Diagonalizing the Softmax: Hadamard Initialization for Tractable Cross-Entropy Dynamics

author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2026

Abstract

In this work, we study cross-entropy (CE) dynamics using a two-layer linear network with orthogonal inputs, the simplest non-convex setting where the CE implicit bias remains unresolved. This coincides with the unconstrained features model used to study neural collapse (NC), a phenomenon occurring in deep classification networks. Our analysis is based on a key observation: Hadamard initialization diagonalizes the softmax operator. This allows us to extend the spectral initialization framework that Saxe et al. [21, 22] developed for squared loss. We prove convergence to NC under spectral CE training and give the first finite-time analysis in this setting via an explicit Lyapunov function that decreases monotonically to NC. We further identify CE-specific phenomena absent under squared loss and show empirically that spectral dynamics model small random initialization.

1. Introduction

The training dynamics of linear networks under mean squared error (MSE) loss have provided a tractable setting for understanding phenomena in deep learning [1, 2, 9, 16, 24]. A central example is the spectral-dynamics framework of Saxe et al. [21, 22], where gradient flow reduces to decoupled dynamics over singular values. This gives a rare trajectory-level theory: it describes not only the limiting solution selected by optimization, but also the full path taken during training.

For classification with cross-entropy loss, no comparable trajectory-level theory is known. The main obstruction is the softmax nonlinearity, which couples singular modes and prevents the spectral reductions that make squared-loss dynamics tractable. This gap is especially important for understanding neural collapse (NC) [20]: the empirical phenomenon in which class means and last-layer weights converge to a simplex equiangular tight frame. Although several works characterize limiting solutions, the mechanism by which gradient-based optimization selects this geometry over the course of training remains poorly understood.

We give the first *trajectory-level* analysis of multiclass CE training dynamics for the CE-UFM, a model popular in the NC literature because it recovers phenomena observed in deep systems [7, 14, 26]. This model also represents the simplest non-convex model for which the implicit bias remained unknown: a two-layer linear network with orthogonal inputs.

- **Softmax diagonalization.** We show that Hadamard initialization diagonalizes the softmax nonlinearity. This enables us to extend the framework of Saxe et al. [21, 22] from MSE to CE, reducing gradient flow to a vector ODE over the singular values of the logit matrix that is sufficiently structured for a study of gradient flow trajectories.
- **Non-monotonicity.** Unlike MSE, CE can exhibit non-monotonic behavior that depends on the number of classes K and that is invisible to asymptotic analyses.

- **Guaranteed convergence to NC.** Despite the non-monotonicity of natural metrics and the non-convex landscape, we prove that gradient flow *always* converges to NC under Hadamard initialization. Unlike asymptotic arguments, we provide an explicit Lyapunov function that decreases monotonically along the full trajectory.

These Hadamard trajectories are genuine gradient-flow paths, so these phenomena are compatible with CE training. Empirically, we show these trajectories also model small random initialization.

Related work. Many works study the emergence of NC through the UFM [19]. For CE, existing analyses rely on explicit L_2 regularization [6, 26], or establish convergence only to a KKT point [11]. Previous implicit bias results have shown that logistic regression converges to the max-margin solution [23]. Subsequent work has extended this to homogeneous networks [12, 13, 17] through KKT characterizations. For spectral dynamics, Saxe et al. [21] showed that deep linear MSE networks can admit tractable explicit dynamics over singular modes. This framework has since been extended to studies of emergence [22], training [3, 5, 15], optimization [8], and linear attention [18].

2. Background

Consider K-class classification with n samples per class. Let $h_{ic} \in \mathbb{R}^d$ denote the embedding, parameterized by a deep network, of sample $i \in [n]$ from class $c \in [K]$. Stack these by class to form the embedding matrix $H \in \mathbb{R}^{d \times Kn}$, and let $W \in \mathbb{R}^{K \times d}$ denote the last layer classifier head.

Neural collapse (NC) [20] is the empirical observation that $H^\top H \propto S \otimes 1_n 1_n^\top$ and $WW^\top \propto S$, where $\otimes$ is the Kronecker product and $S = I_K - \frac{1}{K} 1_K 1_K^\top \in \mathbb{R}^{K \times K}$ is the simplex ETF matrix.

Unconstrained Features Model. The UFM approximates an expressive deep network by treating the embedding h_{ic} of the data as freely optimized vectors. Under CE loss, the UFM minimizes:

$$\mathcal{L}(W, H) = \sum_{c \in [K], i \in [n]} \log\left(1 + \sum\nolimits_{c' \neq c} e^{(Wh_{ic})_{c'} - (Wh_{ic})_c}\right). \tag{1}$$

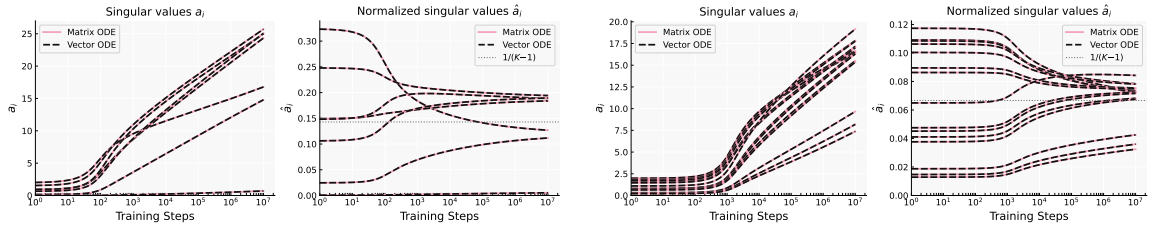
This can be viewed as a two-layer linear network with weights W and H, and input data the standard basis vectors in $\mathbb{R}^{Kn}$. See App. A of Garrod et al. [7] for discussion on the UFM as a modeling tool.

Let $Y = I_K \otimes 1_n^\top \in \mathbb{R}^{K \times nK}$ denote the one-hot label matrix. Define $P(Z)$ to be the *softmax* operation applied to the logit matrix $Z = WH$: i.e., $\left[P(Z)\right]_{ij} = \exp(Z_{ij}) \,/\, \sum_{k=1}^K \exp(Z_{kj})$. Under gradient flow on the CE loss in Eq. (1), the parameters evolve according to

$$\frac{dW}{dt} = (Y - P(Z))H^T, \quad \frac{dH}{dt} = W^T(Y - P(Z)). \tag{2}$$

While Ji et al. [11] showed that this gradient flow converges directionally to a KKT point of a max-margin problem, gradient flow can converge to KKT points that are not even local minima of the max-margin problem [25]. Thus, the question is open: *Does gradient flow* (2) *converge to NC?*

Exact dynamics with MSE loss. Let the SVD of Y be $Y = U\Sigma V^T$, with singular values s_c for $c \in [K]$. Consider spectral initialization: $W = UD_W R^T, H = RD_H V^T$, for orthogonal $R \in \mathbb{R}^{d \times d}$, with D_W and D_H having their only non-zero diagonal entries being $\alpha_1, ..., \alpha_K$ and $\beta_1, ..., \beta_K$. Then, at initialization, the label matrix Y and the logit matrix WH are mutually diagonalizable. Saxe et al. [21, 22] show for MSE loss this implies that $\frac{dW}{dt}, \frac{dH}{dt}$ have the same singular vectors as W, H. Thus, only the singular values of W and H evolve, and can be shown to monotonically approach s_i along sigmoidal trajectories. Saxe et al. [21] further show empirically that small random initialization has the same characteristic behavior as these analytic paths. *Can this framework extend to CE loss?*



(a) $K = 8$ (b) $K = 16$

Figure 1: Illustration of the Hadamard reduction for $K = 8, 16$. The trajectories of the full matrix dynamics (Eq. (2)) and the vector ODE (Eq. (3)) are indistinguishable, verifying Thms. 2 and 3.

3. Hadamard initialization

The obstacle in extending spectral dynamics from MSE to CE is the softmax nonlinearity: for CE we must simultaneously diagonalize Y, WH and $P(WH)$, which is difficult since nonlinearities rarely preserve spectral structure. This obstacle can be overcome by using Hadamard initialization.

Definition 1 *The **Sylvester Hadamard matrices** $\{\Phi_{2^m} : m \in \mathbb{N}\}$ are defined recursively by*

$$\Phi_1 = 1, \quad \Phi_{2^m} = \begin{bmatrix} \Phi_{2^{m-1}} & \Phi_{2^{m-1}} \\ \Phi_{2^{m-1}} & -\Phi_{2^{m-1}} \end{bmatrix} \in \mathbb{R}^{2^m \times 2^m},$$

and are orthogonal $\Phi_{2^m}^T \Phi_{2^m} = 2^m I_{2^m}$.

We first illustrate our technique for $n = 1$ (one data point per class), then extend to general n. We show that setting $U = \frac{1}{\sqrt{K}}\Phi$ achieves mutual diagonalizability despite the softmax.

Theorem 2 *For $K = 2^m, m \in \mathbb{N}$, let $U = \frac{1}{\sqrt{K}}\Phi$, where Φ is the $K \times K$ Sylvester Hadamard matrix. Denote the columns of U by u_i, $i \in [K]$. Then, for any $Z = \sum_i a_i u_i u_i^T$, its softmax satisfies*

$$P(Z) = \sum_i \nu_i u_i u_i^T, \quad \text{where } \nu_i = \frac{\sum_j \Phi_{ij} e^{\frac{1}{K}(\Phi a)_j}}{\sum_j e^{\frac{1}{K}(\Phi a)_j}}.$$

We refer to this initialization as *Hadamard initialization*. This shows that a matrix diagonalizable by a Sylvester–Hadamard matrix remains so after applying softmax. See App. C.1 for a detailed proof.

3.1. Tractable cross-entropy dynamics

Using Theorem 2, we can adapt the arguments of Saxe et al. [21], detailed in Sec. 2. This property is specific to our construction: a generic orthogonal U does not yield mutual diagonalizability. The result is given for general n below. The proof is in App. C.2, with validation in Fig. 1.

Theorem 3 *For $K = 2^m, m \in \mathbb{N}$, let $U = \frac{1}{\sqrt{K}}\Phi$. For the UFM in Eq. (1), with classes of size n, initialize the parameters as $W = UD_W R^T$, and $H = RD_H V^T$, where $R \in \mathbb{R}^{d \times d}$ is orthogonal, $V = U \otimes Q$ where Q is a right singular matrix of 1_n^T, and the only nonzero singular values of D_W and D_H are $\alpha_0, \ldots, \alpha_{K-1}$ and $\beta_0, \ldots, \beta_{K-1}$. Then under balancedness, and absorbing constants into the time and singular value variables, the gradient flow equations reduce to*

$$\frac{da_i}{dt} = \frac{1}{D} b_i a_i \quad i \in [K-1], \quad \frac{da_0}{dt} = 0, \tag{3}$$

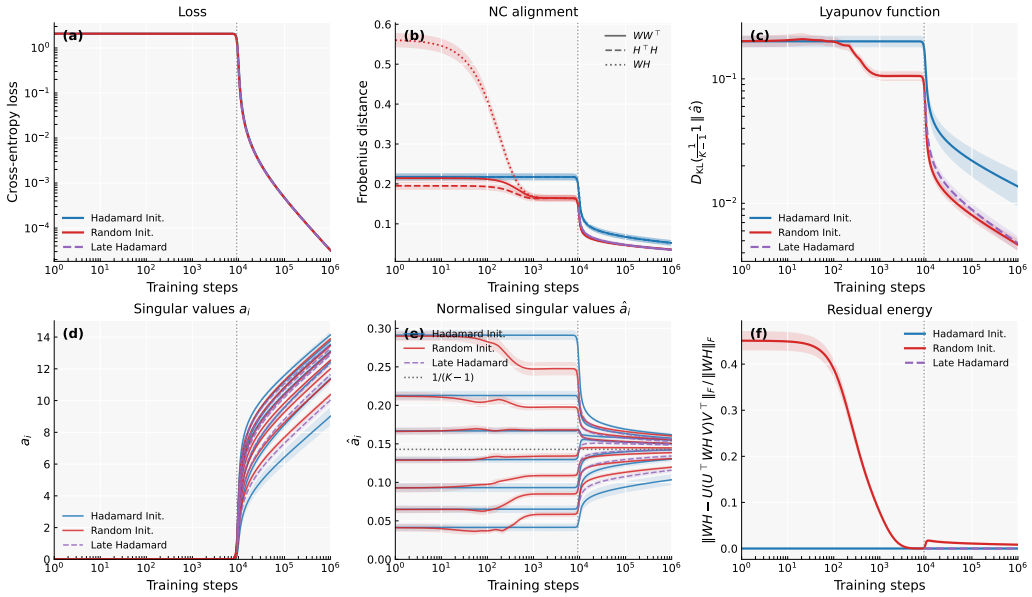


Figure 2: Gradient flow under small random initialization, Hadamard initialization with the same singular values, and late-Hadamard initialized at the end of the alignment phase (vertical dotted line). Parameters $K = 8, d = 40$. mean and std reported over ten runs.

$$b_i = \sum_{j=1}^{K-1} \Psi_{ij} e^{-(\Psi a)_j} \;\; \textit{and} \;\; D = 1 + \sum_{j=1}^{K-1} e^{-(\Psi a)_j} \tag{4}$$

where $a_i = \alpha_i \beta_i$ are the logit singular values, and $\Psi = 1_{K-1} 1_{K-1}^T - \Phi[2:K, 2:K] \in \mathbb{R}^{K-1 \times K-1}$.

Since the first singular value is constant, we drop it from notation and consider $a \in \mathbb{R}^{K-1}$. Unlike for MSE, the terms b_i/D depend on all of the singular values, preventing decoupling, and consequently we do not have closed-form dynamics. Nevertheless, we show below that the equations are sufficiently tractable. Also note that the singular values $a_i(t)$ increase in time. We henceforth assume all singular values are positively initialized, and thus stay positive for all times.

Normalized singular values. From Eq. (3), it is straightforward to show that the L_1 logit norm diverges. This implies that there are no finite minima on the optimization surface. We therefore study the dynamics of the *normalized logit singular values* $\hat{a}_i = a_i / \|a\|_1$:

$$\frac{d}{dt}(\hat{a}_i) = \frac{\hat{a}_i}{D} \left[b_i - \bar{b} \right], \quad \bar{b} = \sum\nolimits_{j \in [K-1]} \hat{a}_j b_j. \tag{5}$$

4. Cross-entropy dynamics and implicit bias

Any direction $\hat{a}$ satisfying $(\Psi \hat{a})_i > 0$ for all $i \in [K-1]$ will have positive margins and so can attain arbitrarily small loss by taking $\|a\|_1 \to \infty$. The question is then which of these directions gradient flow selects. We begin by inspecting the *stable directions*, i.e., those satisfying $\frac{\mathrm{d}\hat{a}}{\mathrm{d}t} = 0$. If a stable direction satisfies $\hat{a}_i > 0$ for all i, then the non-trivial part of the logit matrix is full-rank. It is simple to show the unique full-rank stable direction is $\hat{a} = \frac{1}{K-1} 1_{K-1}$. This in fact corresponds to NC, as shown in App. B.

Convergence to NC via a Lyapunov function. We show that gradient flow exhibits an implicit preference for the full-rank NC solution among all other stable points. We further establish a stronger result than asymptotic convergence: we identify a Lyapunov function, measuring distance to NC, that decreases monotonically along the gradient-flow trajectory. The proof is in App. C.3.

Theorem 4 *Let $K = 2^m, m \in \mathbb{N}$. Let $a(t) \in \mathbb{R}^{K-1}$ be initialized with $a_i(0) > 0$ for all i, that evolves under the time evolution of Eq. (3). Then the metric $D_{\mathrm{KL}}(\frac{1}{K-1}1_{K-1} \,\|\, \hat{a})$, where D_{KL} is the Kullback-Leibler divergence, serves as a Lyapunov function of the flow. Consequently, the metric is decreasing throughout training and we must have $\hat{a} \to 1/(K-1) \cdot 1_{K-1}$ as $t \to \infty$.*

This theorem goes beyond existing asymptotic implicit bias results by showing behavior throughout the full loss surface via a Lyapunov function. Regarding NC, Theorem 4 is the first result proving that NC occurs purely as a result of gradient flow implicit bias under CE loss.

Non-monotonic behaviors. We now show that the L_2-distance metric, which is common in empirical work on NC, does not exhibit monotonic behavior.

$$M = \frac{1}{2}\sum\nolimits_{i\in[K-1]} \left(\hat{a}_i - \frac{1}{K-1}\right)^2. \tag{6}$$

For $K = 2, 4$, this metric decreases monotonically; however, monotonicity breaks down for $K \geq 8$.

Theorem 5 *For $K = 2^m, m \in \mathbb{N}$, let $a(t) \in \mathbb{R}^{K-1}$ be a variable initialized with $a(0) > 0$, that evolves as in Eq. (3). If $K = 2$ or $K = 4$, then the metric M in Eq. (6) monotonically converges to 0. If $K \geq 8$, then there exist initializations $a(0)$ such that the metric M is not monotonic.*

This reveals a qualitative difference from MSE: coupled singular values can induce non-monotonic behavior in natural distance metrics. The proof is in App. C.4, and an example in App. D.

5. Numerical experiments

We now test whether these trajectories capture the qualitative behavior of standard small-random initialization. Fig. 2 reports, for both random and Hadamard initialization, the metrics studied in the preceding sections, together with the residual energy defined in App. D, which measures how much the singular vectors of the logits are aligned with the Hadamard basis. Random trajectories exhibit a two-stage structure. First, during an alignment phase, the residual energy decreases while other metrics are mostly static. After this phase, both trajectories exhibit comparable trends. Fig. 2 also shows a late-Hadamard trajectory initialized after the alignment phase, using singular values matched to those of the random trajectory. This late-Hadamard trajectory provides a representative trajectory of small random initialization after the alignment phase. Additional experiments are in App. D.

6. Conclusion and future work

In this work, we uncover a variety of novel features in CE training by adapting the spectral-dynamics framework of Saxe et al. to the softmax setting. We view demonstrating that such an extension is achievable as our central contribution. Next steps include extending beyond $K = 2^m$ using Fourier matrices, as well as extending to more general initializations. Notably, Theorem 2 extends to any elementwise activation function, opening the door to spectral analyses in nonlinear architectures.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

[1] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. *arXiv preprint arXiv:1810.02281*, 2018.

[2] Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 244–253. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/arora18a.html.

[3] Lukas Braun, Clémentine Dominé, James Fitzgerald, and Andrew Saxe. Exact learning dynamics of deep linear networks with prior knowledge. In *Advances in Neural Information Processing Systems*, volume 35, pages 6615–6629. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/2b3bb2c95195130977a51b3bb251c40a-Paper-Conference.pdf.

[4] Nadav Cohen and Noam Razin. Lecture notes on linear neural networks: A tale of optimization and generalization in deep learning. *arXiv preprint arXiv:2408.13767*, 2024.

[5] Cl'ementine Carla Juliette Domin'e, Nicolas Anguita, Alexandra M Proca, Lukas Braun, Daniel Kunin, Pedro A. M. Mediano, and Andrew Saxe. From lazy to rich: Exact learning dynamics in deep linear networks. *ArXiv*, abs/2409.14623, 2024. URL https://api.semanticscholar.org/CorpusID:272826705.

[6] Connall Garrod and Jonathan P. Keating. The persistence of neural collapse despite low-rank bias: An analytic perspective through unconstrained features. *ArXiv*, abs/2410.23169, 2024. URL https://api.semanticscholar.org/CorpusID:273695529.

[7] Connall Garrod and Jonathan P Keating. Unifying low dimensional observations in deep learning through the deep linear unconstrained feature model. *arXiv preprint arXiv:2404.06106*, 2024.

[8] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/f39ae9ff3a81f499230c4126e01f421b-Paper.pdf.

[9] Suriya Gunasekar, Blake E. Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. *2018 Information Theory and Applications Workshop (ITA)*, pages 1–10, 2017. URL https://api.semanticscholar.org/CorpusID:3909231.

[10] Kathy J Horadam. *Hadamard matrices and their applications*. Princeton university press, 2012.

[11] Wenlong Ji, Yiping Lu, Yiliang Zhang, Zhun Deng, and Weijie J Su. An unconstrained layer-peeled perspective on neural collapse. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=WZ3yjh8coDg.

[12] Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJflg30qKX.

[13] Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 17176–17186. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c76e4b2fa54f8506719a5c0dc14c2eb9-Paper.pdf.

[14] Vignesh Kothapalli, Ebrahim Rasromani, and Vasudev Awatramani. Neural collapse: A review on modelling principles and generalization. *Trans. Mach. Learn. Res.*, 2023, 2022. URL https://api.semanticscholar.org/CorpusID:249461767.

[15] Daniel Kunin, Allan Ravent'os, Cl'ementine Carla Juliette Domin'e, Feng Chen, David Klindt, Andrew Saxe, and Surya Ganguli. Get rich quick: exact solutions reveal how unbalanced initializations promote rapid feature learning. *ArXiv*, abs/2406.06158, 2024. URL https://api.semanticscholar.org/CorpusID:270371640.

[16] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. *arXiv preprint arXiv:2012.09839*, 2020.

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

[18] Nischal Mainali and Lucas Teixeira. Exact learning dynamics of in-context learning in linear transformers and its application to non-linear transformers. *ArXiv*, abs/2504.12916, 2025. URL https://api.semanticscholar.org/CorpusID:277857272.

[19] Dustin G. Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *Sampling Theory, Signal Processing, and Data Analysis*, 20, 2020. URL https://api.semanticscholar.org/CorpusID:227127383.

[20] Vardan Papyan, Xuemei Han, and David L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences of the United States of America*, 117:24652 – 24663, 2020. URL https://api.semanticscholar.org/CorpusID:221172897.

[21] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *CoRR*, abs/1312.6120, 2013. URL https://api.semanticscholar.org/CorpusID:17272965.

[22] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019. doi: 10.1073/pnas.1820226116. URL https://www.pnas.org/doi/abs/10.1073/pnas.1820226116.

[23] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19 (70):1–57, 2018. URL http://jmlr.org/papers/v19/18-188.html.

[24] Zhenfeng Tu, Santiago Aranguri, and Arthur Jacot. Mixed dynamics in linear networks: Unifying the lazy and active regimes. In *Advances in Neural Information Processing Systems*, volume 37, pages 106059–106104. Curran Associates, Inc., 2024.

[25] Gal Vardi, Ohad Shamir, and Nati Srebro. On margin maximization in linear and relu networks. *Advances in Neural Information Processing Systems*, 35:37024–37036, 2022.

[26] Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

Appendix A. Summary of notations

Table 1: Summary of Key Notations

Notation	Description	Context / Definition
K	Number of classes.	Assumed to be $K = 2^m$.
n	Number of samples per class.	Assumed to be balanced.
W	Last-layer weight matrix.	$\mathbb{R}^{K\times d}$
H	Feature matrix.	$\mathbb{R}^{d\times Kn}$
Z	Logit matrix.	$Z = WH$
Y	One-hot label matrix.	$Y = I_K \otimes 1_n^T$
$P(Z)$	Softmax matrix.	Softmax applied columnwise to Z.
S	Simplex ETF matrix.	$S = I_K - \frac{1}{K}1_K 1_K^T$
Φ	Sylvester Hadamard matrix.	$\mathbb{R}^{K\times K}$ matrix (Def. 1).
Ψ	Reduced dynamics matrix.	$1_{K-1}1_{K-1}^T - \Phi[2:K, 2:K]$.
U, V	Left/Right singular vector matrices.	$U = V = \frac{1}{\sqrt{K}}\Phi$ for Hadamard Init.
α_i, β_i	Singular values.	Non-trivial singular values of W and H.
a	Logit singular values (vector).	$a = (a_1, \ldots, a_{K-1})$, $a_i = \alpha_i\beta_i$.
$\hat{a}$	Normalized logit singular values.	$\hat{a}_i = a_i/\|a\|_1$.
b, D	Terms in reduced dynamics.	Defined in Eq. 3.
M	L_2-based metric for NC.	Measures distance from $\hat{a}$ to uniform (Eq. 6).
D_{KL}	KL divergence.	Used as a Lyapunov function (Thm. 4).

Throughout, $\mathbb{R}$ denotes the reals and $\mathbb{N}$ the natural numbers. For a positive integer j, we write $[j] = \{1, \ldots, j\}$, and $e_\ell \in \mathbb{R}^j$ denotes the ℓ-th standard basis vector. I_k represents the $k \times k$ identity matrix, and 1_k denotes an all-ones vector of dimension k. For a given matrix A, A^T is its transpose, and $A[i_1 : i_2, j_1 : j_2]$ indicates the submatrix sliced from rows i_1 to i_2 and columns j_1 to j_2. The operator $\mathrm{diag}(v)$ forms a diagonal matrix using the elements of vector v. We use $\otimes$ for the Kronecker product, $\odot$ for the Hadamard (elementwise) product and $\oplus$ for the bitwise XOR. Finally, $\|\cdot\|_F$, and $\|\cdot\|_1$ denote the Frobenius norm and the L_1 norm, respectively. Some key notation is summarized in Table 1.

Appendix B. Uniform singular values corresponds to neural collapse

To see why uniformity corresponds to the NC solution, note that equal logit singular values (i.e., $a_i = a_j$ for all $i, j \in [K-1]$) imply that the singular values of W and H are equal as well: $\alpha_i = \beta_i = \alpha_j = \beta_j$ for all $i, j \in [K-1]$. Recall that, without loss of generality, the singular value corresponding to the all-ones eigenspace is zero for both matrices. Together, these conditions yield $W \propto \Phi\,\mathrm{diag}(0, 1, \ldots, 1)R^T$ and $H \propto R\,\mathrm{diag}(0, 1, \ldots, 1)\Phi^T$, which in turn give $WW^T \propto S$ and $H^TH \propto S$, where $S = \frac{1}{K}\Phi\,\mathrm{diag}(0, 1, \ldots, 1)\Phi^T$ is the ETF matrix.

Appendix C. Proofs

C.1. Proof of Theorem 2

Let ϕ_i, $i \in \{1, 2, \ldots, K\}$, be the rows of the $K \times K$ Sylvester–Hadamard matrix Φ. We use the following "closure" property (Lemma 3.6 of Horadam [10]), where $i \oplus j$ denotes bitwise XOR:

Fact 1 *For any $i, j \in \{1, \ldots, K\}$, it holds that $\phi_i \odot \phi_j = \phi_{i \oplus j}$.*

For all $i \in [K]$, define matrices $\Phi_i := \Phi \operatorname{diag}(\phi_i) = \left[\phi_j^T \odot \phi_i^T\right]_{j \in [K]}$. Note that $\Phi_1 = \Phi$. Using Fact 1, we can express each Φ_i in terms of a corresponding permutation matrix $\Pi_i = \left[e_{i \oplus j}^T\right]_{j \in [K]}$, where $e_\ell \in \mathbb{R}^K$ denotes the ℓ-th standard basis vector:

$$\Phi_i := \Phi \operatorname{diag}(\phi_i) = \Pi_i \Phi, \qquad \text{for all } i \in [K]. \tag{7}$$

Recall that softmax is applied columnwise. Thus, to prove the advertised identity it suffices for any column $i \in [K]$ that

$$P\left(\frac{1}{K} \Phi \operatorname{diag}(a)\, \phi_i\right) = \frac{1}{K} \Phi \operatorname{diag}\left(\Phi P\left(\frac{1}{K} \Phi a\right)\right) \phi_i \,.$$

Using the property $\operatorname{diag}(u) \cdot v = \operatorname{diag}(v) \cdot u$ for any u, v on both sides of the equality, and applying Eq. (7), this is equivalent to

$$P\left(\frac{1}{K} \Pi_i \Phi a\right) = \frac{1}{K} \Pi_i \Phi \Phi P\left(\frac{1}{K} \Phi a\right).$$

Since $\Phi\Phi = \Phi\Phi^T = K I_K$, this claim reduces to $P(\frac{1}{K} \Pi_i \Phi a) = \Pi_i P(\frac{1}{K} \Phi a)$, which holds by permutation equivariance of the softmax.

C.2. Proof of Theorem 3

Recall $U = \frac{1}{\sqrt{K}} \Phi$, where Φ is the $K \times K$ Sylvester Hadamard matrix, and that we initialize our parameter matrices as: $W = U D_W R^T$, $H = R D_H V^T$, where $R \in \mathbb{R}^{d \times d}$ is orthogonal, and $V = U \otimes Q$ is orthogonal, with Q being a right singular matrix of 1_n^T, meaning it obeys the relation: $1_n^T = \sqrt{n} e_1^{(n)T} Q^T$, where $e_1^{(n)}$ is the first standard basis vector in $\mathbb{R}^n$. Also recall that $D_W \in \mathbb{R}^{K \times d}$ has its only non-zero singular values given by $\alpha_1, \ldots \alpha_K$, and $D_H \in \mathbb{R}^{d \times Kn}$ has its only non-zero singular values given by $\beta_1, \ldots, \beta_K$. Note as a consequence of the definitions of U, V, we can write $Y = \sqrt{n} U [I_K, 0_{K \times K(n-1)}] V^T$.

We also have at initialization that the logit matrix is given by

$$Z = U D_W D_H V^T.$$

Note that $D_W \in \mathbb{R}^{K \times d}$ and $D_H \in \mathbb{R}^{d \times Kn}$. Hence, defining $D_Z = D_W D_H \in \mathbb{R}^{K \times Kn}$, and $D_a = \operatorname{diag}(a_1, ..., a_K)$, where $a_i = \alpha_i \beta_i$, this matrix can be written as:

$$D_Z = [D_a, 0_{K \times K(n-1)}] = D_a \otimes e_1^{(n)T}.$$

Hence we can write

$$Z = U \left(D_a \otimes e_1^{(n)T}\right) (U \otimes Q)^T = U \left[(D_a U^T) \otimes (e_1^{(n)T} Q^T)\right].$$

Next, using that $1_n^T = \sqrt{n} e_1^{(n)T} Q^T$, this becomes:

$$Z = \frac{1}{\sqrt{n}} U \left[(D_a U^T) \otimes 1_n^T \right].$$

Then using that this has repeated columns, this becomes:

$$Z = (U D_{\tilde{a}} U^T) \otimes 1_n^T,$$

where we define $\tilde{a} = \frac{1}{\sqrt{n}} a$. Since this is the same matrix as $U D_{\tilde{a}} U^T$, just with repeated columns, we can apply Theorem 2 to calculate the softmax matrix at initialization:

$$P(Z) = (U D_{\tilde{\nu}} U^T) \otimes 1_n^T,$$

where

$$\tilde{\nu}_i = \frac{\sum_j \Phi_{ij} e^{\frac{1}{K}(\Phi \tilde{a})_j}}{\sum_j e^{\frac{1}{K}(\Phi \tilde{a})_j}}.$$

Now consider the gradient flow equations of our model

$$\frac{dW}{dt} = (Y - P) H^T, \qquad \frac{dH}{dt} = W^T (Y - P).$$

Consider the change of variable given by $W = U\tilde{W}$, $H = \tilde{H} V^T$, these have evolution equations given by:

$$\frac{d}{dt}(\tilde{W}) = U^T (Y - P) V \tilde{H}^T, \qquad \frac{d}{dt}(\tilde{H}) = \tilde{W}^T U^T (Y - P) V.$$

Note that, by construction, both the matrices Y, P can be written in SVD form through the matrices U, V

$$Y = \sqrt{n} U D_Y V^T, \qquad P = \sqrt{n} U D_P V^T,$$

where $D_Y = [I_K, 0_{K \times K(n-1)}] \in \mathbb{R}^{K \times Kn}$ and $D_P = [\text{diag}(\tilde{\nu}), 0_{K \times K(n-1)}] \in \mathbb{R}^{K \times Kn}$, and so

$$U^T (Y - P) V = \sqrt{n} [\text{diag}(1_K - \tilde{\nu}), 0_{K \times K(n-1)}],$$

label this matrix as B. Now returning to the evolution equations for $\tilde{H}$ and $\tilde{W}$, we label the rows of $\tilde{W}$ by $\tilde{w}_a$, and the columns of $\tilde{H}$ by $\tilde{h}_a$. These have evolution equations given by:

$$\frac{d}{dt}((\tilde{w}_i)_j) = \sum_x B_{ix} (\tilde{h}_x)_j, \qquad \frac{d}{dt}((\tilde{h}_j)_i) = \sum_x B_{xj} (\tilde{w}_x)_i.$$

Now using that $B_{ix} = \sqrt{n}(1 - \tilde{\nu}_i)\delta_{ix}$ gives:

$$\frac{d}{dt}((\tilde{w}_i)_j) = \sqrt{n}(1 - \tilde{\nu}_i)(\tilde{h}_i)_j, \qquad \frac{d}{dt}((\tilde{h}_j)_i) = \sqrt{n}(1 - \tilde{\nu}_j)(\tilde{w}_j)_i, \tag{8}$$

where in the second of these equations if $j > K$ the derivative is zero.

Note that using our initialization scheme, we have at initialization $\tilde{w}_i = \alpha_i r_i$, $\tilde{h}_i = \beta_i r_i$, for $i = 1, \ldots, K$, and 0 otherwise, where r_i are the columns of R. One observes from Eq. (8) that the derivatives also point in the same direction. Hence this property is maintained at all times, meaning

the vectors $\tilde{w}_i, \tilde{h}_i$ only evolve in their coefficients, not in their directions. These coefficients then evolve as:

$$\frac{d}{dt}(\alpha_i) = \sqrt{n}(1-\tilde{\nu}_i)\beta_i, \qquad \frac{d}{dt}(\beta_i) = \sqrt{n}(1-\tilde{\nu}_i)\alpha_i.$$

Translating back to our original parameter matrices W, H, these are the evolution equations for their singular values, and the singular vectors are frozen for all times.

It remains to demonstrate these equations take the exact form in the theorem statement. Note from the expression of $\tilde{\nu}_i$ we have

$$1-\tilde{\nu}_i = \frac{1}{\sum_j e^{\frac{1}{K}(\Phi\tilde{a})_j}}\left[\sum_j (1-\Phi_{ij})e^{\frac{1}{K}(\Phi\tilde{a})_j}\right].$$

Then dividing the numerator and denominator by $\exp(\frac{1}{K}\sum_j \tilde{a}_j)$, gives:

$$1-\tilde{\nu}_i = \frac{1}{\sum_j e^{-\frac{1}{K}([1_K 1_K^T-\Phi]\tilde{a})_j}}\left[\sum_j [1_K 1_K^T - \Phi]_{ij} e^{-\frac{1}{K}([1_K 1_K^T-\Phi]\tilde{a})_j}\right] = \frac{b_i}{D},$$

Once we recall that $\tilde{a}_i = \frac{1}{\sqrt{n}}a_i$, this gives

$$\frac{d\alpha_i}{dt} = \frac{\sqrt{n}}{D}b_i\beta_i, \quad \frac{d\beta_i}{dt} = \frac{\sqrt{n}}{D}b_i\alpha_i,$$

where, for $a_i = \alpha_i\beta_i$:

$$b_i = \sum\nolimits_{j\in[K]} [1_K 1_K^T - \Phi]_{ij}\, e^{-\frac{1}{K\sqrt{n}}([1_K 1_K^T-\Phi]a)_j}, \quad D = \sum\nolimits_{j\in[K]} e^{-\frac{1}{K\sqrt{n}}([1_K 1_K^T-\Phi]a)_j}.$$

The first row and column of Φ consist of all ones. Since $[1_K 1_K^T - \Phi]_{1j} = 0$ for all j, it follows that $b_1 = 0$, and therefore α_1, β_1 remain constant. Moreover, since $[1_K 1_K^T - \Phi]_{j1} = 0$ for all j, these trivial singular values do not appear in the evolution equations for the others. We may thus, without loss of generality, set $\alpha_1 = \beta_1 = 0$ and henceforth take $\alpha, \beta \in \mathbb{R}^{K-1}$ to be the vectors of non-trivial singular values, with indices shifted accordingly. Consequently, define $\Psi \in \mathbb{R}^{K-1\times K-1}$ by deleting the first column and row of $1_K 1_K^T - \Phi$, so that $\Psi = 1_{K-1}1_{K-1}^T - X$, where $X = \Phi[2:K, 2:K]$ is the core of the Sylvester Hadamard matrix.

It is simple to verify these equations have the conserved quantities $\alpha_i^2 - \beta_i^2 = C_i$. For simplicity, and following common practice [4, 8, 21, 22], we adopt the balancedness condition that sets $C_i = 0$, thus $\alpha_i = \beta_i$ throughout. Under the balancedness condition $\alpha_i = \beta_i$, we have that the logit singular values evolve as

$$\frac{da_i}{dt} = \frac{d\alpha_i}{dt}\beta_i + \alpha_i\frac{d\beta_i}{dt} = \frac{\sqrt{n}}{D}b_i(\alpha_i^2+\beta_i^2) = \frac{2\sqrt{n}}{D}b_i a_i.$$

Then redefining

$$a' = \frac{1}{K\sqrt{n}}a, \quad t' = 2t\sqrt{n},$$

gives the gradient flow equations given in Eq. (3), once primes are dropped, since this represents a simple rescaling that does not change phenomenology.

C.3. Proof of Theorem 4

The KL divergence $D_{\mathrm{KL}}(p \,\|\, q) = \sum_i p_i \log\left(\frac{p_i}{q_i}\right)$ between two categorical distributions with probabilities p and q is non-negative and equals zero if and only if $p = q$. In our setting, we have:

$$D_{\mathrm{KL}}\Big(\frac{1}{K-1}1_{K-1} \,\|\, \hat{a}\Big) = -\log(K-1) - \frac{1}{K-1}\sum_i \log(\hat{a}_i).$$

For the remainder of the proof we denote this by D_{KL}. Differentiating and using Eq. (5), we obtain

$$\frac{d}{dt}\left(D_{\mathrm{KL}}\right) = -\frac{1}{K-1}\sum_i \frac{1}{\hat{a}_i}\frac{d}{dt}(\hat{a}_i) = -\frac{1}{(K-1)D}\left[\left(\sum_i b_i\right) - (K-1)\bar{b}\right], \tag{9}$$

We now define $x_i = (\Psi a)_i$ and rewrite the derivative in terms of these variables, which recall play the role of margins. First observe that

$$\bar{b} = \sum_i \hat{a}_i b_i = \sum_{ij} \hat{a}_i \Psi_{ij} e^{-(\Psi a)_j} = \frac{1}{\|a\|_1}\sum_{ij} a_i \Psi_{ij} e^{-(\Psi a)_j} = \frac{1}{\|a\|_1}\sum_j x_j e^{-x_j}. \tag{10}$$

Second note

$$\sum_i b_i = \sum_{ij} \Psi_{ij} e^{-(\Psi a)_j} = K\sum_j e^{-(\Psi a)_j} = \left(\sum_i (\Psi\hat{a})_i\right)\sum_j e^{-(\Psi a)_j},$$

where we used $\sum_i \Psi_{ij} = K = \sum_i (\Psi\hat{a})_i$. Hence, we obtain

$$\sum_i b_i = \frac{1}{\|a\|_1}\Big(\sum_i x_i\Big)\Big(\sum_j e^{-x_j}\Big). \tag{11}$$

Substituting Eqns. (10) and (11) into Eq. (9) yields

$$\frac{d}{dt}(D_{\mathrm{KL}}) = \frac{1}{(K-1)D\|a\|_1}\left[(K-1)\sum_j x_j e^{-x_j} - \left(\sum_i x_i\right)\left(\sum_j e^{-x_j}\right)\right].$$

Now, the bracketed term can be rewritten as $\frac{1}{2}\sum_{ij}(x_i - x_j)\left(e^{-x_i} - e^{-x_j}\right)$, yielding

$$\frac{d}{dt}(D_{\mathrm{KL}}) = \frac{1}{2(K-1)D\|a\|_1}\sum_{ij}(x_i - x_j)\left(e^{-x_i} - e^{-x_j}\right).$$

For each pair (i, j), the terms $x_i - x_j$ and $e^{-x_i} - e^{-x_j}$ have opposite signs unless $x_i = x_j$. Hence, every summand is non-positive, vanishing only when $x_i = x_j$. Hence the full sum is always non-positive, and equal to zero only if x is a uniform vector. Because $D > 0$ and $\|a\|_1 > 0$, the prefactor is strictly positive. Moreover, since Ψ is invertible and has 1_{K-1} as an eigenvector, the condition $x \propto 1_{K-1}$ implies $a \propto 1_{K-1}$, and hence $\hat{a} = \frac{1}{K-1}1_{K-1}$.

Thus $D_{\mathrm{KL}} \geq 0$ with equality only at $\hat{a} = \frac{1}{K-1}1_{K-1}$, and $\frac{d}{dt}D_{\mathrm{KL}} \leq 0$ with equality only at the same point. Consequently, D_{KL} decreases monotonically to zero, guaranteeing $\hat{a} \to \frac{1}{K-1}1_{K-1}$.

C.4. Proof of Theorem 5

Recall our choice of metric is given by:

$$M = \frac{1}{2}\sum_{i=1}^{K-1}\left(\hat{a}_i - \frac{1}{K-1}\right)^2.$$

A simple calculation, using the derivatives provided in Eq. (5), gives

$$\frac{dM}{dt} = \frac{1}{D}\sum_i \hat{a}_i^2[b_i - \bar{b}]\,. \tag{12}$$

First note that $M = 0$ in the $K = 2$ case for all positive initializations, and so trivially is monotonic. We begin by proving monotonicity for $K = 4$.

Note in $K = 4$ we have

$$\Psi = \begin{bmatrix} 2 & 0 & 2 \\ 0 & 2 & 2 \\ 2 & 2 & 0 \end{bmatrix} \implies \Psi a = \begin{bmatrix} 2a_1 + 2a_3 \\ 2a_2 + 2a_3 \\ 2a_1 + 2a_2 \end{bmatrix} \implies \exp(-\Psi a) = \begin{bmatrix} e^{-2a_1-2a_3} \\ e^{-2a_2-2a_3} \\ e^{-2a_1-2a_2} \end{bmatrix}.$$

Hence

$$\frac{1}{D}b = \frac{2}{1 + e^{-2a_1-2a_3} + e^{-2a_2-2a_3} + e^{-2a_1-2a_2}} \begin{bmatrix} e^{-2a_1-2a_3} + e^{-2a_1-2a_2} \\ e^{-2a_2-2a_3} + e^{-2a_1-2a_2} \\ e^{-2a_1-2a_3} + e^{-2a_2-2a_3} \end{bmatrix}.$$

Multiplying numerator and denominator by $\exp(2\sum_i a_i)$ gives

$$\frac{1}{D}b = \frac{2}{e^{2(a_1+a_2+a_3)} + e^{2a_1} + e^{2a_2} + e^{2a_3}} \begin{bmatrix} e^{2a_2} + e^{2a_3} \\ e^{2a_1} + e^{2a_3} \\ e^{2a_1} + e^{2a_2} \end{bmatrix}.$$

Writing $x_i = \exp(2a_i)$, this is then

$$\frac{1}{D}b = \frac{2}{x_1x_2x_3 + \sum_i x_i} \begin{bmatrix} \sum_i x_i - x_1 \\ \sum_i x_i - x_2 \\ \sum_i x_i - x_3 \end{bmatrix},$$

and then $\bar{b}$ is given by:

$$\frac{1}{D}\bar{b} = \frac{2}{x_1x_2x_3 + \sum_i x_i}[\sum_i x_i - \hat{a}\cdot x].$$

Hence using the expression for the derivative of our metric, given in Eq. (12), we arrive at:

$$\frac{dM}{dt} = \frac{2}{x_1x_2x_3 + \sum_i x_i}\sum_i \hat{a}_i^2[\hat{a}\cdot x - x_i].$$

The coefficient is strictly positive, and so to argue non-increasing we can drop it. Hence we simply need to show:

$$(\hat{a}\cdot x)\left(\sum_i \hat{a}_i^2\right) - \sum_i \hat{a}_i^2 x_i \le 0,$$

or equivalently:

$$\sum_i \hat{a}_i^2 \leq \frac{\sum_i \hat{a}_i^2 x_i}{\sum_i \hat{a}_i x_i}.$$

Define the quantities:

$$q_i = \frac{\hat{a}_i x_i}{\sum_i \hat{a}_i x_i},$$

noting that they form a probability distribution. Then our inequality can be viewed as comparing the expectation of $\hat{a}$ over two different probability distributions, one with probabilities $\hat{a}$ and the other with probabilities q.

It is then simple to note that, since for any $\|a\|_1 > 0$, x_i is an increasing function of $\hat{a}_i$, the second distribution places more weight on larger values of $\hat{a}$, and so must be at least as large as the other expectation. They are only equal when there are no larger values of $\hat{a}_i$, meaning $\hat{a} = \frac{1}{K-1} 1_{K-1}$. Hence the inequality is satisfied, and the metric is monotonically decreasing.

It remains to show that the metric is not necessarily monotonic decreasing for $K \geq 8$. To show this for $K = 8$, it is sufficient to note that for $a = [1, 1, 1, 1.25, 0.01, 0.01, 0.01]$ the derivative of our metric takes value $\frac{dM}{dt} \approx 7.3 \times 10^{-4}$, and so the metric is increasing for these values. Call this specific counter-example v^*, and denote its corresponding b vector and its weighted average, by b^* and $\bar{b}^*$.

For $K > 8$, define our initialization as $a = [v^*, 0_{K-8}]$, then the derivative of our metric becomes

$$\frac{dM}{dt} = \frac{1}{D} \sum_{i=1}^{7} (\hat{v}_i^*)^2 [b_i - \bar{b}], \qquad \text{where } \bar{b} = \sum_{j=1}^{7} v_i^* b_i.$$

Note that only $b_1, ..., b_7$ are necessary for the computation of the sign. We now use the lemma stated and proven at the end of this section, which gives that when a is only non-zero on at most the first $\frac{K}{2} - 1$ entries, then the first $\frac{K}{2} - 1$ entries of the b vector are just equal to double the values they would be in the $\frac{K}{2}$ dimensional case. Applying this iteratively, we get that if $K = 2^m > 8$, then

$$b_i = 2^{m-3} b_i^*, \qquad \bar{b} = 2^{m-3} \bar{b}^*, \qquad \text{for } i = 1, ..., 7,$$

and so

$$\frac{dM}{dt} = \frac{2^{m-3}}{D} \sum_{i=1}^{7} (\hat{v}_i^*)^2 [b_i^* - \bar{b}^*], \qquad \text{where } \bar{b} = \sum_{j=1}^{7} v_i^* b_i,$$

but this clearly has the same sign as the counter-example in $K = 8$, and so gives a positive derivative.

This provides a counter-example in any K, but it has entries that are zero, which is not permitted. However if we consider the vector

$$a = [v^*, \epsilon 1_{K-8}],$$

then by continuity, as $\epsilon \to 0$ this tends to the positive value achieved by $a = [v^*, 0_{K-8}]$. Hence for any K there exists a non-zero value of ϵ such that $\frac{dM}{dt} > 0$ for $a = [v^*, \epsilon 1_{K-8}]$. Hence counter-examples exist in all $K \geq 8$.

It only remains to prove the following Lemma.

Lemma 6 *Let $K = 2^m$, and $a^* \in \mathbb{R}^{K-1}$, with $b(a^*) = b^* \in \mathbb{R}^{K-1}$ and $\bar{b}^* = \sum_i \hat{a}_i^* b_i^* \in \mathbb{R}^{K-1}$. Let $a' \in \mathbb{R}^{2K-1}$ be given by $a' = [a^*, 0_K]$, and consider the evolution under the dynamics detailed in Eq. (3), but for dimension $2K$, initialized at a'. Denote the corresponding vectors $b(a'), \bar{b}(a') \in \mathbb{R}^{2K-1}$ by b' and $\bar{b}'$ respectively. Then for $i = 1, ..., K-1$, we have*

$$b'_i = 2b_i^*, \qquad \bar{b}' = 2\bar{b}^*.$$

Proof of Lemma 6:

Let Ψ_{K-1} and Ψ_{2K-1} denote the two matrices derived from the Hadamard matrix, as defined in Sec. 3, with the subscript denoting their dimension. Using the construction in terms of Sylvester Hadamard matrices, as defined in Definition 1, these are related in the following way:

$$\Psi_{2K-1} = \begin{bmatrix} \Psi_{K-1} & 0_{K-1} & \Psi_{K-1} \\ 0_{K-1}^T & 2 & 21_{K-1}^T \\ \Psi_{K-1} & 21_{K-1} & 21_{K-1}1_{K-1}^T - \Psi_{K-1} \end{bmatrix}.$$

Hence

$$\Psi_{2K-1}a' = \begin{bmatrix} \Psi_{K-1} & 0_{K-1} & \Psi_{K-1} \\ 0_{K-1}^T & 2 & 21_{K-1}^T \\ \Psi_{K-1} & 21_{K-1} & 21_{K-1}1_{K-1}^T - \Psi_{K-1} \end{bmatrix} \begin{bmatrix} a^* \\ 0 \\ 0_{K-1} \end{bmatrix} = \begin{bmatrix} \Psi_{K-1}a^* \\ 0 \\ \Psi_{K-1}a^* \end{bmatrix},$$

and so

$$\exp(-\Psi_{2K-1}a) = \begin{bmatrix} \exp(-\Psi_{K-1}a^*) \\ 1 \\ \exp(-\Psi_{K-1}a^*) \end{bmatrix},$$

giving

$$b' = \Psi_{2K-1}\exp(-\Psi_{2K-1}a) = \begin{bmatrix} \Psi_{K-1} & 0_{K-1} & \Psi_{K-1} \\ 0_{K-1}^T & 2 & 21_{K-1}^T \\ \Psi_{K-1} & 21_{K-1} & 21_{K-1}1_{K-1}^T - \Psi_{K-1} \end{bmatrix} \begin{bmatrix} \exp(-\Psi_{K-1}a^*) \\ 1 \\ \exp(-\Psi_{K-1}a^*) \end{bmatrix}.$$

From this it is clear that the top $K-1$ entries of b' are given by the vector $2\Psi_{K-1}\exp(-\Psi_{K-1}a) = 2b^*$. Also note

$$\bar{b}' = \sum_j \hat{a}'_j b'_j = \sum_{j=1}^{K-1} \hat{a}_j^* b'_j = 2\bar{b}^*.$$

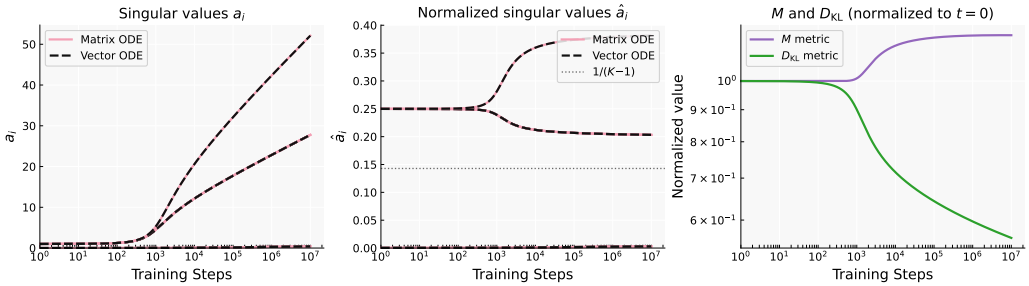


Figure 3: Training with $K = 8$ and $a(0) = [1, 1, 1, 1, 0.001, 0.001, 0.001]$. **Left/Middle:** Evolution of raw/normalized singular values. **Right:** Evolution of the metric M in Eq. (6) and the KL Lyapunov function of Thm. 4, both normalized by their initial values.

Appendix D. Further numerical experiments

D.1. Further details on figures in main body

D.1.1. FIG. 1

For Fig. 1, we use gradient descent with learning rate 0.01 and $1e7$ training steps, recorded at 1000 logarithmically spaced checkpoints. The initial logit singular values are sampled independently from the uniform distribution on $[0, 2]$. We then compare the evolution obtained from the full matrix dynamics with under Hadamard initialization with the reduced vector dynamics in Eq. (3). The two trajectories are initialized with the same singular values. The reduced vector ODE is solved using an adaptive Runge–Kutta method with relative tolerance 10^{-8} and absolute tolerance 10^{-10}; to compare with gradient descent, we evaluate it at continuous times $t_k = \eta k$, where k is the training step and $\eta = 0.01$ is the learning rate.

D.1.2. FIG. 2

For Fig. 2, we use $K = 8$ classes, one sample per class, and 10 random realizations. We train for $1e7$ gradient-descent steps with learning rate 0.01, using 5000 logarithmically spaced checkpoints. The random initialization is $W(0) = e^{-\delta}\frac{A_1}{\sqrt{K}}$, and $H(0) = e^{-\delta}\frac{A_0}{\sqrt{K}}$, where the entries of A_1 and A_0 are sampled independently from a standard Gaussian distribution and we here use $\delta = 12$. We also set $d = 5K$. The Hadamard initialization uses the same initial logit singular values as the corresponding random initialization, but places the singular vectors exactly in the Hadamard eigenspaces. Concretely, we initialize $W = UD_WR^T$ and $H = RD_HV^T$ as described in Theorem 2, $U = V = \frac{1}{\sqrt{K}}\Phi$. Recall from Sec. 3 that the singular value corresponding to the all 1s column of U is not updated by the flow; thus, as in theory, we assume without loss of generality that the respective entry of D_W, D_H is set to zero. To further ensure direct comparison to random initialization, we set the remaining $K - 1$ singular values such that $D_W = D_H$ and they are identical to the largest $K - 1$ singular values of the logit matrix from its corresponding random initialization pair. This isolates the *initial* difference between the two schemes to only the singular vectors.

The residual energy plotted in Fig. 2(f) measures $\|Z - U(U^TZV)V^T\|_F/\|Z\|_F$, for $U = V = \Phi(1 : K, 2 : K)$. We use residual threshold 0.005 to identify the end of the alignment phase. Specifically, we first wait until the residual drops below 0.005, and then define the "burn-in time" as the first logged time at which the residual rises above 0.005 again. This choice selects the point at which the random trajectory has already passed through its closest approach to the

Hadamard subspace. The late-Hadamard trajectory is then initialized at this burn-in time by matching the singular values of the random trajectory and placing the singular vectors in the Hadamard eigenspaces. The vertical dotted line in Fig. 2 indicates the average burn-in time across the 10 realizations.

D.2. Metrics used in the experiments

We track the following metrics as function of iterations:

(a) *Cross-Entropy Loss:* Plots the CE loss in Eq. (1).

(b) *Alignment Metrics:* Tracks the Frobenius norm distance between the nuclear-norm-normalized model components ($WW^T, H^TH, Z = WH$) and the ground-truth Simplex ETF S. For example for the logit matrix $Z = WH$ it plots $\left\| \frac{Z}{\|Z\|_*} - \frac{S}{\|S\|_*} \right\|_F$. For Hadamard Init., where Z and S share singular spaces, this is identical to the (square root of the) metric M in Eq. (6).

(c) *KL Divergence:* Plots the Lyapunov function from Theorem 4, measuring the KL divergence of the normalized logit singular values $\hat{a}$ from the uniform distribution $\frac{1}{K-1}1_{K-1}$.

(d) *Singular Values:* Plots the evolution of the raw singular values $\sigma_i(Z)$ of the logit matrix $Z = WH$. For the *Hadamard Init.* and *ODE* runs, these are equivalent to the main body's logit singular values a_i.

(e) *Normalized Singular Values:* Plots the L_1-normalized singular values $\hat{\sigma}_i(Z) = \sigma_i(Z)/\sum_j \sigma_j(Z)$. For the *Hadamard Init.* and *ODE* runs, these are equivalent to the main body's $\hat{a}_i = a_i/\|a\|_1$.

(f) *Residual Energy:* Measures the fraction of the logit matrix Z that lies outside the signal subspace of the Hadamard Init.. Specifically, for $U = V = \Phi(1:K, 2:K)$, it measures $\|Z - U(U^TZV)V^T\|_F/\|Z\|_F$

We track and plot these metrics in 5000 logarithmically spaced time instants. All results are averaged over 10 runs, with the shaded regions in our plots representing the standard error of the mean.

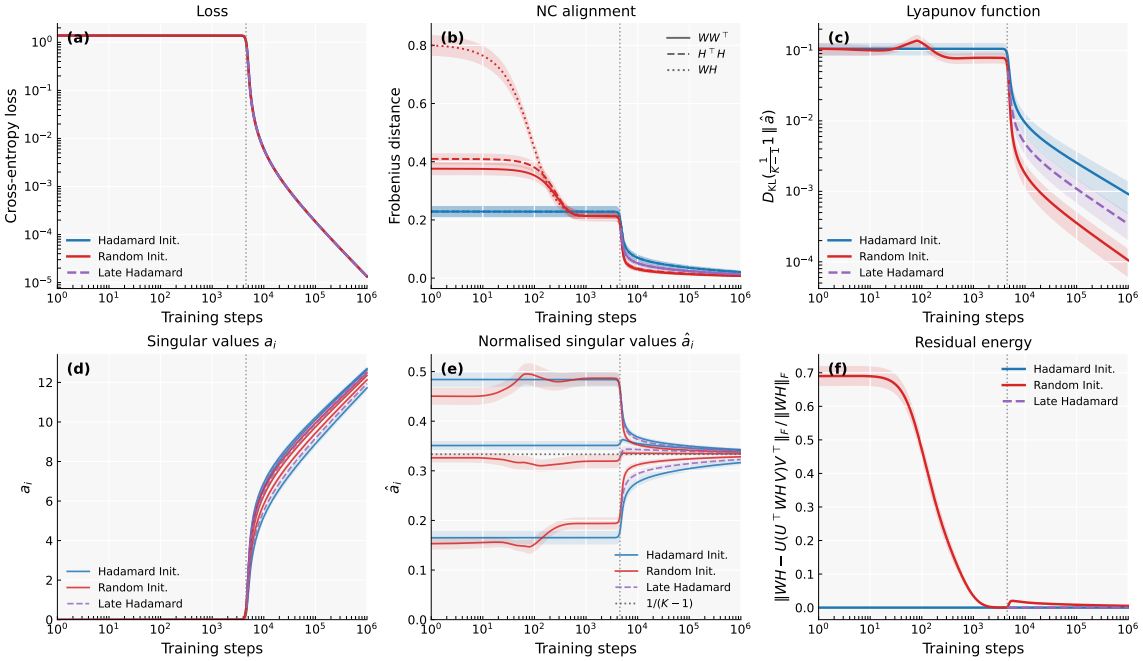
D.3. Non-monotonic behavior example

In Theorem 5 we showed that the metric M, defined in Eq. (6), can display behavior showing the dynamics moving away from NC. Fig. 3 illustrates the severity of this phenomenon, showing the evolution of $\hat{a}_i$ and M for an initialization leading to non-monotonic behavior: M increases monotonically over the displayed trajectory, even if it must eventually go to zero.

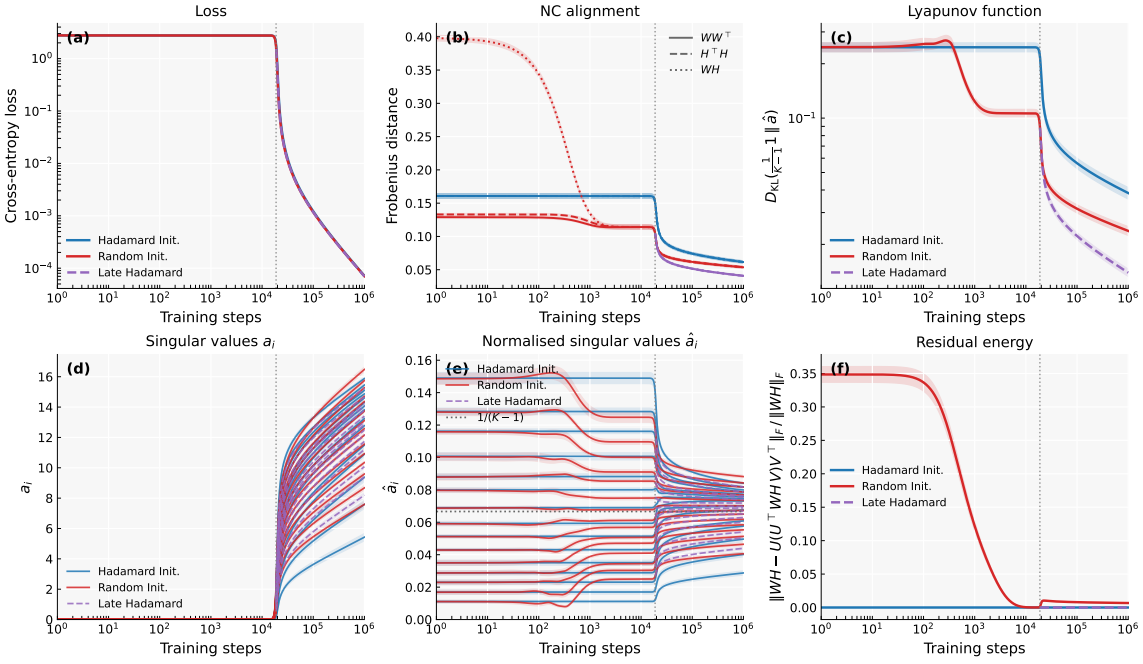
D.4. Additional random-versus-Hadamard experiments for $K = 4$ and $K = 16$

We provide additional experiments comparing small random initialization and Hadamard initialization for $K = 4$ and $K = 16$, with $d = 5K$, in Fig. 4. These experiments complement the $K = 8$ results in Fig. 2 (setup is the same as described in Sec. D.1). In all cases, both random and Hadamard initialization exhibit a similar qualitative two-stage behavior. In the first, small-logit phase, the loss changes slowly and the singular values remain small. For random initialization, this phase

also corresponds to alignment of the logit matrix toward the Hadamard subspace; for Hadamard initialization, the trajectory is already in this subspace, and the normalized singular values remain constant. In the second, large-logit phase, the singular values grow, the loss decreases, and the normalized singular values begin moving toward their limiting equilibrium. This behavior is very similar to that observed in Fig. 2 of the main text.



(a) $K = 4, d = 5K$



(b) $K = 16, d = 5K$

Figure 4: Additional comparison between gradient flow under small random initialization and Hadamard initialization for $K = 4$ and $K = 16$. Plots are the same as Fig. 2 other than updated class number. The two-stage behavior of random initialization is visible in both cases.