)^{3/2} < 0$, the cubic polynomial in (13) has 3 real, distinct root $r_1^i < r_2^i < r_3^i$. The intuition is that by initializing θ_0^i at a value that is very close to r_2^i and applying the time scaling $t \mapsto \delta t$, $\delta \gg 1$ (i.e. we are "fast-forwarding" the solution $\theta_1(\delta t)$), we can clearly identify the time threshold T^i after which the signal is fully learned. More formally,

Theorem 7. *$\forall\, i = 1, \ldots, k$, if $\theta_1^i(0) = r_2^i + e^{-\delta}$ ($\delta > 0$), then the solution $\theta_1^i(\delta t)$ converges to a step function:*

$$\theta_1^i(\delta t) \to r_2^i \mathbb{I}_{\{t<T^i\}} + \alpha^i \mathbb{I}_{\{t=T^i\}} + r_3^i \mathbb{I}_{\{t>T^i\}} \qquad \textit{as } \delta \to +\infty, \tag{14}$$

where $\mathbb{I}_{\{t\in A\}}$ is the indicator function on the set A,

$$T^i = \frac{2}{(r_3^i - r_2^i)(r_2^i - r_1^i)}, \qquad \alpha^i \in (r_2^i, r_3^i). \tag{15}$$

Furthermore, T^i is an increasing function of λ^i: for $\lambda^1 > \lambda^2 > \ldots > \lambda^k$ and fixed FA constant d, we have $T^1 > T^2 > \ldots > T^k$.

Similar conclusions hold when $\theta_1^i(0) = r_2^i - e^{-\delta}$. Therefore, as the value of λ^i increases, the value of T^i increases as well: negligible features of the data are learned sooner than the dominant ones.

On the other hand, Theorem 7 shows that such a backward incremental learning does *not* happen with high probability: indeed, we are requiring all the parameters K^i (which depends on the initialization) to be sufficiently negative and the probability of sampling initial conditions that will yield anti-regularization depends on the value of the signals λ^i (i.e., it is data-dependent): the bigger the value of λ^i, the smaller the probability. Nevertheless, we will see in the next subsection how such a phenomenon can be safely bypassed and we discuss some properties that suggest that the initialization scheme (7) can guarantee the correct sequential learning of features as GD exhibits.

4.2 Case $K = 0$: evidence of incremental learning

Imposing $K^i = 0$ for all i's means to recur to the initialization scheme (7): $\theta_1^i(0) = \theta_0^i$, $\theta_2^i(0) = \frac{1}{2d}(\theta_0^i)^2$. In this case, Theorem 2 already provides exponential convergence rates that depend on the

[2] The claim is correct if we sample the initial conditions $\theta_1(0), \theta_2(0)$ independently and generically: for example, from a uniform distribution $\mathcal{U}([-R, R]) \times \mathcal{U}([-R, R])$, for some $R \in \mathbb{R}_{>0}$.

magnitude of the signal ($|\theta_2^i(t)\theta_1^i(t) - \lambda^i| < Ce^{-\frac{3}{2}(d\lambda^i)^{\frac{2}{3}}t}$ as $t \to +\infty$), suggesting that the bigger the magnitude of the signal λ^i, the faster the convergence.

If $\theta_0^i < 0$, there exists a unique time T_0^i (vanishing time) such that $\theta_1^i(T_0^i)\theta_2^i(T_0^i) = 0$ and $\theta_1^i(t)\theta_2^i(t) \lesseqgtr 0$ for $t \lesseqgtr T_0^i$ respectively, where

$$T_0^i = \frac{\pi}{(r^i)^2 3\sqrt{3}} + \frac{1}{3(r^i)^2} \ln\left(\frac{(r^i - \theta_0^i)^2}{(\theta_0^i)^2 + r^i\theta_0^i + (r^i)^2}\right) - \frac{2}{(r^i)^2\sqrt{3}} \arctan\left(\frac{2\theta_0^i + r^i}{r^i\sqrt{3}}\right). \tag{16}$$

We can interpret such a vanishing time as the first milestone of the learning process. In the regime as $\theta_0^i \to -\infty \; \forall \, i = 1, \dots, k$, we have

$$T_0^i = \frac{4\pi}{3(r^i)^2\sqrt{3}} + \mathcal{O}\left(\frac{1}{(\theta_0^i)^2}\right); \tag{17}$$

where $r^i = \sqrt[3]{2d\lambda^i}$, therefore for bigger λ^i's (i.e. dominant singular values) the vanishing time happens sooner: for $\lambda^1 > \lambda^2 > \ldots > \lambda^k$ and fixed FA constant $d > 0$, $T_0^1 < T_0^2 < \ldots < T_0^k$.

5 Discrete dynamics

5.1 Forward Euler discretization

In this section, we are interested in analyzing the discrete FA dynamics and its convergence properties. For the discrete time setting, we focus again on a 2-layer linear NN. The standard forward Euler method yields the following discretized system:

$$\boldsymbol{W}_1^{(t+1)} = \boldsymbol{W}_1^{(t)} + \eta \boldsymbol{M}\left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2^{(t)}\boldsymbol{W}_1^{(t)}\right) \tag{18}$$

$$\boldsymbol{W}_2^{(t+1)} = \boldsymbol{W}_2^{(t+1)} + \eta\left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2^{(t)}\boldsymbol{W}_1^{(t)}\right)\left(\boldsymbol{W}_1^{(t)}\right)^\top \tag{19}$$

and performing again the SVD change of variables, the system becomes

$$\tilde{\boldsymbol{W}}_1^{(t+1)} = \tilde{\boldsymbol{W}}_1^{(t)} + \eta \boldsymbol{D}\left(\boldsymbol{\Lambda}_{xy} - \tilde{\boldsymbol{W}}_2^{(t)}\tilde{\boldsymbol{W}}_1^{(t)}\right) \tag{20}$$

$$\tilde{\boldsymbol{W}}_2^{(t+1)} = \tilde{\boldsymbol{W}}_2^{(t)} + \eta\left(\boldsymbol{\Lambda}_{xy} - \tilde{\boldsymbol{W}}_2^{(t)}\tilde{\boldsymbol{W}}_1^{(t)}\right)\left(\tilde{\boldsymbol{W}}_1^{(t)}\right)^\top \tag{21}$$

If we set the initial conditions $\boldsymbol{W}_1^{(0)}$, $\boldsymbol{W}_2^{(0)}$ as diagonal, the matrices remain diagonal for all times and the dynamic decouples. For each $i = 1, \dots, k$, we have

$$\left(\theta_1^{(t+1)}\right)^i = \left(\theta_1^{(t)}\right)^i + \eta d^i\left(\lambda^i - \left(\theta_2^{(t)}\right)^i\left(\theta_1^{(t)}\right)^i\right) \tag{22}$$

$$\left(\theta_2^{(t+1)}\right)^i = \left(\theta_2^{(t)}\right)^i + \eta\left(\lambda^i - \left(\theta_2^{(t)}\right)^i\left(\theta_1^{(t)}\right)^i\right)\left(\theta_1^{(t)}\right)^i \tag{23}$$

The following result holds (the index i is suppressed for simplicity).

Theorem 8. *Consider the system (22)–(23). With zero initialization $\theta_1^{(0)} = \theta_2^{(0)} = 0$, $\forall \, \lambda, d > 0$ and with constant step size η small enough, the product $\theta_2^{(t)}\theta_1^{(t)}$ converges to the true signal λ with linear rates: $\left|\theta_2^{(t)}\theta_1^{(t)} - \lambda\right| \leq Cq^t$, for some $C > 0$.*

For the sake of clarity, we deferred the precise prescription for the magnitude of η and the linear rate q to Appendix D, where the full proof of the Theorem can be found.

5.2 Modified forward discretization

In addition to the standard discretization strategy, we are proposing here a new discretization technique that is particularly instrumental for the FA setting, as it will allow a thorough inspection of its convergence guarantees and convergence rates for NNs of any depth. Inspired by the fact that the behaviour of the second layer strictly depend on the choices that we impose on the first layer in

the continuous dynamics (Theorem 2 and initialization (7)), we slightly modify the update rule for second weight matrix $\boldsymbol{W}_2$ in the following way:

$$\boldsymbol{W}_1^{(t+1)} = \boldsymbol{W}_1^{(t)} + \eta \boldsymbol{M} \left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2^{(t)} \boldsymbol{W}_1^{(t)} \right) \tag{24}$$

$$\boldsymbol{W}_2^{(t+1)} = \boldsymbol{W}_2^{(t+1)} + \eta \left(\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2^{(t)} \boldsymbol{W}_1^{(t)} \right) \left(\boldsymbol{W}_1^{\left(t+\frac{1}{2}\right)} \right)^\top \tag{25}$$

with $\boldsymbol{W}_1^{\left(t+\frac{1}{2}\right)} = \frac{1}{2} \left(\boldsymbol{W}_1^{(t+1)} + \boldsymbol{W}_1^{(t)} \right)$. The above discretization technique can be considered as a hybrid between the Euler forward method and the trapezoidal rule (Iserles, 1996), although we are still evaluating the error $\boldsymbol{\Sigma}_{xy} - \boldsymbol{W}_2^{(t)} \boldsymbol{W}_1^{(t)}$ at the present time t. The underlying idea is that we want to already use the information encoded in the new $(t+1)$-iterate of $\boldsymbol{W}_1$ to update the second layer by averaging $\boldsymbol{W}_1^{(t)}$ and $\boldsymbol{W}_1^{(t+1)}$.

Performing again the SVD change of variables, and setting $\boldsymbol{W}_1^{(0)}$, $\boldsymbol{W}_2^{(0)}$ as diagonal, we obtain a scalar system of the form ($\forall\, i = 1, \dots, k$)

$$\left(\theta_1^i\right)^{(t+1)} = \left(\theta_1^i\right)^{(t)} + \eta d^i \left(\lambda^i - \left(\theta_2^i\right)^{(t)} \left(\theta_1^i\right)^{(t)} \right) \tag{26}$$

$$\left(\theta_2^i\right)^{(t+1)} = \left(\theta_2^i\right)^{(t)} + \eta \left(\lambda^i - \left(\theta_2^i\right)^{(t)} \left(\theta_1^i\right)^{(t)} \right) \left(\theta_1^i\right)^{\left(t+\frac{1}{2}\right)}. \tag{27}$$

A convergence result follows (suppressing the i-dependence).

Theorem 9. *Consider the system (26)–(27). With zero initialization $\theta_1^{(0)} = \theta_2^{(0)} = 0$, $\forall\, \lambda, d > 0$ and with constant step size $\eta < \frac{2}{3(2d\lambda)^{\frac{2}{3}}}$, the product $\theta_2^{(t)}\theta_1^{(t)}$ linearly converges to the true signal λ: $|\theta_2^{(t)}\theta_1^{(t)} - \lambda| \leq 3\lambda(1 - \frac{3\eta}{2}(2d\lambda)^{\frac{2}{3}})^t$.*

The discretization scheme easily generalizes to the case of L layers for $L \geq 3$ (we again suppress the i-index for readability):

$$\theta_\ell^{(t+1)} = \theta_\ell^{(t)} + \eta d_\ell \theta_{[1:\ell-1]}^{\left(t+\frac{1}{2}\right)} \left(\lambda - \theta_L^{(t)} \dots \theta_1^{(t)} \right), \qquad \ell = 1, \dots, L, \tag{28}$$

with $d_L = 1$. Also in this case, the dynamics closely mimic the continuous counterpart, reducing the system to a single recurrence equation for $\{\theta_1^{(t)}\}$. By selecting a step size η small enough, the sequence of products $\{\theta_L^{(t)} \dots \theta_2^{(t)} \theta_1^{(t)}\}$ will converge to the signal λ with exponential rate.

Theorem 10. *Consider the system (28) with $\lambda, d_\ell > 0$ and zero initialization $\theta_\ell^{(0)} = 0$, $\forall\, \ell = 1, \dots, L$. If the constant step size satisfies $\eta < (d_1 \gamma (\mathfrak{K} \lambda^{\gamma-1})^{1/\gamma})^{-1}$, where $\gamma = 2^L - 1$ and $\mathfrak{K}$ is a suitable positive constant depending on the FA constants $d_1, \dots, d_{L-1}$, then the product $\prod_{\ell=1}^L \theta_\ell^{(t)}$ linearly converges to the true signal λ: $|\theta_L^{(t)} \dots \theta_1^{(t)} - \lambda| \leq C_\lambda (1 - \eta d_1 \gamma \left(\mathfrak{K} \lambda^{\gamma-1}\right)^{\frac{1}{\gamma}})^t$, for some constant $C_\lambda \in \mathbb{R}_{>0}$ only dependent on λ.*

The proofs of Theorems 9 and 10 can be found in Appendix D.

6 Experiments

In this last section, we discuss one ML application of our theoretical results. The set-up is borrowed from Gidel et al. (2019) (Section 4.2), where a linear autoencoder is considered.

For an autoencoder, the (true) output is equal to its input $\boldsymbol{y} = \boldsymbol{x} \in \mathbb{R}^d$ and we want to compare the reconstruction properties of $\boldsymbol{W}_2^{(t)} \boldsymbol{W}_1^{(t)}$ (2 layer NN) and $\boldsymbol{W}_3^{(t)} \boldsymbol{W}_2^{(t)} \boldsymbol{W}_1^{(t)}$ (3 layer NN) computed via FA versus GD. In this experiment, we have $d = o = 20$ and $k = 5$. We generate $n = 1000$ synthetic data $\{\boldsymbol{x}_i\}$ in the following way: each data point is given as $\boldsymbol{x}_i = \boldsymbol{A}\boldsymbol{z}_i + \boldsymbol{\epsilon}_i$, where $\boldsymbol{A} \in \mathbb{R}^{d \times k}$ is a fixed matrix with entries sampled as $\boldsymbol{A}_{kl} \sim \mathcal{U}([0,1])$, $\boldsymbol{z}_i \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{\Lambda} := \operatorname{diag}\left\{4, 2, 1, \frac{1}{2}, \frac{1}{4}\right\}\right)$ and the noise $\boldsymbol{\epsilon}_i \sim 10^{-3}\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_d)$. We initialize the weight matrices to a close-to-zero initialization $\boldsymbol{W}_1^{(0)}, \boldsymbol{W}_2^{(0)}, \boldsymbol{W}_3^{(0)} \sim 10^{-5}\mathcal{U}([0,1])$: this will guarantees the FA dynamics to avoid the

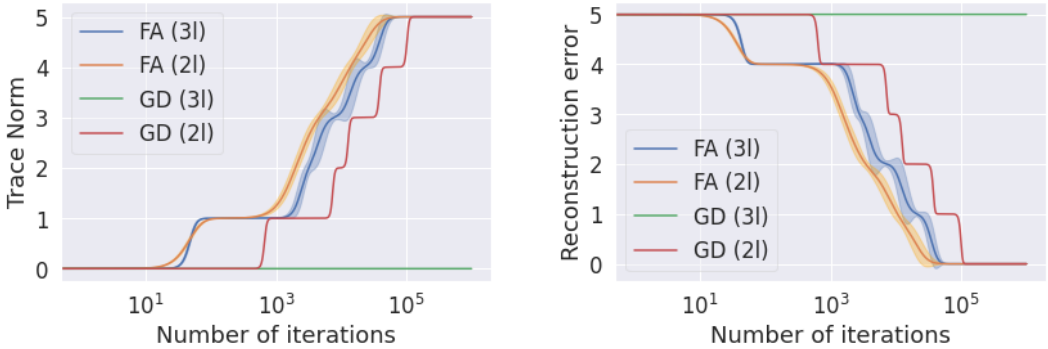


Figure 2: Comparison of trace norms and reconstruction errors for linear autoencoders. For the FA experiments, we are plotting the average (with error bars) of 15 experiments.

anti-regularization pattern and to converge to the signal (Theorems 8, 9 and 10).
We generate the FA matrices $\boldsymbol{M}$ (2-layers) and $\boldsymbol{M}_1, \boldsymbol{M}_2$ (3-layers) with entries $\boldsymbol{M}_{ij} \sim \mathcal{U}([0,1])$. Keeping the same initial conditions $\boldsymbol{W}_\ell^{(0)}$ and same fixed matrix $\boldsymbol{A}$, we repeat the FA training 15 times, sampling a different set of FA matrices each time: in Figure 2, we report the average (with error bars) of the trace norm $\|\boldsymbol{\Theta}^{(t)}\|_*$ with $\boldsymbol{\Theta}^{(t)} := \boldsymbol{W}_2^{(t)}\boldsymbol{W}_1^{(t)}$ or $\boldsymbol{\Theta}^{(t)} := \boldsymbol{W}_3^{(t)}\boldsymbol{W}_2^{(t)}\boldsymbol{W}_1^{(t)}$ respectively, as well as thee average of the reconstruction errors $\left\|\boldsymbol{\Theta}^{(t)} - \boldsymbol{A}\boldsymbol{\Lambda}\boldsymbol{A}^\top\right\|_2$, as a function of t the number of iterations.

We can see that the FA learning dynamics is distinctly faster than GD, which is learning at a much slower pace, as we initialized the dynamics close to one of GD's local maxima. In particular, the 3-layer NN trained with GD hasn't yet started to learn at the end of the selected training period of the experiment. Notably, the FA training for the 3-layer NN hints at the presence of an implicit regularization behavior, similar to what has been observed and proved in Gidel et al. (2019) for a 2-layer NN. Additional technical details can be found in Appendix E.

7 Final discussions and insights

In the present paper we rigorously analyze a biologically plausible optimization algorithm, we describe its potential as well as its limitations in possible ML applications. We provide convergence guarantees in both the continuous and discrete-time settings; additionally, we thoroughly discuss implicit regularization phenomena that may arise with certain initialization schemes and that may deeply affect the optimization effectiveness. The occurrence of drastically opposite regularization by only modifying an algorithm's initialization showcases the critical importance of the latter for ML applications.

Although the analysis is conducted only for linear models, the results give already numerous insights on the behaviour of nonlinear architectures. The analysis of Theorem 5 shows that, in some regimes, the deeper layers behave as "a power" of the first layer: this result suggests that each layer is converging at a different speed. It thus defines a hierarchy (in terms of speed of convergence) between the layers of the network. Furthermore, one can notice that in the linear case FA "removes" some stationary points: unlike GD that potentially has many stationary points that are saddle points, all the stationary points of FA are global minima. Therefore, we believe that FA will also "remove" some undesirable stationary points in the non-linear case. Several challenges remain, but the above pieces of intuition will be critical for the analysis in the nonlinear setting.

Beyond this, it is enticing to conduct a comparison between FA and GD and illustrate the advantages of one algorithm over the other (and in which setting). All of these directions will be the central objects of study in future works.

REFERENCES

Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. In *International Conference on Learning Representations*, 2018.

Sergey Bartunov, Adam Santoro, Blake Richards, Luke Marris, Geoffrey E. Hinton, and Timothy Lillicrap. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Tien Chu, Kamil Mykitiuk, Miron Szewczyk, Adam Wiktor, and Zbigniew Wojna. Training DNNs in O(1) memory with MEM-DFA using random matrices. *arXiv:2012.11745*, 2020.

Francis Crick. The recent excitement about neural networks. *Nature*, 337, 1989.

Charlotte Frenkel, Martin Lefebvre, and David Bol. Learning without feedback: Fixed random learning signals allow for feedforward training of deep neural networks. *Frontiers in Neuroscience*, 15, 2021.

Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. *arXiv:1904.13262*, 2019.

J Gilmer, C. Raffel, S. S. Schoenholz, M. Raghu, and J. Sohl-Dickstein. Explaining the learning dynamics of direct feedback alignment. *ICLR*, 2017.

Daniel Gissin, Shai Shalev-Shwartz, and Amit Daniely. The implicit bias of depth: How incremental learning drives generalization. *arXiv:1909.12051*, 2019.

Stephen Grossberg. Competitive learning: from interactive activation to adaptive resonance. *Cogn. Sci.*, 11, 1987.

Arieh Iserles. *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, 1996.

Julien Launay, Iacopo Poli, and Florent Krzakala. Principled training of neural networks with direct feedback alignment. *arXiv:1906.04554*, 2019.

Julien Launay, Iacopo Poli, François Boniface, and Florent Krzakala. Direct feedback alignment scales to modern deep learning tasks and architectures. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

J. Lee, R. Zhang, W. Zhang, Y. Liu, and P. Li. Spike-Train level Direct Feedback Alignment: Sidestepping backpropagation for on-chip training of spiking neural nets. *Front Neurosci.*, 2020.

Timothy P. Lillicrap, Daniel Cownden, Douglas B. Tweed, and Colin J. Akerman. Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, 7, 2016.

Theodore H. Moskovitz, Ashok Litwin-Kumar, and L.F. Abbott. Feedback alignment in deep convolutional networks. *arXiv:1812.06488*, 2018.

A. Nøkland. Direct feedback alignment provides learning in deep neural networks. *arXiv:1609.01596*, 2016.

C. C. Paige and M. A. Saunders. Towards a generalized singular value decomposition. *SIAM Journal on Numerical Analysis*, 18(3):398–405, 1981.

Maria Refinetti, Stéphane d'Ascoli, Ruben Ohana, and Sebastian Goldt. The dynamics of learning with feedback alignment. *arXiv:2011.12428*, 2020.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323, 1986.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. A mathematical theory of semantic development in deep neural networks. *arXiv:1810.10531*, 2018.

Charles F. van Loan. Generalizing the singular value decomposition. *SIAM Journal on Numerical Analysis*, 13(1):76–83, 1976.