NEURAL LOFI WITH BACKWARD COUPLING: A SPECTRAL THEORY OF CROSS-LAYER FEATURE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

During joint training, backpropagation allows later layers to reshape earlier representations, a feedback mechanism absent from purely layer-wise learning. We extend Neural Low-Degree Filtering (Neural LoFi), a recently proposed layer-wise spectral surrogate for feature learning, to capture this interaction. Starting from the exact backpropagation gradients, we derive a local coupling between a neuron's incoming weights and its corresponding next-layer column. In a controlled small-initialization regime with smooth activations and fixed surroundings, these dynamics amplify singular modes of a cross-layer correlation operator. Each growing mode pairs an earlier feature with an upper-layer response through shared neuron coefficients. This motivates a backward-coupling algorithm in which upper-layer sensitivities generate sample-dependent effective labels for earlier spectral steps, while corrected features refine upward transmission. A function-space variational principle characterizes the selected feature–response pairs by maximizing the label correlation of their product under kernel-norm constraints. For three-layer networks, we derive self-consistent continuous-time dynamics for the filtered hidden-layer features, the activation kernels, and the network output under the proposed backward coupling. In a hierarchical multi-index model, we establish a feature-recovery separation: backward coupling recovers lower-layer directions invisible to the original forward label-weighted statistic. The linear specialization recovers residual-dependent singular-mode dynamics, while paired modes provide a dynamical mechanism for linked low-rank covariance components consistent with the rainbow-network picture. Experiments with fully connected and convolutional networks on binary CIFAR-10 show improved accuracy over forward Neural LoFi at matched widths.

1 INTRODUCTION

Deep neural networks learn hierarchies of task-adapted representations through backpropagation (Rumelhart et al., 1986; LeCun et al., 2015). Yet a hierarchy in the forward computation does not imply a layer-wise order of learning. During joint training, features emerging in later layers reshape the learning signals received below, while changes in earlier representations alter what the upper layers can learn. Understanding this feedback requires going beyond how features are constructed successively. It raises a central question:

How does learning in later layers change which features earlier layers can discover?

For purely layer-wise learning, Neural Low-Degree Filtering (Neural LoFi) provides an explicit spectral construction (Dandi et al., 2026b). Given the current representation, each layer selects directions through low-degree correlations with the labels and passes them through a nonlinear random lift. Depth changes the geometry in which subsequent layers search for signal, making structure that is complex in the input accessible through low-degree statistics of intermediate representations. However, this construction proceeds only forward: features learned above cannot refine the directions selected below. A useful upper representation may therefore become available only after the lower feature-selection step that could benefit from it has already been completed.

Existing theories describe complementary aspects of feature learning. Neural-tangent limits characterize training with essentially fixed representations (Jacot et al., 2018; Lee et al., 2019); mean-field

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

and dynamical-field frameworks describe evolving features in shallow and deep nonlinear networks (Mei et al., 2018; Chizat & Bach, 2018; Nguyen & Pham, 2020; Bordelon & Pehlevan, 2022; Hajjar et al., 2024). Feature-space mean-field models and tensor programs further describe feature learning at infinite width (Fang et al., 2021; Yang & Hu, 2021). Deep linear networks admit explicit analyses of coupled singular-mode dynamics (Saxe et al., 2014), while spectral analyses of quadratic and diagonal networks connect learned weight structure to generalization (Defilippis et al., 2026b). Backward feature correction studies how joint training with higher layers can refine lower representations (Allen-Zhu & Li, 2023).

In this paper, we seek a tractable and interpretable spectral mechanism that identifies how an upper-layer response changes the "earlier" features. We extend Neural LoFi with a tractable and interpretable spectral surrogate that captures backward feature correction and, in addition, identifies a distinct effect of joint training: learning how lower-level cues selectively activate higher-level features. This couples the learning of lower representations to the learning of their transmission upward. Starting from the exact backpropagation gradients, we couple a neuron's incoming row to the next-layer column that reads its activation. For a jointly trained block at small initialization, with smooth activations and fixed surroundings, we prove that the leading dynamics amplify singular modes of a cross-layer correlation operator. Each mode pairs an earlier feature with an upper response, with the same growing coefficients across intermediate neurons in both weight factors. The paired updates coordinate what a neuron detects with how its activation is transmitted upward: the incoming row acquires the lower feature, while the outgoing column couples it to the selected upper response. With linear activations, the residual-driven row–column equations are exact and recover the singular-mode framework of Saxe et al. (2014).

This mechanism motivates BackLoFi, a spectral correction of a forward LoFi network that visits layer boundaries from the top down. Upper-layer sensitivities define sample-dependent effective labels for earlier spectral steps; the corrected nonlinear activations then determine the corresponding upward transmission. The algorithm uses spectral selection and normalized corrections as a surrogate for the local dynamics. Its selection rule has a function-space interpretation: an earlier feature χ and an upper response ψ maximize $\widehat{\mathbb{E}}_n[y\psi\chi]$, subject to the kernel-norm constraints of their respective spaces. The search for a lower feature thus depends on the response learned above as well as on the input representation and the labels. That backward feedback can make a previously invisible lower feature detectable, providing a drastic advantage to cross-layer training over layer-wise one.

We illustrate this in a hierarchical multi-index model (Nichani et al., 2023; Wang et al., 2023; Fu et al., 2025; Dandi et al., 2026a; Tabanelli et al., 2026) with a visible anchor and additional hidden directions. In its Gaussian analysis, the original label-weighted second moment has no signal in the hidden block, while modulation by a sensitivity of the upper conditional predictor creates one. This illustrates a staircase learning mechanism through the backward pass.

A second theoretical contribution is a closed description of coupled nonlinear feature learning in a three-layer network. For a continuous variant with fixed Gaussian lifts, continual forward filtering, and readout gradient flow, we prove convergence to self-consistent feature and kernel dynamics. The evolving kernels determine subsequent feature selection, while an accumulated coupling retains earlier column corrections as lower features change. This provide an unifying description of how multi-layer kernel adaptivity Bach (2021) arise through cross-layer feature learning. It generalizes and extend the seminal work of Saxe et al. (2014) that was limited to linear networks.

Further, this mechanism also allows to provide an explanation for the rainbow-network picture of (Guth et al., 2024): Shared neuron coefficients yield linked low-rank covariance components in adjacent weight matrices, providing a local dynamical mechanism predicting the rainbow structure.

Finally, mechanistic experiments with fully connected and convolutional networks on binary CIFAR-10 show considerable improved accuracy over forward Neural LoFi at matched widths. They demonstrates that using upper-layer information to revise earlier representations is useful beyond the controlled theoretical setting, and that the predicted mechanism translate into real data scenarios.

2 Neural Low-Degree Filtering with backward coupling

Backpropagation, and the guiding approximation — We first discuss the exact updates of a neuron's incoming row and outgoing column, then use their local coupling to motivate our spectral

algorithm. Consider data $\mathcal{D}_n = \{(\mathbf{x}_\mu, y_\mu)\}_{\mu=1}^n$, with centered scalar labels, and write $\widehat{\mathbb{E}}_n$ for the empirical average. Let $\mathbf{z}_0 : \mathcal{X} \to \mathbb{R}^{p_0}$ be a given input feature map; raw inputs are the special case $\mathbf{z}_0(\mathbf{x}) = \mathbf{x}$. A depth-$L$ network constructs

$$\mathbf{h}_\ell = \mathbf{W}_\ell \mathbf{z}_{\ell-1}, \qquad \mathbf{z}_\ell = \frac{1}{\sqrt{p_\ell}}\sigma(\mathbf{h}_\ell), \qquad \widehat{f} = \langle \mathbf{a}_L, \mathbf{z}_L \rangle + b. \tag{1}$$

Here $\mathbf{W}_\ell \in \mathbb{R}^{p_\ell \times p_{\ell-1}}$ and the activation acts coordinatewise. The sensitivity of the output to the preactivations, $\mathbf{g}^\ell = \partial \widehat{f}/\partial \mathbf{h}_\ell$, obeys the backward recursion

$$\mathbf{g}^L = \frac{\sigma'(\mathbf{h}_L)}{\sqrt{p_L}} \odot \mathbf{a}_L, \qquad \mathbf{g}^\ell = \frac{\sigma'(\mathbf{h}_\ell)}{\sqrt{p_\ell}} \odot (\mathbf{W}_{\ell+1}^\top \mathbf{g}^{\ell+1}). \tag{2}$$

For the square loss, write $\Delta = y - \widehat{f}$ for the residual; gradient flow then satisfies

$$\dot{\mathbf{W}}_\ell = \widehat{\mathbb{E}}_n[\Delta\, \mathbf{g}^\ell \mathbf{z}_{\ell-1}^\top]. \tag{3}$$

Neuron j of layer ℓ owns two weight vectors: its incoming row, whose transpose we write $\mathbf{w}_{\ell,j}$ (row j of $\mathbf{W}_\ell$), and its outgoing column $\boldsymbol{\omega}_{\ell+1,j}$ (column j of $\mathbf{W}_{\ell+1}$), through which the next layer reads its activation (Figure 1). By (2) and (3), their updates by gradient descent are

$$\dot{\mathbf{w}}_{\ell,j} = \widehat{\mathbb{E}}_n\left[\Delta\, \frac{\sigma'(h_{\ell,j})}{\sqrt{p_\ell}} \langle \boldsymbol{\omega}_{\ell+1,j}, \mathbf{g}^{\ell+1} \rangle \mathbf{z}_{\ell-1}\right], \quad \dot{\boldsymbol{\omega}}_{\ell+1,j} = \widehat{\mathbb{E}}_n[\Delta\, z_{\ell,j} \mathbf{g}^{\ell+1}]. \tag{4}$$

Each update depends on the other vector: the outgoing column sets the sample-dependent gain driving the row, while the row's activation determines the column update.

Layer-wise vs coupled training — In (Dandi et al., 2026b), the authors considered the layer-wise "Neural LoFi regime" of the first equation in which the layers above are still untrained and vanishingly small. The factor $\langle \boldsymbol{\omega}_{\ell+1,j}, \mathbf{g}^{\ell+1}(\mathbf{x}) \rangle$, the effective readout of neuron j, then reduces to a sample-independent random constant $\bar{a}_j$, and the residual to the label. For an activation smooth near zero, $\sigma'(h_{\ell,j}) \simeq \sigma'(0) + \sigma''(0) h_{\ell,j}$ gives

$$\dot{\mathbf{w}}_{\ell,j} \simeq \frac{\bar{a}_j}{\sqrt{p_\ell}} \Big(\sigma'(0)\, \widehat{\mathbb{E}}_n[y\, \mathbf{z}_{\ell-1}] + \sigma''(0)\, \widehat{\mathbb{E}}_n[y\, \mathbf{z}_{\ell-1} \mathbf{z}_{\ell-1}^\top]\, \mathbf{w}_{\ell,j} \Big). \tag{5}$$

The rows evolve independently under the same label-weighted moments. Neural LoFi retains the first-moment direction and the leading second-moment eigenvectors by absolute eigenvalue, then applies a random lift. This leads to Alg.1 and weights of the form (9).

Consider now a *joint training* procedure. With a learned upper representation, the effective readout becomes sample dependent. To isolate the coupling, freeze the surrounding maps $\mathbf{g}^{\ell+1}$ and $\mathbf{z}_{\ell-1}$ and start the row–column pair small. For a smooth activation with $\sigma(0) = 0$, retain the constant derivative $\sigma'(0)$ in the row update and the linear activation in the column update. Assuming $\sigma'(0) > 0$ gives

$$\dot{\mathbf{w}}_{\ell,j} \simeq \frac{\sigma'(0)}{\sqrt{p_\ell}} \widetilde{\mathbf{A}}_\ell^\top \boldsymbol{\omega}_{\ell+1,j}, \qquad \dot{\boldsymbol{\omega}}_{\ell+1,j} \simeq \frac{\sigma'(0)}{\sqrt{p_\ell}} \widetilde{\mathbf{A}}_\ell \mathbf{w}_{\ell,j}, \qquad \widetilde{\mathbf{A}}_\ell := \widehat{\mathbb{E}}_n[\Delta\, \mathbf{g}^{\ell+1} \mathbf{z}_{\ell-1}^\top]. \tag{6}$$

At a fixed residual, the same matrix acts in both directions and its singular pairs determine the growing modes. Close to initialization, under the initialization assumptions of Appendix B, the prediction is small, so $\Delta \simeq y$ and the frozen coupling matrix becomes

$$\mathbf{A}_\ell := \widehat{\mathbb{E}}_n[y\, \mathbf{g}^{\ell+1} \mathbf{z}_{\ell-1}^\top]. \tag{7}$$

Around a fitted baseline f_0, the corresponding local gradient approximation instead uses $y - f_0$. Write s_i and $(\mathbf{u}_i, \mathbf{v}_i)$ for the singular values and pairs of $\mathbf{A}_\ell$.

Theorem 1 (Informal: amplification across adjacent layers)**.** *Under vanishing initialization and fixed surrounding layers, for small enough time, a singular mode $(\mathbf{u}_i, \mathbf{v}_i, s_i)$ of the frozen $\mathbf{A}_\ell$, with non-zero initial component, has growing components:*

$$\langle \mathbf{w}_{\ell,j}(t), \mathbf{v}_i \rangle \simeq c_{ji} e^{\sigma'(0) s_i t/\sqrt{p_\ell}}, \qquad \langle \boldsymbol{\omega}_{\ell+1,j}(t), \mathbf{u}_i \rangle \simeq c_{ji} e^{\sigma'(0) s_i t/\sqrt{p_\ell}}, \tag{8}$$

with a random shared amplitude c_{ji} set by initialization. See App. B for details and error bounds.

Collecting the growing amplitudes of the linear approximation (6) over the neurons of layer ℓ, $\mathbf{q}_i = (c_{ji})_j$, the growing parts have the rank-one directions $\mathbf{q}_i \mathbf{v}_i^\top$ in $\mathbf{W}_\ell$ and $\mathbf{u}_i \mathbf{q}_i^\top$ in $\mathbf{W}_{\ell+1}$, which we call a *column spike*. The same coefficients $\mathbf{q}_i$ specify which intermediate neurons learn the feature and which columns transmit it.

Algorithm 1 Forward Neural Low-Degree Filtering (Dandi et al., 2026b)

Require: Data $\{(\mathbf{x}_\mu, y_\mu)\}_{\mu=1}^n$, depth L, ranks k_ℓ, widths p_ℓ; $\mathbf{z}_0 = \mathbf{x}$, $p_0 = d$.

1: **for** $\ell = 1, \dots, L$ **do**
2: $\quad \widehat{\mathbf{m}}_\ell \leftarrow \widehat{\mathbb{E}}_n[y\,\mathbf{z}_{\ell-1}],\ \widehat{\mathbf{C}}_\ell \leftarrow \widehat{\mathbb{E}}_n[y\,\mathbf{z}_{\ell-1}\mathbf{z}_{\ell-1}^\top]$ ▷ label-weighted moments
3: $\quad \mathbf{F}_\ell \leftarrow [\,\widehat{\mathbf{m}}_\ell/\|\widehat{\mathbf{m}}_\ell\| \mid$ leading $k_\ell - 1$ eigenvectors of $\widehat{\mathbf{C}}_\ell$ by $|\lambda|\,]$ ▷ filter
4: $\quad \mathbf{z}_\ell \leftarrow \sigma(\mathbf{R}_\ell \mathbf{F}_\ell^\top \mathbf{z}_{\ell-1})/\sqrt{p_\ell},\ \mathbf{R}_\ell \in \mathbb{R}^{p_\ell \times k_\ell}$ Gaussian ▷ random lift
5: **end for**
6: Fit the readout $(\mathbf{a}_L, b)$ on $\mathbf{z}_L$; **return** $\widehat{f}(\mathbf{x}) = \langle \mathbf{a}_L, \mathbf{z}_L(\mathbf{x}) \rangle + b$

Algorithm 2 Neural Low-Degree Filtering with Backward Coupling (BackLoFi)

Require: Forward Neural LoFi network (Algorithm 1) $\mathbf{W}_\ell = \mathbf{R}_\ell \mathbf{F}_\ell^\top$, $\ell = 1, \dots, L$, with readout $(\mathbf{a}_L, b)$; spike ranks r_ℓ, scale $\beta > 0$, passes P.

1: **for** $t = 1, \dots, P$ **do**
2: $\quad$ Compute $\mathbf{h}_\ell, \mathbf{z}_\ell$ for all ℓ by (1); $\mathbf{g}^L \leftarrow \frac{\sigma'(\mathbf{h}_L)}{\sqrt{p_L}} \odot \mathbf{a}_L$ ▷ exact top sensitivity
3: $\quad$ **for** $\ell = L-1, \dots, 1$ **do** ▷ boundaries, top-down
4: $\quad\quad (\mathbf{U}_\ell, \mathbf{V}_\ell) \leftarrow$ leading r_ℓ singular pairs of $\mathbf{A}_\ell \leftarrow \widehat{\mathbb{E}}_n[y\,\mathbf{g}^{\ell+1}\mathbf{z}_{\ell-1}^\top]$
5: $\quad\quad \mathbf{s}_\ell \leftarrow \mathbf{U}_\ell^\top \mathbf{g}^{\ell+1}$ ▷ scores
6: $\quad\quad \mathbf{Q}_\ell^0 \leftarrow \mathbf{W}_\ell \mathbf{V}_\ell$ ▷ spike columns at order zero
7: $\quad\quad \mathbf{F}_\ell \leftarrow [\mathbf{F}_\ell \mid \mathbf{V}_\ell \mathbf{\Gamma}_\ell^{\mathrm{row}}],\ \mathbf{R}_\ell \leftarrow [\mathbf{R}_\ell \mid \mathbf{R}_\ell^{\leftarrow}]$ ▷ rows of layer ℓ
8: $\quad\quad$ Recompute $\mathbf{h}_\ell, \mathbf{z}_\ell$ ▷ features of the corrected rows
9: $\quad\quad \mathbf{Q}_\ell \leftarrow \widehat{\mathbb{E}}_n[\mathbf{z}_\ell (y\mathbf{s}_\ell)^\top]$ ▷ correlations with the modified labels
10: $\quad\quad \mathbf{F}_{\ell+1} \leftarrow [\mathbf{F}_{\ell+1} \mid \mathbf{Q}_\ell \mathbf{\Gamma}_\ell^{\mathrm{col}}],\ \mathbf{R}_{\ell+1} \leftarrow [\mathbf{R}_{\ell+1} \mid \mathbf{U}_\ell]$ ▷ columns of layer $\ell+1$
11: $\quad\quad \mathbf{g}^\ell \leftarrow \frac{\sigma'(\mathbf{h}_\ell)}{\sqrt{p_\ell}} \odot \big(\widehat{\mathbb{E}}_n[\mathbf{W}_{\ell+1}^\top \mathbf{g}^{\ell+1}] + \mathbf{Q}_\ell^0 \mathbf{s}_\ell\big)$ ▷ first-order signal (12)
12: $\quad$ **end for**
13: $\quad$ Recompute $\mathbf{z}_L$; refit $(\mathbf{a}_L, b)$
14: **end for**
15: **return** $\widehat{f}(\mathbf{x}) = \langle \mathbf{a}_L, \mathbf{z}_L(\mathbf{x}) \rangle + b$

Neural LoFi with backward coupling — Our algorithm, BackLoFi (Algorithm 2), is motivated by this result and uses these singular modes to construct a spectral surrogate for cross-layer feature learning. On a fitted network, using labels in place of residuals is a design choice; the theorem justifies this replacement near initialization.

The starting point is to use a *first pass* with forward Neural LoFi (Dandi et al., 2026b) (Algorithm 1), which builds the network one layer at a time from the input up. Given $\mathbf{z}_{\ell-1}$, a filter $\mathbf{F}_\ell \in \mathbb{R}^{p_{\ell-1} \times k_\ell}$ collects the k_ℓ directions amplified in (5): the leading label weighted moments. A Gaussian lift $\mathbf{R}_\ell \in \mathbb{R}^{p_\ell \times k_\ell}$ then distributes them over the p_ℓ neurons,

$$\mathbf{W}_\ell = \mathbf{R}_\ell \mathbf{F}_\ell^\top. \tag{9}$$

After the forward pass, a backward pass visits boundaries from the top down, then refits the readout. Boundary ℓ couples the rows of $\mathbf{W}_\ell$ to the columns of $\mathbf{W}_{\ell+1}$. It receives the sensitivity updated above and reads the representation below, which has not yet changed during the pass.

First, retain the r_ℓ leading singular pairs of $\mathbf{A}_\ell$: $\mathbf{V}_\ell = [\mathbf{v}_i]$ selects lower features and $\mathbf{U}_\ell = [\mathbf{u}_i]$ selects upper responses (line 4). These modes and responses are frozen during the boundary update and recomputed at the next pass.

Second, install the lower directions with fresh Gaussian lift columns $\mathbf{R}_\ell^{\leftarrow}$. Theorem 1 motivates coupling their upward transmission to the same intermediate neurons. The algorithm estimates this transmission from the corrected activations. After the row update, recompute $\mathbf{z}_\ell$ and set

$$\mathbf{s}_\ell = \mathbf{U}_\ell^\top \mathbf{g}^{\ell+1}, \qquad \widetilde{\mathbf{y}}_\ell = y\,\mathbf{s}_\ell, \qquad \mathbf{Q}_\ell = \widehat{\mathbb{E}}_n[\mathbf{z}_\ell \widetilde{\mathbf{y}}_\ell^\top]. \tag{10}$$

Each column of $\mathbf{Q}_\ell$ is a first moment under a modified label. Equivalently, $\mathbf{U}_\ell \mathbf{Q}_\ell^\top$ is the label-weighted column gradient projected onto the retained upper directions. Computing $\mathbf{Q}_\ell$ after the

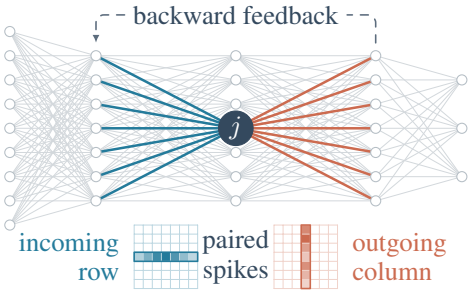

Figure 1: **Backward coupling and paired spikes.** The incoming row (blue) and outgoing column (red) meet at neuron j. Upper-layer sensitivity changes the feature learned below, whose activation changes upward transmission. Growing modes form paired rank-one spikes with shared neuron coefficients. The small matrices illustrate a mode concentrated on neuron j: an incoming row and its matching outgoing column.

row step includes features absent from the forward filter and uses the corrected nonlinear activations without linearizing them. The two updates are

$$\mathbf{W}_\ell \leftarrow \mathbf{W}_\ell + \mathbf{R}_\ell^{\leftarrow} \mathbf{\Gamma}_\ell^{\mathrm{row}} \mathbf{V}_\ell^\top, \qquad \mathbf{W}_{\ell+1} \leftarrow \mathbf{W}_{\ell+1} + \mathbf{U}_\ell \mathbf{\Gamma}_\ell^{\mathrm{col}} \mathbf{Q}_\ell^\top, \tag{11}$$

implemented sequentially at lines 7–10. The positive diagonal matrices $\mathbf{\Gamma}_\ell^{\mathrm{row}}$ and $\mathbf{\Gamma}_\ell^{\mathrm{col}}$ normalize correction amplitudes using a common scale β (Appendix F). Fresh row lifts and separate normalization modify the theorem's paired gradient flow while preserving coupling through the activations.

Finally, propagate the sensitivity to the boundary below. At the top, $\mathbf{g}^L$ is exact. Below, the exact sensitivity $\mathbf{g}^\ell$ applies the chain rule (2) through the corrected $\mathbf{W}_{\ell+1}$. A local estimate is also appealing. Since the column update (11) adds $\mathbf{U}_\ell \mathbf{\Gamma}_\ell^{\mathrm{col}} \mathbf{Q}_\ell^\top$ to $\mathbf{W}_{\ell+1}$, the new term to product $\mathbf{W}_{\ell+1}^\top \mathbf{g}^{\ell+1}$ is $\mathbf{Q}_\ell \mathbf{\Gamma}_\ell^{\mathrm{col}} \mathbf{s}_\ell$: of the upper sensitivity, the new columns transmit only the scores. The first-order estimate keeps this term, with its loading replaced by a linearized estimate $\mathbf{Q}_\ell^0$ from the pre-correction rows, and replaces the pre-existing contribution by its training mean:

$$\mathbf{g}^\ell = \frac{\sigma'(\mathbf{h}_\ell)}{\sqrt{p_\ell}} \odot \left(\widehat{\mathbb{E}}_n[\mathbf{W}_{\ell+1}^\top \mathbf{g}^{\ell+1}] + \mathbf{Q}_\ell^0 \, \mathbf{s}_\ell \right), \qquad \mathbf{Q}_\ell^0 \propto \mathbf{W}_\ell \mathbf{V}_\ell. \tag{12}$$

The matrices in this estimate are saved at the start of the pass; the lower gate uses the corrected activations. Each column of $\mathbf{Q}_\ell^0$ is rescaled to match the installed column's coordinate RMS. Both the mean substitution and the pre-correction loading are additional approximations. Appendix F specifies the normalization, the derivation, and the alternative signals. Subsequent passes repeat the sweep on the corrected network.

Over the rainbow — A first consequence is that this provide a concrete mechanism for the rainbow network of Guth et al. (2024). First, a forward Neural Lofi network *is* a rainbow network albeit a simple one. Indeed each layer is a linear dimension reduction $\mathbf{F}_\ell$ followed by white random features $\mathbf{R}_\ell$, its weight covariance $\mathbf{F}_\ell \mathbf{F}_\ell^\top$ has low rank, and the dependence between layers passes only through the filters, the random lifts being independent.

In their scattering-network experiments, Guth et al. (2024, Fig. 10) observe amplification of selected covariance directions that largely preserves the initial neuron configuration. Theorem 1 gives a local mechanism for this behavior: its growing modes amplify fixed, initialization-dependent neuron patterns $\mathbf{q}_i$. Backward coupling then goes beyond this picture: Shared coefficients link the weight covariances of consecutive layers: if their covariance is $\mathbf{C}_q$, the selected mode matrices $\mathbf{V}_\ell$ and $\mathbf{U}_\ell$ produce $\mathbf{V}_\ell \mathbf{C}_q \mathbf{V}_\ell^\top$ in incoming rows and $\mathbf{U}_\ell \mathbf{C}_q \mathbf{U}_\ell^\top$ in outgoing columns. They also couple the weights themselves, which the rainbow model takes to be independent across layers up to a rotation: the column of $\mathbf{W}_{\ell+1}$ that reads neuron j carries the coefficient q_{ji} tied to its row, and the lift of layer $\ell+1$ acquires the deterministic columns $\mathbf{u}_i$ next to its random ones. In this picture it is joint training that *paints the rainbow*: the layers above choose which covariance a layer grows, the growth starts from the initialization and keeps it, and the covariances of consecutive layers come out tied to each other through the neurons they share.

Coupled modes in linear networks — The coupling has an exact counterpart in the one-hidden-layer *linear* network $f(x) = \mathbf{W}_2 \mathbf{W}_1 x$, studied by Saxe et al. (2014, Sections 1.3 and 2). For unnormalized linear activations, whitened inputs and squared-loss gradient flow, aligned positive modes of the input–target correlation $\widehat{\mathbb{E}}_n[yx^\top] = \sum_i s_i u_i v_i^\top$ obey

$$\alpha_i := \|\mathbf{W}_1 v_i\|, \quad \beta_i := \|\mathbf{W}_2^\top u_i\|, \qquad \dot{\alpha}_i = \beta_i(s_i - \alpha_i \beta_i), \quad \dot{\beta}_i = \alpha_i(s_i - \alpha_i \beta_i). \tag{13}$$

The growing components scale as $e^{s_i t}$ while $\alpha_i \beta_i \ll s_i$; their product then saturates at s_i. For deeper networks with aligned singular directions, each layer's update contains the product of the mode amplitudes in all other layers. With D weight matrices and equal positive amplitudes across layers, the end-to-end mode strength m_i satisfies

$$\dot{m}_i = D\, m_i^{2-2/D}(s_i - m_i). \tag{14}$$

For $D = 2$, this gives exponential growth at small m_i, followed by saturation. For $D > 2$, the additional powers of m_i change the early amplification law. Appendix D derives both cases. Theorem 1 identifies the local counterpart for nonlinear networks: a small adjacent row and column amplify modes of a cross-layer correlation matrix determined by the surrounding features. With scalar targets the linear correlation has rank at most one; input-dependent nonlinear sensitivities can generate multiple cross-layer modes. Backward coupling thus lets the response learned above alter feature selection below, while the corrected activations determine transmission upward. Section 3 characterizes this selection in function space and follows the resulting kernel evolution.

3 Lessons from learning with backward coupling

Backward coupling has a function-space description parallel to, but richer than, Neural LoFi: a selection principle at fixed representation, followed by an evolution of the representation kernels.

A relevance–complexity trade-off for pairs — Neural LoFi selects a feature whose square correlates maximally with the label, $\max_{\|\varphi\|_{\mathcal{H}_{\ell-1}}=1} |\widehat{\mathbb{E}}_n[y\varphi^2]|$ (Dandi et al., 2026b). Backward coupling instead selects a pair: let $K_{\ell-1}(\mathbf{x}, \mathbf{x}') = \langle \mathbf{z}_{\ell-1}(\mathbf{x}), \mathbf{z}_{\ell-1}(\mathbf{x}')\rangle$ and $G_{\ell+1}(\mathbf{x}, \mathbf{x}') = \langle \mathbf{g}^{\ell+1}(\mathbf{x}), \mathbf{g}^{\ell+1}(\mathbf{x}')\rangle$, with RKHSs $\mathcal{H}_{\ell-1}$ and $\mathcal{H}^g_{\ell+1}$. The leading singular pair of $\mathbf{A}_\ell$ gives $\psi = \langle \mathbf{u}, \mathbf{g}^{\ell+1}\rangle$ and $\chi = \langle \mathbf{v}, \mathbf{z}_{\ell-1}\rangle$ solving

$$s_1(\mathbf{A}_\ell) = \max_{\|\psi\|_{\mathcal{H}^g_{\ell+1}} \le 1,\, \|\chi\|_{\mathcal{H}_{\ell-1}} \le 1} \widehat{\mathbb{E}}_n[y\,\psi\,\chi]. \tag{15}$$

The RKHS constraints favor features that are simple to express in each layer's geometry. Within these constraints, (15) selects a lower feature χ and an upper response ψ whose product is maximally correlated with y; neither feature needs to be predictive on its own. The paired row and column updates learn the lower cue in the same neurons that transmit the upper response, allowing the cue to modulate where that response is expressed. For example, a fur-texture cue could be coupled to an upper response associated with an eye or ear detector, strengthening that response where fur is present when their conjunction is informative about the label. This selective activation refines where higher-level features are expressed, and thereby the geometry of the representation.

Exact theory for Kernel dynamics with three layers — Joint training changes both the representations and the sensitivities that guide feature selection. Theorem 2 describes their coupled evolution in a three-layer network and its sequential infinite-width limit for a continuous version of backward coupling. In this version, the first-layer filtered features follow the selected lower mode, forward Neural LoFi updates the second-layer features as the lower kernel changes, and gradient flow trains the final readout. The two Gaussian lifts remain fixed, and successive column updates accumulate. Appendix C gives the algorithm, initialization and assumptions.

The evolution closes self-consistently in terms of the filtered features $\mathbf{h}^\star_{1,t}(x) = \mathbf{F}_{1,t}^\top \mathbf{z}_{0,t}(x) \in \mathbb{R}^{k_1}$ and $\mathbf{h}^\star_{2,t}(x) = \mathbf{F}_{2,t}^\top \mathbf{z}_{1,t}(x) \in \mathbb{R}^{k_2}$ of layers $1, 2$, the scalar output $h_{3,t}(x)$, the accumulated column coupling C_t and the response S_t. Let $\mathbf{r} \sim \mathcal{N}(0, I_{k_1})$ and $\eta \sim \mathcal{N}(0, I_{k_2})$ be the independent Gaussian lift rows. The coupling $C_t(\eta, \mathbf{r})$ records the accumulated column corrections between the two hidden layers. Define

$$S_t(\eta, s) = w_{3,t}(\eta)\sigma'(s), \qquad h_{3,t}(x) = \mathbb{E}_\eta[w_{3,t}(\eta) z_{2,t}(\eta, x)].$$

Here $w_{3,t}(\eta)$ is the readout coefficient of upper neuron η, and $S_t(\eta, s)$ gives its contribution to the output sensitivity at preactivation s. The domains of C_t and S_t are independent of network width. The same selected lower perturbation is added to every first-layer filtered coordinate, with one shared rate. The resulting feature maps are

$$\begin{aligned} h_{1,t}(\mathbf{r}, x) &= \mathbf{r}^\top \mathbf{h}^\star_{1,t}(x), \qquad z_{j,t} = \sigma(h_{j,t}), \\ h_{2,t}(\eta, x) &= \eta^\top \mathbf{h}^\star_{2,t}(x) + \mathbb{E}_{\mathbf{r}}[C_t(\eta, \mathbf{r}) z_{1,t}(\mathbf{r}, x)]. \end{aligned} \tag{16}$$

Evaluating the response at the current preactivation gives $g_t^2(\eta, x) = S_t(\eta, h_{2,t}(\eta, x))$. The kernels average products at x, x' over the same Gaussian lift row:

$$K_{1,t}(x, x') = \mathbb{E}_{\mathbf{r}}[z_{1,t}(\mathbf{r}, x) z_{1,t}(\mathbf{r}, x')],$$
$$K_{2,t}(x, x') = \mathbb{E}_{\eta}[z_{2,t}(\eta, x) z_{2,t}(\eta, x')], \qquad G_t(x, x') = \mathbb{E}_{\eta}[g_t^2(\eta, x) g_t^2(\eta, x')].$$

Theorem 2 (Informal: coupled dynamics in three layers)**.** *Under the assumptions of Appendix C, the discrete updates converge under the sequential limits $dt \to 0$, $p_1 \to \infty$ and $p_2 \to \infty$ to the following set of self-consistent equations, defined over the training inputs. At each time t, let (ψ_t, χ_t) denote the maximizer of* (15) *for the current sensitivity kernel G_t and the input kernel K_0, with value $\rho_t > 0$. The filtered features, coupling and response satisfy*

$$\dot{h}^{\star}_{1,t,a} = \gamma^{\mathrm{row}} \chi_t \quad (a = 1, \dots, k_1), \qquad \mathbf{h}^{\star}_{2,t} = \mathsf{FwdLoFi}_2(K_{1,t}, y), \tag{17}$$

$$\partial_t C_t(\eta, \mathbf{r}) = \frac{\gamma^{\mathrm{col}}}{\rho_t} \widehat{\mathbb{E}}_n[y \chi_t(X) g_t^2(\eta, X)] \widehat{\mathbb{E}}_n[y \psi_t(X) z_{1,t}(\mathbf{r}, X)],$$
$$\partial_t S_t(\eta, s) = \sigma'(s) \widehat{\mathbb{E}}_n[(y - h_{3,t}(X)) z_{2,t}(\eta, X)], \tag{18}$$

where $\mathsf{FwdLoFi}_2(K_{1,t}, y)$ gives the forward Neural LoFi features at layer 2 from the current kernel and labels. The fixed positive rates $\gamma^{\mathrm{row}}, \gamma^{\mathrm{col}}$ set the strengths of the row and column updates. The output satisfies

$$\begin{aligned} \dot{h}_{3,t}(x) = {} & \widehat{\mathbb{E}}_n[K_{2,t}(x, X)(y - h_{3,t}(X))] + \mathbb{E}_{\eta}[g_t^2(\eta, x)\eta]^{\top} \dot{\mathbf{h}}^{\star}_{2,t}(x) \\ & + \gamma^{\mathrm{col}} \psi_t(x) \widehat{\mathbb{E}}_n[K_{1,t}(x, X) y \psi_t(X)] \\ & + \mathbb{E}_{\eta, \mathbf{r}}[g_t^2(\eta, x) C_t(\eta, \mathbf{r}) \dot{z}_{1,t}(\mathbf{r}, x)]. \end{aligned} \tag{19}$$

Theorem 2 extends the layer-wise kernel constructions of forward Neural LoFi and AGOP-based Deep Recursive Feature Machines (Radhakrishnan et al., 2024; Beaglehole et al., 2024; Dandi et al., 2026b) to a continuous evolution in which upper and lower representations are *revised together*. The learned upper response changes which lower feature is selected, learning that feature then changes the upper kernel and the response itself. Thus the space in which features are selected evolves with training, in contrast with selection of hierarchical features from a prescribed family (Bach, 2008). Following this evolution requires only the current features, output, accumulated coupling C_t and response S_t. The equations are autonomous in these quantities, with earlier column updates retained in C_t, without the history integrals and two-time response functions of the dynamical mean-field formulation of Bordelon & Pehlevan (2022).

Because the corrections are selected by (15), this evolution can be studied through the signal-to-noise ratio of correlations between layers. An upper response can make a lower feature detectable through its correlation with the modified label $y\psi_t$, even when the feature had little correlation with y alone. For the current kernels, an informative leading population singular pair is reliably selected when its gap dominates the empirical operator error (Yu et al., 2015). Controlling this error along the trajectory would extend the emergence predictions of Neural LoFi (Dandi et al., 2026b) from successive layers to joint learning, using kernel concentration and spectral perturbation theory to predict when an upper response enables a new lower feature to emerge.

4 Mechanistic numerical experiments

We now illustrate the considerable advantages of backward feature correction with BackLofi over Lofi and layer-wise training in different situations.

Synthetic setting: The Deep Staircase — We first consider a class of synthetic hierarchical models (Nichani et al., 2024; Wang et al., 2023; Fu et al., 2025; Tabanelli et al., 2026; Dandi et al., 2026b; Wortsman-Zurich et al., 2026), adapted here to isolate the role of backward information. The construction is also related to staircase models (Abbe et al., 2021; 2022; 2023; Dandi et al., 2024a;b; Troiani et al., 2025; Tabanelli et al., 2025), with one key difference: after the visible features have been learned, the next useful directions become accessible only through a backward signal. We refer to this mechanism as the *deep staircase*.

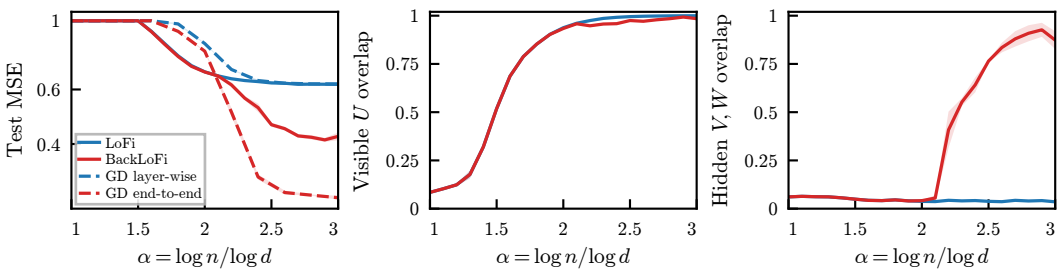

Figure 2: **Backward coupling unlocks the deep staircase.** Performance as a function of the sample exponent $\alpha = \log n / \log d$. *Left:* Test MSE. *Center:* Overlap with the visible first-level block U. *Right:* Overlap with the hidden blocks V, W. Both spectral methods recover the anchor; the backward correction also recovers the hidden blocks. The prediction panel additionally compares layer-wise and end-to-end gradient descent.

Specifically, we consider n samples $(\mathbf{x}_\mu, y_\mu)$ with Gaussian inputs $\mathbf{x}_\mu \sim \mathcal{N}(0, I_d)$ and scalar outputs $y_\mu \in \mathbb{R}$, and define $\alpha = \log n / \log d$. The first level contains $d_1 = p_U + p_V + p_W$ linear features, split into three independent blocks with distinct roles. The block $\mathbf{h}_U$ is the *visible anchor*, which can already be identified by the ordinary forward hierarchy. The blocks $\mathbf{h}_V$ and $\mathbf{h}_W$ are also relevant to the target, but are constructed to be invisible to the corresponding forward population statistic. Schematically,

$$\mathbf{x} \in \mathbb{R}^d \;\longrightarrow\; \mathbf{h} = (\mathbf{h}_U, \mathbf{h}_V, \mathbf{h}_W) \in \mathbb{R}^{d_1}, \qquad \begin{cases} \mathbf{h}_U \in \mathbb{R}^{p_U} & \text{visible feature,} \\ \mathbf{h}_V \in \mathbb{R}^{p_V} & \text{hidden feature,} \\ \mathbf{h}_W \in \mathbb{R}^{p_W} & \text{hidden feature,} \end{cases} \longrightarrow\; y \in \mathbb{R}. \tag{20}$$

Up to centering and variance normalization, the target has the form

$$y = g(A(\mathbf{h}_U) + B(\mathbf{h}_U)\, S(\mathbf{h}_V, \mathbf{h}_W)), \tag{21}$$

where A and B are scalar features of the visible block and $S(\mathbf{h}_V, \mathbf{h}_W) = \mathrm{sign}(\mathbf{h}_V^\top C \mathbf{h}_W)$ contains the hidden interaction. Appendix E gives the complete teacher and its population analysis.

We can now discuss the advantage of the backward pass. At population level, the second-order forward statistic recovers the visible U-block but has no component on the hidden V/W subspace. Once the visible hierarchy has been learned, an upper sensitivity s_U removes this cancellation:

$$\mathbb{E}\big[y\, \mathbf{h}_V \mathbf{h}_W^\top\big] = 0, \qquad \mathbb{E}\big[y\, s_U\, \mathbf{h}_V \mathbf{h}_W^\top\big] \neq 0. \tag{22}$$

Figure 2 exhibits the corresponding separation for both gradient descent and its spectral surrogate: joint training and BackLoFi recover hidden directions missed by layer-wise GD and Forward LoFi, respectively. Theorem 3 further proves subspace recovery with oracle sensitivities for $n \gg d \max\{p_U, p_V, p_W\}$, with the learned-sensitivity extension discussed in Appendix E. This provides a tractable instance in which backward information enables feature discovery unavailable to the forward hierarchy.

Image classification — We now test whether the coupling isolated in Section 2 accounts for what joint training learns on real data. We run Algorithm 2 on a fully connected and on a convolutional multi-layer network on binary CIFAR-10, and compare it at matched width with forward Neural LoFi, with the random-features network of the same architecture, and with full-batch gradient descent (Figure 3; settings in Appendix H).

One can read Fig. 3 in three steps, each holding for both architectures. First, BackLoFi improves on forward LoFi at every training-set size and at equal width. The information that the layers above return to a boundary selects directions that the forward filter, built from the same representation, did not find. The gain is largest when data are scarce: on the convolutional network, BackLoFi trained on $5{,}000$ images comes within 0.15 points of forward Neural LoFi trained on $50{,}000$. The replacement of features builds up over a few backward passes, until about half of the forward columns have been replaced (see Appendix G).

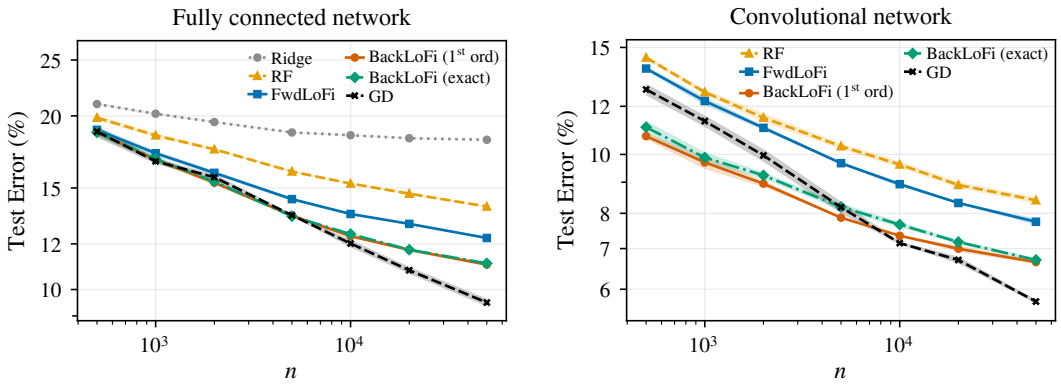

Figure 3: **Backward coupling improves on forward Neural LoFi and approaches gradient descent.** Test error vs number of training samples n on binary CIFAR-10 (animal vs. vehicle), at matched width, for a fully connected network (*left*) and a convnet (*right*). Ridge is a linear regression on the pixels, RF the random-features network of the same architecture, FwdLoFi forward Neural LoFi, BackLoFi Algorithm 2 with the first-order or the exact backward signal, and GD full-batch gradient descent on the same architecture. Curves are means over seeds and bands one std error; Spectral configurations and GD hyperparameters are selected on mean test error (see App. H).

Second, the exact and the first-order backward signals give nearly the same curves. Of the sensitivity of the layers above, the first-order signal keeps only the scores $\mathbf{s}_\ell$, one number per selected mode. What the layers above pass down is therefore captured by these r_ℓ numbers per sample.

Third, BackLoFi follows gradient descent on the same architecture. It matches it up to $n = 5{,}000$ on the fully connected network and is ahead of it on the convolutional one, by two points at $n = 500$. Gradient descent pulls ahead only on the largest training sets, where BackLoFi still recovers about half of the gap between forward Neural LoFi and joint training. This is consistent with joint training using more than a few coupled modes per boundary once data are abundant. The same mechanism appeared in the deep staircase above, where directions invisible to the forward statistic become visible through the response of the layers above.

Over most of the sample range, then, what joint training adds to layer-wise learning is a low-rank coupling between the rows of a layer and the columns that read them. Installing a few such modes per boundary, with no gradient step and using only the scores of the layers above, reproduces it.

5 Conclusion

Backward Neural LoFi extends layer-wise spectral learning by coupling lower features to upper responses: information learned above changes which directions are selected below. The resulting feature–response pairs admit a variational characterization, while their local dynamics provide a mechanism for joint amplification and structured weight covariances. For a continuous three-layer variant, we prove convergence to a self-consistent evolution of the nonlinear features, kernels, and output. The hierarchical experiment illustrates hidden-feature discovery, and the image experiments show improvements over forward learning at matched width. Controlling repeated corrections beyond the local regime and establishing finite-sample guarantees for the implemented algorithm with learned upper scores remain open. We believe these results help to open new directions into the understanding of the *unreasonable efficiency* of learning with deep networks (LeCun et al., 2015).

ETHICS STATEMENT

This work is theoretical and uses only a public image benchmark (CIFAR-10); it involves no human subjects, personal data, or foreseeable harmful application, and raises no specific concern under the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

Assumptions and proofs of all theoretical results are given in the appendices, the implemented algorithm is specified in Appendix F and the settings of every figure in Appendix H. The code reproducing all figures will be released with the final version.

AI USE STATEMENT

In this work, we used generative AI tools to assist with developing and checking individual steps of the proofs, drafting and revising parts of the proofs, and developing and debugging the experimental code. Additionally, we used generative AI tools to improve the organization and exposition of the manuscript, prepare and revise text, figure captions and layouts, and prepare plots from numerical results. The authors reviewed and revised the AI-assisted work and take responsibility for the final content, including all mathematical claims, proofs, text, code, and other artifacts.

REFERENCES

Emmanuel Abbe, Enric Boix-Adsera, Matthew S Brennan, Guy Bresler, and Dheeraj Nagaraj. The staircase property: How hierarchical structure can guide deep learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 26989–27002. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/e2db7186375992e729165726762cb4c1-Paper.pdf.

Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks. In Po-Ling Loh and Maxim Raginsky (eds.), *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 4782–4887. PMLR, 02–05 Jul 2022.

Emmanuel Abbe, Enric Boix-Adserà, and Theodor Misiakiewicz. SGD learning on neural networks: leap complexity and saddle-to-saddle dynamics. In Gergely Neu and Lorenzo Rosasco (eds.), *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 2552–2623. PMLR, 12–15 Jul 2023.

Zeyuan Allen-Zhu and Yuanzhi Li. Backward feature correction: How deep learning performs deep (hierarchical) learning. In *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 4598–4598, 2023. URL https://proceedings.mlr.press/v195/allen-zhu23a.html.

Francis Bach. The quest for adaptivity. *Machine Learning Research Blog*, 155, 2021.

Francis R. Bach. Exploring large feature spaces with hierarchical multiple kernel learning. In *Advances in Neural Information Processing Systems*, volume 21, pp. 105–112, 2008. URL https://papers.nips.cc/paper_files/paper/2008/hash/a4a042cf4fd6bfb47701cbc8a1653ada-Abstract.html.

Jinho Baik, Gérard Ben Arous, and Sandrine Péché. Phase transition of the largest eigenvalue for nonnull complex sample covariance matrices. *Annals of probability*, 33(5):1643–1697, 2005.

Lorenzo Bardone and Sebastian Goldt. Sliding down the stairs: how correlated latent variables accelerate learning with neural networks. In *International Conference on Machine Learning*, volume 235, pp. 3024–3045, 2024.

Daniel Beaglehole, Adityanarayanan Radhakrishnan, Parthe Pandit, and Mikhail Belkin. Mechanism of feature learning in convolutional neural networks. *arXiv preprint arXiv:2309.00570*, 2023.

Daniel Beaglehole, Peter Súkeník, Marco Mondelli, and Mikhail Belkin. Average gradient outer product as a mechanism for deep neural collapse. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ec045a5ca2d8cfc528591b4c34296370-Abstract-Conference.html.

Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon. Decoupled greedy learning of cnns. In *International Conference on Machine Learning*, pp. 736–745, 2020.

Gérard Ben Arous, Reza Gheissari, and Aukosh Jagannath. Online stochastic gradient descent on non-convex losses from high-dimensional inference. *Journal of Machine Learning Research*, 22 (106):1–51, 2021.

Gérard Ben Arous, Cédric Gerbelot, and Vanessa Piccolo. Stochastic gradient descent in high dimensions for multi-spiked tensor pca. *Communications on Pure and Applied Mathematics*, 79(10): 2291–2369, 2026. doi: https://doi.org/10.1002/cpa.70056. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.70056.

Enric Boix-Adserà, Neil Rohit Mallinar, James B. Simon, and Mikhail Belkin. FACT: A first-principles alternative to the neural feature ansatz for how networks learn representations. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=j4964wtJMz.

Fabrizio Boncoraglio, Vittorio Erba, Emanuele Troiani, Yizhou Xu, Florent Krzakala, and Lenka Zdeborová. Single-head attention in high dimensions: A theory of generalization, weights spectra, and scaling laws. *arXiv preprint arXiv:2509.24914*, 2025.

Tony Bonnaire, Giulio Biroli, and Chiara Cammarota. The role of the time-dependent hessian in high-dimensional optimization. *Journal of Statistical Mechanics: Theory and Experiment*, 2025 (8):083401, 2025.

Blake Bordelon and Cengiz Pehlevan. Self-consistent dynamical field theory of kernel evolution in wide neural networks. In *Advances in Neural Information Processing Systems*, volume 35, pp. 32240–32256, 2022. URL https://arxiv.org/abs/2205.09653.

Joan Bruna. The multiscale single-index model: A stylized model for hierarchical feature learning. *arXiv preprint arXiv:2607.03347*, 2026. URL https://arxiv.org/abs/2607.03347.

Francesco Cagnetta, Leonardo Petrini, Umberto M. Tomasini, Alessandro Favero, and Matthieu Wyart. How deep neural networks learn compositional data: The random hierarchy model. *Physical Review X*, 14(3):031001, 2024. doi: 10.1103/PhysRevX.14.031001. URL https://arxiv.org/abs/2307.02129.

Taehun Cha, Daniel Beaglehole, Adityanarayanan Radhakrishnan, and Donghun Lee. The weight gram matrix captures sequential feature linearization in deep networks. *arXiv preprint arXiv:2605.06258*, 2026.

Lénaïc Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Alex Damian, Loucas Pillaud-Vivien, Jason Lee, and Joan Bruna. Computational-statistical gaps in gaussian single-index models. In *The Thirty Seventh Annual Conference on Learning Theory*, pp. 1262–1262. PMLR, 2024.

Alex Damian, Jason Lee, and Joan Bruna. The generative leap: Tight sample complexity for efficiently learning gaussian multi-index models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 28276–28311. Curran Associates, Inc., 2025. doi: 10.52202/085713-0953.

Yatin Dandi, Florent Krzakala, Bruno Loureiro, Luca Pesce, and Ludovic Stephan. How two-layer neural networks learn, one (giant) step at a time. *Journal of Machine Learning Research*, 25(349): 1–65, 2024a. URL http://jmlr.org/papers/v25/23-1543.html.

Yatin Dandi, Emanuele Troiani, Luca Arnaboldi, Luca Pesce, Lenka Zdeborová, and Florent Krzakala. The benefits of reusing batches for gradient descent in two-layer networks: Breaking the curse of information and leap exponents. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9991–10016. PMLR, 21–27 Jul 2024b.

Yatin Dandi, Luca Pesce, Lenka Zdeborová, and Florent Krzakala. The computational advantage of depth in learning high-dimensional hierarchical targets. *Advances in Neural Information Processing Systems*, 38:160705–160764, 2026a.

Yatin Dandi, Matteo Vilucchio, Luca Arnaboldi, Hugo Tabanelli, and Florent Krzakala. Deep learning as Neural Low-Degree Filtering: A spectral theory of hierarchical feature learning. *arXiv preprint arXiv:2605.13612*, 2026b. URL https://arxiv.org/abs/2605.13612.

Leonardo Defilippis, Florent Krzakala, Bruno Loureiro, and Antoine Maillard. Optimal scaling laws in learning hierarchical multi-index models. *arXiv preprint arXiv:2602.05846*, 2026a.

Leonardo Defilippis, Yizhou Xu, Julius Girardin, Vittorio Erba, Emanuele Troiani, Lenka Zdeborová, Bruno Loureiro, and Florent Krzakala. Scaling laws and spectra of shallow neural networks in the feature learning regime. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=Q3yLIIkt7z.

Samuel F Edwards and Raymund C Jones. The eigenvalue spectrum of a large symmetric random matrix. *Journal of Physics A: Mathematical and General*, 9(10):1595–1603, 1976.

Vittorio Erba, Emanuele Troiani, Lenka Zdeborová, and Florent Krzakala. The nuclear route: Sharp asymptotics of erm in overparameterized quadratic networks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Cong Fang, Jason Lee, Pengkun Yang, and Tong Zhang. Modeling from features: a mean-field framework for over-parameterized deep neural networks. In *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 1887–1936, 2021. URL https://proceedings.mlr.press/v134/fang21a.html.

Hengyu Fu, Zihao Wang, Eshaan Nichani, and Jason D. Lee. Learning hierarchical polynomials of multiple nonlinear features with three-layer networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Wulfram Gerstner, Marco Lehmann, Vasiliki Liakoni, Dane Corneil, and Johanni Brea. Eligibility traces and plasticity on behavioral time scales: experimental support of neohebbian three-factor learning rules. *Frontiers in neural circuits*, 12:53, 2018.

Florentin Guth, Brice Ménard, Gaspar Rochette, and Stéphane Mallat. A rainbow in deep network black boxes. *Journal of Machine Learning Research*, 25(350):1–59, 2024. URL https://jmlr.org/papers/v25/23-1573.html.

Karl Hajjar, Lénaïc Chizat, and Christophe Giraud. Training integrable parameterizations of deep neural networks in the infinite-width limit. *Journal of Machine Learning Research*, 25(196): 1–130, 2024. URL https://www.jmlr.org/papers/v25/21-1260.html.

Bernd Illing, Wulfram Gerstner, and Johanni Brea. Biologically plausible deep learning—but how far can we go with shallow networks? *Neural Networks*, 118:90–101, 2019.

Bernd Illing, Jean Ventura, Guillaume Bellec, and Wulfram Gerstner. Local plasticity rules can learn deep representations using self-supervised contrastive predictions. In *Advances in Neural Information Processing Systems*, volume 34, pp. 30365–30379, 2021.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Daniel Kunin, Giovanni Luca Marchetti, Feng Chen, Dhruva Karkada, James B. Simon, Michael R. DeWeese, Surya Ganguli, and Nina Miolane. Alternating gradient flows: A theory of feature learning in two-layer neural networks. In *Advances in Neural Information Processing Systems*, 2025.

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, pp. 9346–9360, 2020.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi: 10.1038/nature14539.

Dong-Hyun Lee, Saizheng Zhang, Asja Fischer, and Yoshua Bengio. Difference target propagation. In *Machine Learning and Knowledge Discovery in Databases*, pp. 498–515. Springer, 2015.

Jaehoon Lee, Lechao Xiao, Samuel S. Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

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: 13276, 2016.

Yue M Lu and Gen Li. Phase transitions of spectral initialization for high-dimensional non-convex estimation. *Information and Inference: A Journal of the IMA*, 9(3):507–541, 2020.

Antoine Maillard, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborová. Phase retrieval in high dimensions: Statistical and computational phase transitions. *Advances in Neural Information Processing Systems*, 33:11071–11082, 2020.

Giovanni Luca Marchetti, Daniel Kunin, Adele Myers, Francisco Acosta, and Nina Miolane. Sequential group composition: A window into the mechanics of deep learning. In *International Conference on Machine Learning*, 2026.

Charles H Martin and Michael W Mahoney. Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *Journal of Machine Learning Research*, 22(165):1–73, 2021.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018. doi: 10.1073/pnas.1806579115.

Marco Mondelli and Andrea Montanari. Fundamental limits of weak recovery with applications to phase retrieval. In *Conference On Learning Theory*, pp. 1445–1450. PMLR, 2018.

Andrea Montanari and Zihao Wang. Phase transitions for feature learning in neural networks. *arXiv preprint arXiv:2602.01434*, 2026.

Phan-Minh Nguyen and Huy Tuan Pham. A rigorous framework for the mean field limit of multilayer neural networks, 2020. URL `https://arxiv.org/abs/2001.11443`.

Eshaan Nichani, Alex Damian, and Jason D. Lee. Provable guarantees for nonlinear feature learning in three-layer neural networks. In *Advances in Neural Information Processing Systems*, volume 36, pp. 10828–10875, 2023.

Eshaan Nichani, Alex Damian, and Jason D Lee. Provable guarantees for nonlinear feature learning in three-layer neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

Arild Nøkland. Direct feedback alignment provides learning in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Arild Nøkland and Lars Hiller Eidnes. Training neural networks with local error signals. In *International Conference on Machine Learning*, pp. 4839–4850, 2019.

Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273, 1982.

Adityanarayanan Radhakrishnan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin. Mechanism for feature learning in neural networks and backpropagation-free machine learning models. *Science*, 383(6690):1461–1467, 2024. doi: 10.1126/science.adi5639.

Adityanarayanan Radhakrishnan, Mikhail Belkin, and Dmitriy Drusvyatskiy. Linear recursive feature machines provably recover low-rank matrices. *Proceedings of the National Academy of Sciences*, 122(13):e2411325122, 2025. doi: 10.1073/pnas.2411325122.

Yunwei Ren, Yatin Dandi, Florent Krzakala, and Jason D. Lee. Provable learning of random hierarchy models and hierarchical shallow-to-deep chaining. In *Proceedings of Thirty Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 5556–5597. PMLR, 2026a. URL https://proceedings.mlr.press/v336/ren26a.html.

Yunwei Ren, Zihao Wang, and Jason D. Lee. Algorithmic separation between constant-depth and logarithmic-depth neural networks. *arXiv preprint arXiv:2607.25200*, 2026b. URL https://arxiv.org/abs/2607.25200.

Valentina Ros, Gérard Ben Arous, Giulio Biroli, and Chiara Cammarota. Complex energy landscapes in spiked-tensor and simple glassy models: Ruggedness, arrangements of local minima, and phase transitions. *Physical Review X*, 9(1):011003, 2019.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0.

Terence D Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural networks*, 2(6):459–473, 1989.

Stefano Sarao Mannelli, Giulio Biroli, Chiara Cammarota, Florent Krzakala, and Lenka Zdeborová. Who is afraid of big bad minima? analysis of gradient-flow in spiked matrix-tensor models. *Advances in neural information processing systems*, 32, 2019.

Stefano Sarao Mannelli, Giulio Biroli, Chiara Cammarota, Florent Krzakala, Pierfrancesco Urbani, and Lenka Zdeborová. Marvels and pitfalls of the langevin algorithm in noisy high-dimensional inference. *Physical Review X*, 10(1):011057, 2020.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv:1312.6120, 2014. URL https://arxiv.org/abs/1312.6120.

Hugo Tabanelli, Pierre Mergny, Lenka Zdeborová, and Florent Krzakala. Computational thresholds in multi-modal learning via the spiked matrix-tensor model. *arXiv preprint arXiv:2506.02664*, 2025.

Hugo Tabanelli, Yatin Dandi, Luca Pesce, and Florent Krzakala. Deep learning of compositional targets with hierarchical spectral methods. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/61650.

Emanuele Troiani, Yatin Dandi, Leonardo Defilippis, Lenka Zdeborová, Bruno Loureiro, and Florent Krzakala. Fundamental computational limits of weak learnability in high-dimensional multi-index models. In Yingzhen Li, Stephan Mandt, Shipra Agrawal, and Emtiyaz Khan (eds.), *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 2467–2475. PMLR, 03–05 May 2025.

Cédric Villani. *Topics in Optimal Transportation*, volume 58 of *Graduate Studies in Mathematics*. American Mathematical Society, 2003.

Matteo Vilucchio, Yatin Dandi, Matéo Pirio Rossignol, Cédric Gerbelot, and Florent Krzakala. Asymptotics of non-convex generalized linear models in high-dimensions: A proof of the replica formula. *arXiv preprint arXiv:2502.20003*, 2025.

Zihao Wang, Eshaan Nichani, and Jason D Lee. Learning hierarchical polynomials with three-layer neural networks. *arXiv preprint arXiv:2311.13774*, 2023.

Arie Wortsman-Zurich, Hugo Tabanelli, Yatin Dandi, Florent Krzakala, and Bruno Loureiro. Scaling laws from sequential feature recovery: A solvable hierarchical model, 2026. URL https://arxiv.org/abs/2605.14567.

Greg Yang and Edward J. Hu. Tensor Programs IV: Feature learning in infinite-width neural networks. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11727–11737, 2021. URL https://proceedings.mlr.press/v139/yang21c.html.

Yi Yu, Tengyao Wang, and Richard J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015. doi: 10.1093/biomet/asv008. URL https://arxiv.org/abs/1405.0680.

Bohan Zhang, Zihao Wang, Hengyu Fu, and Jason D. Lee. Neural networks learn generic multi-index models near information-theoretic limit. *arXiv preprint arXiv:2511.15120*, 2025.

A FURTHER RELATED WORKS

We collect here the lines of work that the main text touches only briefly, organized by their relation to backward coupling.

Feature correction by the layers above — The closest antecedent is the *backward feature correction* of Allen-Zhu & Li (2023), where jointly trained upper layers refine the representation learned below. Backward coupling isolates a narrower and more explicit effect: an upper response, entering as a modified label, changes which directions the lower spectral step can see at all. The same reversal of the forward order appears in a different guise in the alternating-gradient-flow analyses of Kunin et al. (2025), where features of a two-layer network activate one after the other through plateaus and sharp drops, and in the sequential learning of compositional tasks of Marchetti et al. (2026).

Multi-pass learning and staircases — A single low-degree statistic of the input misses targets with information or leap exponent above two (Ben Arous et al., 2021; 2026; Damian et al., 2024; Troiani et al., 2025). Repeated passes over the data change this: reusing batches transforms the effective label and lets gradient descent escape these constraints (Dandi et al., 2024b), and staircase-type targets are learned one correlation at a time, each step being unlocked by the previous one (Abbe et al., 2021; 2023; Bardone & Goldt, 2024; Tabanelli et al., 2025). The generative-exponent picture (Damian et al., 2025) quantifies what label transformations can achieve. BackLoFi sits in this family with a specific choice of transformation, the sample-wise response of the fitted layers above, and a specific place where it acts, the spectral step of an earlier layer.

Hierarchical learning and depth — Dandi et al. (2026a) study Gaussian hierarchical single- and multi-index targets, showing how successive feature learning reduces the effective dimension and yields a sample-complexity advantage over shallow networks. In a related compositional setting, Tabanelli et al. (2026) recover latent features with layer-wise spectral estimators and establish sample-complexity gains over shallow kernel methods. Wortsman-Zurich et al. (2026) derive recovery thresholds for individual features with power-law strengths: as the sample size increases, successively weaker features become detectable, producing power-law decay of the prediction error. For a multiscale single-index model, Bruna (2026) proves recovery by online SGD on a correlation objective with all layers evolving on the same timescale.

For discrete compositional data, Cagnetta et al. (2024) introduce the Random Hierarchy Model and relate the emergence of representations invariant to equivalent feature combinations to the detectability of correlations between input patches and labels. Ren et al. (2026a) prove efficient learning of these hierarchies under nondegeneracy assumptions by training a deep convolutional network layer by layer. For Boolean targets with hierarchical Fourier structure, Ren et al. (2026b) give a layer-wise coordinate-descent algorithm for logarithmic-depth networks and prove an approximation separation from constant-depth networks with polynomial width, regular activations, and controlled spectral norms.

Hessian and spectral learning — Label-weighted second-order operators drive spectral initializations and recovery in single- and multi-index estimation (Lu & Li, 2020; Mondelli & Montanari, 2018; Maillard et al., 2020; Vilucchio et al., 2025), and describe the early dynamics of shallow networks through Hessian-like objects (Bonnaire et al., 2025; Zhang et al., 2025; Montanari & Wang, 2026; Defilippis et al., 2026a); the selection of spectral outliers against a bulk follows the BBP/EJ mechanism (Baik et al., 2005; Edwards & Jones, 1976), as in spiked matrix–tensor models (Ros et al., 2019; Sarao Mannelli et al., 2019; 2020). Neural LoFi (Dandi et al., 2026b) applies one such operator per layer. The cross-layer matrix $\mathbf{A}_\ell$ of this paper correlates two representations through the labels; its leading singular pairs select the coupled features.

Spectral structure of trained weights — Trained networks show low-rank, non-random structure in their weights: spikes over a bulk in quadratic and diagonal networks (Defilippis et al., 2026b), nuclear-norm-type capacity control in overparameterized quadratic networks (Erba et al., 2025), similar spectra in attention layers (Boncoraglio et al., 2025), and structured weight spectra across architectures (Martin & Mahoney, 2021). The rainbow model (Guth et al., 2024) organizes these observations into layers of random features with learned low-rank covariances, assumed independent

across layers up to a rotation. Section 3 shows what backward coupling adds: linked covariances and a neuron-by-neuron dependence between the rows of one layer and the columns of the next.

Feature learning from gradients of a fitted predictor — Recursive feature machines update the feature metric with the average gradient outer product (AGOP) of a fitted predictor (Radhakrishnan et al., 2024; 2025; Beaglehole et al., 2023; Boix-Adserà et al., 2026; Cha et al., 2026). These methods also feed information from the output back into the features, through the gradient of a predictor that must first be fitted. In backward coupling, the upper sensitivity $\mathbf{g}^{\ell+1}$ plays a similar role: together with y, it weights a first moment of the fixed lower representation, producing the cross-layer matrix $\mathbf{A}_\ell$.

Local and backpropagation-free training — Layer-wise and local training rules avoid an exact backward pass: feedback alignment and its direct variants replace the transposed weights by random matrices (Lillicrap et al., 2016; Nøkland, 2016; Launay et al., 2020), local losses train layers separately (Nøkland & Eidnes, 2019; Belilovsky et al., 2020), target propagation passes layer-wise targets (Lee et al., 2015), and label-modulated Hebbian rules approximate spectral operators (Oja, 1982; Sanger, 1989; Gerstner et al., 2018; Illing et al., 2019; 2021). The backward signal of BackLoFi is itself low-rank and approximate, carried by a few scores per sample, in the spirit of these rules.

B Coupled mode amplification: proof of Theorem 1

We now give the full formal statement and assumptions for Theorem 1. Theorem B.1 specifies the initialization scales and training interval, and bounds the error between gradient flow and the coupled linear dynamics used in the main text. Its mode formulas identify the shared growing component of each incoming row and matching outgoing column.

B.1 Assumptions and formal statement

We use the network notation of Section 2. Fix a neuron j in layer $\ell < L$, with incoming row $\mathbf{w}_{\ell,j}$ and matching outgoing column $\boldsymbol{\omega}_{\ell+1,j}$. We train this pair jointly and keep all other weights frozen. With $x = \mathbf{z}_{\ell-1}$, the next preactivation is

$$\mathbf{h}_{\ell+1}(x) = \mathbf{h}_{\ell+1,0}(x) + \frac{1}{\sqrt{p_\ell}}\boldsymbol{\omega}_{\ell+1,j}\,\sigma(\langle \mathbf{w}_{\ell,j}, x\rangle). \tag{23}$$

Here $\mathbf{h}_{\ell+1,0}$ is the fixed contribution of the remaining neurons, which may already have been trained.

Assumption 1 (Joint training of one pair in a fixed network)**.** The dataset, inputs x, contextual preactivations $\mathbf{h}_{\ell+1,0}(x)$, depth and widths are fixed. The selected pair $(\mathbf{w}_{\ell,j}, \boldsymbol{\omega}_{\ell+1,j})$ follows equal-rate Euclidean gradient flow for $\frac{1}{2}\widehat{\mathbb{E}}_n[(y-f)^2]$; all other weights remain frozen during the entire training interval. The subsequent layers have no biases and the output intercept is zero.

Assumption 2 (Activation)**.** The activation is C^3, with bounded derivatives on the relevant bounded preactivation sets, $\sigma(0) = 0$, and $\sigma'(0) > 0$.

Assumption 3 (Layer-wise vanishing initialization)**.** Initialize $\mathbf{w}_{\ell,j}(0) = \alpha_\ell \overline{\mathbf{w}}_{\ell,j}$ and $\boldsymbol{\omega}_{\ell+1,j}(0) = \alpha_{\ell+1}\overline{\boldsymbol{\omega}}_{\ell+1,j}$. The frozen subsequent weights are $\mathbf{W}_k = \alpha_k \overline{\mathbf{W}}_k$ for $k = \ell+2, \dots, L$, and $\mathbf{a}_L = \alpha_{L+1} a_0$. The normalized initial vectors $\overline{\mathbf{w}}_{\ell,j}$, $\overline{\boldsymbol{\omega}}_{\ell+1,j}$, matrices $\overline{\mathbf{W}}_k$ and readout a_0 are fixed. The scales are positive and satisfy

$$\alpha_\ell \longrightarrow 0, \qquad \alpha_{k+1} = o(\alpha_k) \quad (k = \ell, \dots, L). \tag{24}$$

This is the scale hierarchy of Dandi et al. (2026b, Assumption 1), applied to the selected pair and the frozen weights above it; the previously learned context may remain nonzero.

To obtain the linear approximation, evaluate the upper sensitivity at $\mathbf{h}_{\ell+1,0}(x)$, with the selected neuron's contribution removed. This gives the fixed sensitivity and correlation matrix

$$\mathbf{g}_0(x) = \left.\frac{\partial f}{\partial \mathbf{h}_{\ell+1}}\right|_{\mathbf{h}_{\ell+1}=\mathbf{h}_{\ell+1,0}(x)}, \qquad \mathbf{A}_\ell = \widehat{\mathbb{E}}_n[y\mathbf{g}_0 x^\top]. \tag{25}$$

The activation derivative in the row and column updates contributes the factor $\sigma'(0)/\sqrt{p_\ell}$. The largest growth rate of their linear approximation is therefore

$$\lambda = \frac{\sigma'(0)}{\sqrt{p_\ell}} \|\mathbf{A}_\ell\|_2.$$

Both $\mathbf{A}_\ell$ and λ depend on the initialization scales; they remain fixed during training.

The sensitivity contains one weight factor from each subsequent layer, including the readout, so its size is bounded by a constant times $\prod_{k=\ell+2}^{L+1} \alpha_k$. We assume that averaging against the labels and inputs leaves a correlation of this same order:

$$\lambda \geq c \prod_{k=\ell+2}^{L+1} \alpha_k \quad \text{for some fixed } c > 0 \text{ and all sufficiently small scales.} \tag{26}$$

This condition excludes cancellation of the leading correlation that drives the coupled growth.

Theorem B.1 (Formal version of Theorem 1)**.** *Under Assumptions 1–3 and* (26)*, let the linear approximation solve*

$$\dot{\mathbf{w}}^{\mathrm{lin}}_{\ell,j} = \frac{\sigma'(0)}{\sqrt{p_\ell}} \mathbf{A}_\ell^\top \boldsymbol{\omega}^{\mathrm{lin}}_{\ell+1,j}, \qquad \dot{\boldsymbol{\omega}}^{\mathrm{lin}}_{\ell+1,j} = \frac{\sigma'(0)}{\sqrt{p_\ell}} \mathbf{A}_\ell \mathbf{w}^{\mathrm{lin}}_{\ell,j}$$

from the same initialization. Choose a time horizon $T \geq 0$, which may depend on the initialization scales, such that

$$\alpha_\ell e^{\lambda T} \longrightarrow 0.$$

This condition keeps the linearly amplified weights small throughout $[0, T]$, so that the nonlinear corrections remain lower order. There is a constant C, independent of the scales and time, such that for sufficiently small initialization scales, uniformly for $0 \leq t \leq T$,

$$\begin{aligned} \|(\mathbf{w}_{\ell,j}(t), \boldsymbol{\omega}_{\ell+1,j}(t))\| &\leq C\alpha_\ell e^{\lambda t}, \\ \|(\mathbf{w}_{\ell,j}(t) - \mathbf{w}^{\mathrm{lin}}_{\ell,j}(t), \boldsymbol{\omega}_{\ell+1,j}(t) - \boldsymbol{\omega}^{\mathrm{lin}}_{\ell+1,j}(t))\| &\leq C\alpha_\ell^2 e^{\lambda t}(e^{\lambda t} - 1). \end{aligned} \tag{27}$$

The pair norm is Euclidean. In particular, the error divided by $\alpha_\ell e^{\lambda t}$ tends uniformly to zero. For each singular triple, $\mathbf{A}_\ell v_i = s_i u_i$ and $\mathbf{A}_\ell^\top u_i = s_i v_i$, write $c^{\pm}_{ji} = \frac{1}{2}(\langle \mathbf{w}_{\ell,j}(0), v_i\rangle \pm \langle \boldsymbol{\omega}_{\ell+1,j}(0), u_i\rangle)$. Then

$$\begin{aligned} \langle \mathbf{w}_{\ell,j}(t), v_i\rangle &= c^+_{ji} e^{\sigma'(0) s_i t/\sqrt{p_\ell}} + c^-_{ji} e^{-\sigma'(0) s_i t/\sqrt{p_\ell}} + \epsilon^w_{ji}(t), \\ \langle \boldsymbol{\omega}_{\ell+1,j}(t), u_i\rangle &= c^+_{ji} e^{\sigma'(0) s_i t/\sqrt{p_\ell}} - c^-_{ji} e^{-\sigma'(0) s_i t/\sqrt{p_\ell}} + \epsilon^\omega_{ji}(t), \end{aligned} \tag{28}$$

with both errors satisfying the second bound in (27)*.*

The two weights share the growing amplitude c^+_{ji}, as stated in Theorem 1. Since the outgoing seed is smaller, $c^+_{ji} = \frac{1}{2}\alpha_\ell \langle \overline{\mathbf{w}}_{\ell,j}, v_i\rangle + o(\alpha_\ell)$. The error bound is an additive estimate for each mode, measured against the largest growth scale $\alpha_\ell e^{\lambda t}$.

Remark (gradient descent). We use gradient flow for convenience. For full-batch gradient descent with step size Δt, the linear mode factors after m steps are $(1 \pm \Delta t\, \sigma'(0) s_i/\sqrt{p_\ell})^m$. The same remainder estimates and discrete Grönwall give the corresponding bounds relative to these linear iterates. If $\Delta t\, \lambda < 1$, replacing the factors by their exponentials introduces relative error $O(\Delta t\, \lambda^2 T)$. This error vanishes for fixed Δt: as shown below, $\lambda = o(\alpha_\ell)$, and the condition on T gives $\lambda^2 T \leq \lambda \log(1/\alpha_\ell) \to 0$.

B.2 Proof

Proof sketch. Lemma 1 bounds the prediction and the change in upper sensitivity as the trained pair moves. Using these bounds, Proposition 1 controls the error in replacing the gradient updates by the fixed linear system of Theorem B.1. Lemma 2 solves this system and bounds how much it amplifies an error introduced at an earlier time. In the final proof, we integrate the remainder with this amplification factor and show that the weights remain small throughout $[0, T]$.

All constants C below are uniform in the initialization scales and time; they may depend on the fixed data, architecture, normalized seeds, derivative bounds and the constant in (26). Write $\rho = \|(\mathbf{w}_{\ell,j}, \boldsymbol{\omega}_{\ell+1,j})\|$. We establish the local estimates for $\rho \le r_*$, with a fixed small radius $r_* \le 1$, before showing that the trajectory stays in this ball.

Lemma 1 (Stability of the upper sensitivity)**.** *Under the assumptions of Theorem B.1, for sufficiently small scales and $\rho \le r_*$, uniformly over the training inputs,*

$$\|\mathbf{g}^{\ell+1} - \mathbf{g}_0\| \le C\lambda\rho^2, \qquad \|\mathbf{g}^{\ell+1}\| \le C\lambda, \qquad |f| \le C\lambda. \tag{29}$$

The upper preactivation changes by at most $C\rho^2$, and $\lambda = o(\alpha_\ell)$.

Proof. By $\sigma(0) = 0$ and the bounded inputs,

$$\|\mathbf{h}_{\ell+1}(x) - \mathbf{h}_{\ell+1,0}(x)\| \le C\|\boldsymbol{\omega}_{\ell+1,j}\|\,\|\mathbf{w}_{\ell,j}\| \le C\rho^2. \tag{30}$$

Thus the upper preactivations stay in a fixed bounded neighborhood when $\rho \le r_*$. As a function of $\mathbf{h}_{\ell+1}$, the output is given by the fixed upper layers. Forward propagation bounds its size, while the chain rule bounds its first two derivatives. Each bound contains the scale of every subsequent weight matrix and the readout. The activation derivatives are bounded, and any additional weight factors are bounded as the scales vanish. Hence

$$|f| + \left\|\frac{\partial f}{\partial \mathbf{h}_{\ell+1}}\right\| + \left\|\frac{\partial^2 f}{\partial \mathbf{h}_{\ell+1}^2}\right\|_2 \le C \prod_{k=\ell+2}^{L+1} \alpha_k \le C\lambda,$$

where the last inequality uses (26). The mean-value theorem and (30) give the sensitivity difference in (29). The same derivative bound at the fixed context, together with the definition of $\mathbf{A}_\ell$, gives

$$\lambda \le C \prod_{k=\ell+2}^{L+1} \alpha_k = o(\alpha_\ell),$$

by the initialization hierarchy. □

This estimate controls the errors from freezing the upper sensitivity and replacing the residual by the label in the two gradient updates.

Proposition 1 (Linear coupling of a row and its matching column)**.** *Under the assumptions of Lemma 1, for sufficiently small scales and $\rho \le r_*$,*

$$\left\|\left(\dot{\mathbf{w}}_{\ell,j} - \frac{\sigma'(0)}{\sqrt{p_\ell}}\mathbf{A}_\ell^\top \boldsymbol{\omega}_{\ell+1,j},\, \dot{\boldsymbol{\omega}}_{\ell+1,j} - \frac{\sigma'(0)}{\sqrt{p_\ell}}\mathbf{A}_\ell \mathbf{w}_{\ell,j}\right)\right\| \le C\lambda(\rho^2 + \alpha_\ell\rho). \tag{31}$$

Proof. The exact gradient updates in (4) are

$$\dot{\mathbf{w}}_{\ell,j} = \frac{1}{\sqrt{p_\ell}}\widehat{\mathbb{E}}_n[(y-f)\sigma'(h_{\ell,j})(\boldsymbol{\omega}_{\ell+1,j}^\top \mathbf{g}^{\ell+1})x], \qquad \dot{\boldsymbol{\omega}}_{\ell+1,j} = \frac{1}{\sqrt{p_\ell}}\widehat{\mathbb{E}}_n[(y-f)\mathbf{g}^{\ell+1}\sigma(h_{\ell,j})]. \tag{32}$$

Since $|h_{\ell,j}| \le C\rho$, Taylor's theorem gives

$$\sigma'(h_{\ell,j}) = \sigma'(0) + O(\rho), \qquad \sigma(h_{\ell,j}) = \sigma'(0)h_{\ell,j} + O(\rho^2).$$

Replacing the activation and its derivative by these leading terms, the sensitivity by $\mathbf{g}_0$, and the residual by y gives the linear system in Theorem B.1. By Lemma 1, the Taylor remainders contribute at most $C\lambda\rho^2$, freezing the sensitivity contributes at most $C\lambda\rho^3$, and replacing the residual contributes at most $C\lambda^2\rho$. Using $\rho \le 1$ and $\lambda = o(\alpha_\ell)$ gives (31). □

The next lemma gives the modes of this linear approximation and the growth bound used to integrate the remainder.

Lemma 2 (Amplification of the linear singular modes)**.** *Every solution of the linear system in Theorem B.1 satisfies, for $0 \le s \le t$,*

$$\|(\mathbf{w}_{\ell,j}^{\mathrm{lin}}(t), \boldsymbol{\omega}_{\ell+1,j}^{\mathrm{lin}}(t))\| \le e^{\lambda(t-s)}\|(\mathbf{w}_{\ell,j}^{\mathrm{lin}}(s), \boldsymbol{\omega}_{\ell+1,j}^{\mathrm{lin}}(s))\|.$$

The mode formulas (28) *hold for these linear trajectories with both errors equal to zero.*

Proof. The linear system is generated by the symmetric matrix

$$\frac{\sigma'(0)}{\sqrt{p_\ell}} \begin{pmatrix} 0 & \mathbf{A}_\ell^\top \\ \mathbf{A}_\ell & 0 \end{pmatrix},$$

whose largest eigenvalue is λ. Its eigendirections are multiplied by factors at most $e^{\lambda(t-s)}$ between times s and t, proving the norm bound. Projecting the row and column equations onto a singular pair and then taking their sum and difference gives

$$\langle \mathbf{w}_{\ell,j}^{\mathrm{lin}}(t), v_i \rangle \pm \langle \boldsymbol{\omega}_{\ell+1,j}^{\mathrm{lin}}(t), u_i \rangle = 2c_{ji}^{\pm} e^{\pm \sigma'(0) s_i t/\sqrt{p_\ell}}. \tag{33}$$

Adding and subtracting these identities gives (28). Both weights have the same growing amplitude, and their decaying components have opposite signs. For $s_i = 0$, both projections are constant. □

Proof of Theorem B.1. We combine the gradient approximation from Proposition 1 with the growth bound in Lemma 2 to control the error over $[0, T]$. The initial norm satisfies $\rho(0) \le K\alpha_\ell$ for a fixed $K > 0$. Let $E(t)$ be the norm of the difference between the trained pair and its linear approximation. Integrating the remainder bound with the linear growth factor gives, while $\rho \le r_*$,

$$E(t) \le C\lambda \int_0^t e^{\lambda(t-s)} [\rho(s)^2 + \alpha_\ell \rho(s)]\, ds. \tag{34}$$

Suppose $\rho(s) \le 2K\alpha_\ell e^{\lambda s}$ for $0 \le s \le t$. Evaluating the two integrals yields

$$E(t) \le C\alpha_\ell^2 e^{\lambda t}[(e^{\lambda t} - 1) + \lambda t] \le C\alpha_\ell^2 e^{\lambda t}(e^{\lambda t} - 1),$$

where the second inequality uses $\lambda t \le e^{\lambda t} - 1$. Recall that T is chosen so that $\alpha_\ell e^{\lambda T} \to 0$. For every such $t \le T$,

$$\frac{E(t)}{\alpha_\ell e^{\lambda t}} \le C\alpha_\ell (e^{\lambda T} - 1) \longrightarrow 0.$$

For sufficiently small scales, the linear trajectory bound therefore implies $\rho(t) \le K\alpha_\ell e^{\lambda t} + E(t) < \frac{3}{2} K\alpha_\ell e^{\lambda t}$. Also $2K\alpha_\ell e^{\lambda T} < r_*$. The bound assumed on ρ holds initially and cannot reach equality before T, by this strict improvement. Thus the flow stays in the bounded neighborhood and continues to T, proving (27). Finally, Lemma 2 gives the projections of the linear trajectory. Projection onto a unit singular vector cannot increase the error, so the same bound proves (28). □

C Proof of Theorem 2

We start by defining the continuous-time variant of Algorithm 2 studied in Theorem 2. It is described below for a three-layer network by Algorithm 3:

Algorithm 3 Three-layer BackLoFi: updates for the continuous-time limit

Require: Forward Neural LoFi features $\mathbf{h}_1^\star, \mathbf{h}_2^\star$ and readout w_3; fixed positive correction rates $\gamma^{\mathrm{row}}, \gamma^{\mathrm{col}}$; step size dt; duration T.

1: Set $C = 0$ and keep both Gaussian lifts fixed.
2: **for** each time step of size dt, up to time T **do**
3: **Select the coupled features.**
Compute $g^2 = w_3\sigma'(h_2)$ and the leading singular pair $(u, \mathbf{v})$ of $\widehat{\mathbb{E}}_n[y\, g^2 \otimes \mathbf{z}_0]$.
Set $\psi = \langle u, g^2 \rangle$, $\chi = \langle \mathbf{v}, \mathbf{z}_0 \rangle$; hold these scores fixed for this step.
4: **Correct the first layer.** $h_{1,a}^\star \leftarrow h_{1,a}^\star + dt\, \gamma^{\mathrm{row}} \chi$ for every a.
5: **Refresh forward LoFi.** Recompute z_1, K_1 and set $\mathbf{h}_2^\star \leftarrow \mathsf{FwdLoFi}_2(K_1, y)$.
6: **Add the column spike.** Using the updated z_1, set
$q \leftarrow \widehat{\mathbb{E}}_n[y\psi z_1], \qquad C \leftarrow C + dt\, \gamma^{\mathrm{col}} u \otimes q.$
7: **Train the readout.** Recompute h_2, z_2, h_3, then set
$w_3 \leftarrow w_3 + dt\, \widehat{\mathbb{E}}_n[(y - h_3) z_2].$
8: **end for**
9: **return** $h_3 = \langle w_3, z_2 \rangle$.

C.1 Full description of the Theorem and the underlying order parameters

We now give the full description of Theorem 2 and the order parameters that allow the equations to close. We study the continuous-time limit of Algorithm 3, followed by the width limits $p_1 \to \infty$ and $p_2 \to \infty$.

Filtered features and accumulated coupling. The input map $\mathbf{z}_0$ is fixed, with kernel $K_0(x, x') = \langle \mathbf{z}_0(x), \mathbf{z}_0(x') \rangle$. The filtered features $\mathbf{h}^\star_{1,t}(x) \in \mathbb{R}^{k_1}$ and $\mathbf{h}^\star_{2,t}(x) \in \mathbb{R}^{k_2}$ describe the two hidden layers before their Gaussian lifts. The first starts from $\mathbf{h}^\star_{1,0} = \mathsf{FwdLoFi}_1(K_0, y)$ and follows the lower correction; the second is refreshed as $\mathbf{h}^\star_{2,t} = \mathsf{FwdLoFi}_2(K_{1,t}, y)$.

Let $\mathbf{r} \sim \nu = \mathcal{N}(0, I_{k_1})$ and $\eta \sim \mu = \mathcal{N}(0, I_{k_2})$ be the fixed lift rows. The scalar function $C_t(\eta, \mathbf{r})$ records the total column coupling between these rows, starting from $C_0 = 0$. Its action is

$$(C_t f)(\eta) = \mathbb{E}_{\mathbf{r}}[C_t(\eta, \mathbf{r}) f(\mathbf{r})]. \tag{35}$$

The hidden preactivations and activations are therefore

$$\begin{aligned} h_{1,t}(\mathbf{r}, x) &= \mathbf{r}^\top \mathbf{h}^\star_{1,t}(x), & z_{1,t} &= \sigma(h_{1,t}), \\ h_{2,t}(\eta, x) &= \eta^\top \mathbf{h}^\star_{2,t}(x) + (C_t z_{1,t}(\cdot, x))(\eta), & z_{2,t} &= \sigma(h_{2,t}). \end{aligned} \tag{36}$$

Thus earlier columns act on the current lower features through $C_t z_{1,t}$. The arguments of C_t have dimensions k_2 and k_1, independent of network width. Neuron inner products are Gaussian expectations, or empirical averages at finite width; input coefficient inner products are Euclidean. This normalization absorbs the factors $1/\sqrt{p_j}$ used for the feature vectors in the main text.

Output and sensitivity. The scalar output is $h_{3,t} = \langle w_{3,t}, z_{2,t} \rangle$. To track how readout learning changes the sensitivity, retain the response $S_t(\eta, s) = w_{3,t}(\eta)\sigma'(s)$. Evaluating it at the current preactivation gives

$$g_t^2(\eta, x) = S_t(\eta, h_{2,t}(\eta, x)). \tag{37}$$

The output $h_{3,t}$ alone does not determine this sensitivity when the hidden features change; S_t retains the readout's response to that change. Together, $\mathbf{h}^\star_{1,t}, \mathbf{h}^\star_{2,t}, C_t, S_t, h_{3,t}$ determine the feature and sensitivity kernels:

$$\begin{aligned} K_{1,t}(x, x') &= \mathbb{E}_{\mathbf{r}}[z_{1,t}(\mathbf{r}, x) z_{1,t}(\mathbf{r}, x')], \\ K_{2,t}(x, x') &= \mathbb{E}_{\eta}[z_{2,t}(\eta, x) z_{2,t}(\eta, x')], \qquad G_t(x, x') = \mathbb{E}_{\eta}[g_t^2(\eta, x) g_t^2(\eta, x')]. \end{aligned} \tag{38}$$

Selection and continuous updates. At each time, (ψ_t, χ_t) maximizes (15) for the current kernels G_t, K_0. The singular-vector equations become

$$\rho_t \psi_t(x) = \widehat{\mathbb{E}}_n[G_t(x, X) y \chi_t(X)], \qquad \rho_t \chi_t(x) = \widehat{\mathbb{E}}_n[K_0(x, X) y \psi_t(X)], \tag{39}$$

where both features have unit RKHS norm and $\rho_t > 0$ is the achieved correlation. With fixed correction rates $\gamma^{\mathrm{row}}, \gamma^{\mathrm{col}} > 0$, Algorithm 3 gives

$$\dot{h}^\star_{1,t,a}(x) = \gamma^{\mathrm{row}} \chi_t(x), \qquad a = 1, \dots, k_1, \tag{40}$$

$$\partial_t C_t(\eta, \mathbf{r}) = \frac{\gamma^{\mathrm{col}}}{\rho_t} \widehat{\mathbb{E}}_n[y \chi_t(X) g_t^2(\eta, X)] \widehat{\mathbb{E}}_n[y \psi_t(X) z_{1,t}(\mathbf{r}, X)], \tag{41}$$

$$\partial_t S_t(\eta, s) = \sigma'(s) \widehat{\mathbb{E}}_n[(y - h_{3,t}(X)) z_{2,t}(\eta, X)]. \tag{42}$$

The first equation corrects the lower filter. The second accumulates its paired column spikes, using $u_t = \rho_t^{-1} \widehat{\mathbb{E}}_n[y \chi_t g_t^2]$ and $q_t = \widehat{\mathbb{E}}_n[y \psi_t z_{1,t}]$. The third is the final-layer gradient-flow update, with s held fixed. The output evolves by (19).

Forward filter. The forward map $\mathsf{FwdLoFi}_\ell(K, y)$ is Algorithm 1 expressed in feature space. Let $\mathcal{H}_K$ be the RKHS of the current kernel K. Following Dandi et al. (2026b), its features are selected by

$$\begin{aligned} f_0 &\in \operatorname*{arg\,max}_{\|f\|_{\mathcal{H}_K} = 1} \widehat{\mathbb{E}}_n[y f], \\ f_a &\in \operatorname*{arg\,max}_{\substack{\|f\|_{\mathcal{H}_K} = 1 \\ \langle f, f_b \rangle_{\mathcal{H}_K} = 0,\ 1 \le b < a}} \left| \widehat{\mathbb{E}}_n[y f^2] \right|, \qquad a = 1, \dots, k_\ell - 1. \end{aligned} \tag{43}$$

The first coordinate maximizes first-order correlation; the remaining coordinates successively maximize second-order correlation. Orthogonality is measured in $\mathcal{H}_K$ among the second-order features. The map returns $\mathsf{FwdLoFi}_\ell(K, y) = (f_0, f_1, \ldots, f_{k_\ell - 1})$. At initialization, choose the sign of each second-order forward feature and of χ to have positive empirical inner product with a fixed reference function. At later steps, use its previous value as the reference, and change the sign of ψ together with that of χ.

Assumption 4 (Data, lifts and initialization)**.** The training set is finite, and the numbers of filtered coordinates k_1, k_2 are fixed. The lift laws are $\nu = \mathcal{N}(0, I_{k_1})$ and $\mu = \mathcal{N}(0, I_{k_2})$, or their finite empirical versions. All lift rows are held fixed in time. The positive correction rates $\gamma^{\mathrm{row}}, \gamma^{\mathrm{col}}$ are fixed. Initially $C_0 = 0$. Choose $h_{3,0}$ in the span of the initial features $z_{2,0}$ on the training inputs, and set S_0 by (46).

Assumption 5 (Activation)**.** The activation is C^2, and $\sigma, \sigma', \sigma''$ are bounded.

Assumption 6 (Spectral gaps)**.** On $[0, T]$, the label-weighted first moments used by the forward filters have norms bounded away from zero. The retained eigenvalues are uniformly separated in absolute value from zero, from one another, and from the discarded spectrum. The leading coupled singular value stays uniformly separated from zero and from the remaining singular values.

Assumption 7 (Sequential width limit)**.** The two lifts are sampled from independent nested Gaussian sequences. The data, ranks, initial prediction vector and correction rates are the same across widths. The same reference functions initialize the signs at every width, with nonzero inner product with the limiting features. The initial upper Gaussian-population kernel matrix is positive definite. The population trajectory satisfies Assumption 6 on $[0, T]$.

Theorem C.1 (Precise version of Theorem 2)**.** *Under Assumptions 4–6, the current-state equations have a unique local solution. On $[0, T]$, the piecewise-linear interpolants of Algorithm 3 converge to it uniformly as $dt \to 0$, with error $O_T(dt)$. Under Assumption 7, the continuous flows also converge almost surely, first as $p_1 \to \infty$ at fixed admissible p_2, then as $p_2 \to \infty$, to the solution of* (17)–(19)*, with expectations taken over both Gaussian lifts. Convergence is uniform on $[0, T]$ for C_t, S_t in the fixed norms specified in* (50)*, and on the data and $[0, T]$ for the filtered features, output, selected scores and current kernels $K_{1,t}, K_{2,t}, G_t$.*

Here an admissible fixed upper sample means that the mixed Gaussian-lower/empirical-upper flow realizes the prescribed initial predictions, has a stable initial kernel inverse, and satisfies Assumption 6 on $[0, T]$. The width statement concerns sufficiently large upper widths, for which these conditions hold.

C.2 Proof of Theorem C.1

Proof sketch. To obtain a closed evolution and compare networks of different widths, we represent the accumulated columns by a kernel $C_t(\eta, \mathbf{r})$ on the fixed Gaussian label spaces. Each selected feature pair contributes a rank-one increment, and $C_t z_{1,t}$ records the accumulated action on the current lower features. Differentiating $C_t z_{1,t}$ therefore retains both newly added columns and the action of earlier columns on moving features. Together with the readout update, this gives the output equation and the evolving sensitivity. The spectral gaps ensure that the selected features vary continuously with the state, yielding a locally Lipschitz flow. Each iteration of Algorithm 3 approximates (40)–(42) over an interval of length dt with error $O(dt^2)$, giving an $O_T(dt)$ error on $[0, T]$. Finally, we compare empirical and Gaussian lifts on the same probability spaces. Stability of the flow lets us average the first lift and then the second, while retaining the dependence of learned features on the sampled neurons.

We start by establishing the continuity of the maximizing feature pair with respect to the evolving order parameters.

Lemma 3 (Continuity of the maximizing feature pair)**.** *Fix the training data, input feature map $\mathbf{z}_0$ and lift laws μ, ν. Under Assumptions 5 and 6, the selected singular pair $(u, \mathbf{v})$ and scores (ψ, χ) depend locally Lipschitz-continuously on the current order parameters $(\mathbf{h}_1^\star, C, S)$, with $\mathbf{h}_2^\star = \mathsf{FwdLoFi}_2(K_1, y)$ and signs chosen as above. Use the empirical norm $\|f\|_n^2 = \widehat{\mathbb{E}}_n[\|f(X)\|^2]$ for the filtered features and scores, the Hilbert–Schmidt norm for C, and the $L^2(\mu; \mathcal{D})$ norm for S, where $\mathcal{D} = \mathrm{span}\{\sigma'\}$ and $\|b\sigma'\|_{\mathcal{D}} = |b|$. The statement holds on bounded sets with $\|S(\eta, \cdot)\|_{\mathcal{D}} \le M$ almost everywhere, for fixed $M < \infty$.*

Proof. The first-layer features $z_1(\mathbf{r}, x) = \sigma(\mathbf{r}^\top \mathbf{h}_1^\star(x))$ depend Lipschitz-continuously on $\mathbf{h}_1^\star$: the derivative of σ is bounded and $\mathbf{r}$ has finite second moment. Their label-weighted first and second moments inherit this local regularity. Assumption 6 keeps the first moment nonzero and separates the retained eigenvalues of the second-moment operator. Normalizing the first moment and applying eigenvector perturbation bounds therefore give local Lipschitz continuity of the forward features $\mathbf{h}_2^\star = \mathsf{FwdLoFi}_2(K_1, y)$.

For two sets of order parameters $(\mathbf{h}_1^\star, C, S)$ and $(\bar{\mathbf{h}}_1^\star, \bar{C}, \bar{S})$, let $h_2, \bar{h}_2$ be the corresponding upper preactivations. Equation (36) and the bound $\|Cf\|_2 \le \|C\|_{\mathrm{HS}}\|f\|_2$ give

$$\|h_2 - \bar{h}_2\|_2 \le L\big(\|\mathbf{h}_1^\star - \bar{\mathbf{h}}_1^\star\|_n + \|C - \bar{C}\|_{\mathrm{HS}}\big).$$

The L^2 norms on preactivations and sensitivities average over the upper label and training inputs. Since $g^2(\eta, x) = S(\eta, h_2(\eta, x))$, the corresponding sensitivities satisfy

$$\|g^2 - \bar{g}^2\|_2 \le \|\sigma'\|_\infty \|S - \bar{S}\|_{L^2(\mu;\mathcal{D})} + M\|\sigma''\|_\infty \|h_2 - \bar{h}_2\|_2.$$

These estimates give local Lipschitz continuity of g^2 in $(\mathbf{h}_1^\star, C, S)$. Since $A = \widehat{\mathbb{E}}_n[y\, g^2 \otimes \mathbf{z}_0]$ is bounded and linear in g^2, A^*A also depends locally Lipschitz-continuously on these order parameters. Its leading eigenvalue ρ^2 is positive and simple. Fixing the sign of its unit eigenvector $\mathbf{v}$ by positive overlap with a reference, the equations

$$A^*A\mathbf{v} = \rho^2\mathbf{v}, \qquad \|\mathbf{v}\|_2 = 1$$

have an invertible linearization. On $\mathbf{v}^\perp$, the inverse of $\rho^2 I - A^*A$ has norm at most $1/(\rho^2 - \rho_2^2)$, where ρ_2 is the second singular value of A (zero when A has rank one). Assumption 6 keeps this eigenvalue difference bounded away from zero. The implicit function theorem gives smooth dependence of ρ and $\mathbf{v}$ on A, and hence local Lipschitz dependence on the order parameters. The identities $u = A\mathbf{v}/\rho$, $\psi = \langle u, g^2\rangle$ and $\chi = \langle \mathbf{v}, \mathbf{z}_0\rangle$ give the same regularity for the left singular vector and both maximizing functions. □

We next verify that the column updates accumulate in C_t as defined in (35), and derive the evolution of their action $C_t z_{1,t}$ on the changing lower features.

Proposition 2 (Accumulated column coupling)**.** *Let μ, ν be the Gaussian lift laws or their empirical versions. Suppose $u_t \in L^2(\mu)$ and $q_t \in L^2(\nu)$ are continuous in t, and $\gamma^{\mathrm{col}} \ge 0$ is fixed. The column update $\dot{C}_t = \gamma^{\mathrm{col}} u_t \otimes q_t$, $C_0 = 0$, has a unique continuously differentiable Hilbert–Schmidt solution, whose scalar kernel is given in $L^2(\mu \otimes \nu)$ by*

$$C_t(\eta, \mathbf{r}) = \int_0^t \gamma^{\mathrm{col}} u_\tau(\eta) q_\tau(\mathbf{r})\, d\tau. \tag{44}$$

It maps functions of the lower label $\mathbf{r}$ to functions of the upper label η through (35)*. For any continuously differentiable lower-feature curve $z_{1,t}$,*

$$\partial_t(C_t z_{1,t}) = \dot{C}_t z_{1,t} + C_t \dot{z}_{1,t}.$$

Proof. A rank-one update has kernel $u_t(\eta)q_t(\mathbf{r})$ and Hilbert–Schmidt norm $\|u_t\|_{L^2(\mu)}\|q_t\|_{L^2(\nu)}$. Integrating in this Hilbert space gives the unique solution and the bound

$$\|C_t\|_{\mathrm{HS}} \le \int_0^t \gamma^{\mathrm{col}} \|u_\tau\|_{L^2(\mu)} \|q_\tau\|_{L^2(\nu)}\, d\tau.$$

The kernel therefore belongs to $L^2(\mu \otimes \nu)$. Since applying a Hilbert–Schmidt operator to an $L^2(\nu)$ function is continuous and bilinear, the product rule gives the displayed derivative. □

Combining this representation with the readout update gives the closed response and output equations.

Proposition 3 (Closed evolution of the response and output)**.** *Under Assumptions 4–6, let $(\mathbf{h}_{1,t}^\star, C_t, w_{3,t})$ solve the lower-feature and column equations* (40)–(41) *and final-layer gradient flow under squared loss, with the prescribed initialization. Then $S_t(\eta, s) = w_{3,t}(\eta)\sigma'(s)$ satisfies* (42)*, and $h_{3,t} = \langle w_{3,t}, z_{2,t}\rangle$ satisfies* (19)*. These derivatives, and those of the feature maps and their kernels, are determined by the current values of $(\mathbf{h}_{1,t}^\star, \mathbf{h}_{2,t}^\star, C_t, S_t, h_{3,t})$.*

Proof. The update rates are continuous in the order parameters by Lemma 3. Thus any solution of their integral equations is continuously differentiable in time. We derive the response and output identities for such a solution; its existence and uniqueness are established in the proof of Theorem C.1 below.

Write the readout as $w_{3,t} \in L^2(\mu)$. The cross-layer operator is $A_t = \widehat{\mathbb{E}}_n[y\, g_t^2 \otimes \mathbf{z}_0]$. For a nonzero singular pair, $A_t v_t = \rho_t u_t$ gives $u_t = \rho_t^{-1}\widehat{\mathbb{E}}_n[y\chi_t g_t^2]$; applying the upper feature map gives $\rho_t\psi_t(x) = \widehat{\mathbb{E}}_n[G_t(x,X)y\chi_t(X)]$. The adjoint equation similarly gives $\rho_t\chi_t(x) = \widehat{\mathbb{E}}_n[K_0(x,X)y\psi_t(X)]$. Each singular vector lies in the span of its feature map and therefore has the minimum coefficient norm among representations of the selected feature, which equals the RKHS norm of that feature. In particular, the unit vectors u_t, v_t give $\|\psi_t\|_{\mathcal{H}_{G_t}} = \|\chi_t\|_{\mathcal{H}_{K_0}} = 1$.

Every lower filtered coordinate follows (40). The remaining continuous updates are

$$\begin{aligned} \dot{C}_t &= \gamma^{\mathrm{col}} u_t \otimes q_t, \qquad \dot{w}_{3,t} = \widehat{\mathbb{E}}_n[(y - h_{3,t}(X))z_{2,t}(\cdot, X)], \\ q_t &= \widehat{\mathbb{E}}_n[y\psi_t(X)z_{1,t}(\cdot,X)], \qquad u_t = \rho_t^{-1}\widehat{\mathbb{E}}_n[y\chi_t(X)g_t^2(\cdot,X)]. \end{aligned} \tag{45}$$

Here $(u \otimes q)b = u\langle q, b\rangle$, and $h_{2,t}(\eta, x) = \eta^\top \mathbf{h}^\star_{2,t}(x) + (C_t z_{1,t}(x))(\eta)$. Substituting the two representers into $\dot{C}_t$ gives (41). The loading q_t fits the modified label on current lower activations, and u_t selects the upper response.

To initialize the response from $h_{3,0}$, choose coefficients b_0 on the training inputs satisfying

$$\widehat{\mathbb{E}}_n[K_{2,0}(x,X)b_0(X)] = h_{3,0}(x) \qquad \text{for every training input } x.$$

A solution exists by Assumption 4. The readout $w_{3,0}(\eta) = \widehat{\mathbb{E}}_n[b_0(X)z_{2,0}(\eta,X)]$ realizes $h_{3,0}$ and lies in the span of the initial activation profiles. Any other readout with the same predictions differs by a function orthogonal to that span. Hence $w_{3,0}$ has minimum norm and is independent of the chosen coefficient solution.

The corresponding initial response is

$$S_0(\eta, s) = \sigma'(s)\widehat{\mathbb{E}}_n[b_0(X)z_{2,0}(\eta,X)]. \tag{46}$$

Holding s fixed and applying the readout gradient update gives (42); evaluating at $h_{2,t}$ gives the current sensitivity in (37). Its total derivative therefore contains both readout learning and the changing activation derivative:

$$\begin{aligned} \partial_t g_t^2(\eta, x) = \sigma'(h_{2,t}(\eta,x))\widehat{\mathbb{E}}_n[(y - h_{3,t}(X))z_{2,t}(\eta,X)] \\ + \partial_s S_t(\eta, h_{2,t}(\eta,x))\, \dot{h}_{2,t}(\eta,x). \end{aligned} \tag{47}$$

The first term changes the response even with the hidden features held fixed. The second changes it as the upper preactivation moves, with $\partial_s S_t(\eta, s) = w_{3,t}(\eta)\sigma''(s)$. Both update G_t, hence the next selected scores (ψ_t, χ_t), which in turn update the lower filter and the column coupling. Since $\partial_t S_t$ is $\sigma'(s)$ times a function of η, the evolution preserves the product form $S_t(\eta, s) = w_{3,t}(\eta)\sigma'(s)$ of the initialization.

Proposition 2 represents all installed columns by the current kernel C_t. The accumulated columns act on the current lower features through $C_t z_{1,t}$. Differentiating this product gives

$$\begin{aligned} \dot{h}_{2,t}(\eta,x) &= \eta^\top \dot{\mathbf{h}}^\star_{2,t}(x) + (\dot{C}_t z_{1,t}(x))(\eta) + (C_t \dot{z}_{1,t}(x))(\eta), \\ \dot{z}_{1,t}(\mathbf{r},x) &= \sigma'(\mathbf{r}^\top \mathbf{h}^\star_{1,t}(x))\gamma^{\mathrm{row}}\chi_t(x)\sum_{a=1}^{k_1} r_a. \end{aligned} \tag{48}$$

The three terms in the first line refresh the forward filter, install the current column, and propagate lower-feature motion through previously installed columns. The second line follows because every filtered coordinate has the same velocity: $\mathbf{r}^\top \dot{\mathbf{h}}^\star_{1,t} = \gamma^{\mathrm{row}}\chi_t \sum_a r_a$. The change in the first-layer kernel then refreshes the second-layer forward filter:

$$\begin{aligned} \dot{K}_{1,t}(x,x') &= \mathbb{E}_{\mathbf{r}}\big[\dot{z}_{1,t}(\mathbf{r},x)z_{1,t}(\mathbf{r},x') + z_{1,t}(\mathbf{r},x)\dot{z}_{1,t}(\mathbf{r},x')\big], \\ \dot{\mathbf{h}}^\star_{2,t} &= D_K \mathsf{FwdLoFi}_2(K_{1,t}, y)[\dot{K}_{1,t}], \end{aligned}$$

where $D_K\mathsf{FwdLoFi}_2$ is the derivative of the forward filter with respect to its kernel. Thus all three derivatives are determined by the current filtered features and their prescribed velocity.

Differentiate $h_{3,t} = \langle w_{3,t}, z_{2,t}\rangle$ to obtain

$$\dot{h}_{3,t}(x) = \widehat{\mathbb{E}}_n[K_{2,t}(x,X)(y - h_{3,t}(X))] + \mathbb{E}_\eta[g_t^2(\eta,x)\dot{h}_{2,t}(\eta,x)],$$
$$\mathbb{E}_\eta[g_t^2(\eta,x)(\dot{C}_t z_{1,t}(x))(\eta)] = \gamma^{\mathrm{col}}\psi_t(x)\widehat{\mathbb{E}}_n[K_{1,t}(x,X)y\psi_t(X)]. \tag{49}$$

The second identity uses $\langle g_t^2, u_t\rangle = \psi_t$ and $\langle q_t, z_{1,t}(x)\rangle = \widehat{\mathbb{E}}_n[K_{1,t}(x,X)y\psi_t(X)]$. Substituting (48) gives the four terms of (19), including $\mathbb{E}_\eta[g_t^2(C_t\dot{z}_{1,t})]$ for earlier columns. These derivatives also hold for Gaussian lifts. The readout equation keeps $w_{3,t}$ uniformly bounded over neurons on finite intervals; the formulas for u_t, q_t and C_t then give $|\dot{h}_{1,t}(\mathbf{r},x)| \le C_T\|\mathbf{r}\|$ and $|\dot{h}_{2,t}(\eta,x)| \le C_T(1+\|\eta\|)$. Bounded activation derivatives and finite Gaussian second moments therefore justify differentiating the neuron expectations by dominated convergence. Conversely, the output equation and the response update give identical derivatives for $h_{3,t}$ and $\langle w_{3,t}, z_{2,t}\rangle$. Their equal initialization therefore preserves equality. Since $g_t^2 = S_t(\eta, h_{2,t})$ and $\mathbf{h}_{2,t}^\star = \mathsf{FwdLoFi}_2(K_{1,t}, y)$, the variational pair and the representers in (45) are determined by the current state, as are the feature and kernel derivatives. □

Proof of Theorem C.1. We now combine Lemma 3 and Propositions 2 and 3 to prove Theorem C.1. Proposition 2 gives the evolution of the accumulated columns and their action on the changing lower features; Proposition 3 gives the closed response and output equations. Lemma 3 supplies the local Lipschitz bounds used to establish existence and uniqueness, control the time-discretization error, and pass to the two successive width limits.

The filtered features give the Gaussian projections $\mathbf{r}^\top\mathbf{h}_1^\star(x)$ and $\eta^\top\mathbf{h}_2^\star(x)$. The coupling C maps the lower activation $z_1(\cdot,x) \in L^2(\nu)$ to the upper-preactivation correction $Cz_1(\cdot,x) \in L^2(\mu)$, while $S(\eta,\cdot)$ gives the sensitivity evaluated at the corrected upper preactivation. Their convergence is therefore naturally formulated on the Hilbert spaces of functions of these Gaussian variables: we measure C in Hilbert–Schmidt norm and S in $L^2(\mu;\mathcal{D})$, with the response norm defined in Lemma 3. Together with the filtered features on the training inputs, this gives the state space

$$\mathbb{R}^{nk_1} \times \mathrm{HS}(L^2(\nu), L^2(\mu)) \times L^2(\mu;\mathcal{D}) \tag{50}$$

for $(\mathbf{h}_1^\star, C, S)$. Here HS denotes Hilbert–Schmidt operators: the squared norm of C is the integral of its squared kernel over both neuron labels. Since $S(\eta,s) = w_3(\eta)\sigma'(s)$, the chosen response norm satisfies

$$\|S\|_{L^2(\mu;\mathcal{D})}^2 = \mathbb{E}_\eta[|w_3(\eta)|^2].$$

The same identity holds for differences of two responses. We can therefore prove existence, uniqueness and convergence for the coefficient function w_3, then recover S by multiplying by the fixed function σ'. Since $\|\chi_t\|_{\mathcal{H}_{K_0}} = 1$, the reproducing property gives $|\chi_t(x)| \le \sqrt{K_0(x,x)}$. Hence

$$\|\dot{\mathbf{h}}_{1,t}^\star\|_n = \sqrt{k_1}\,\gamma^{\mathrm{row}}\|\chi_t\|_n \le \gamma^{\mathrm{row}}\sqrt{k_1\widehat{\mathbb{E}}_n[K_0(X,X)]}.$$

The fixed input kernel and filter rank make this bound uniform in width.

The canonical initial readout is a finite linear combination of bounded activations, and its update satisfies

$$\|\dot{w}_{3,t}\|_\infty \le \|\sigma\|_\infty(\|y\|_n + \|\sigma\|_\infty\|w_{3,t}\|_2).$$

Lemma 3 gives local Lipschitz continuity of the right-hand sides of (40)–(42) in $(\mathbf{h}_1^\star, C, S)$, while the response coefficients are bounded. Integrating these right-hand sides from the prescribed initial state gives a contraction on bounded trajectories near the initial state over a sufficiently short interval: the displayed readout bound keeps the coefficients bounded, and the Lipschitz constant is multiplied by the interval length. This proves local existence and uniqueness. Grönwall bounds $\|w_{3,t}\|_2$ and hence $\|w_{3,t}\|_\infty$ on finite intervals, so the solution extends while Assumption 6 holds. Proposition 3 then identifies its output with (19).

We now show that Algorithm 3 converges to these equations as $dt \to 0$, with the lifts fixed. Denote the changes over one iteration by $\Delta\mathbf{h}_1^\star, \Delta C, \Delta S$. The lower-feature update is exactly dt times (40). The column and readout updates use the features obtained after the earlier updates in that iteration.

Those features differ from their values at the start of the iteration by $O(dt)$. The Lipschitz bounds therefore give

$$\begin{aligned}\Delta \mathbf{h}_1^\star &= dt\, \dot{\mathbf{h}}_{1,t}^\star,\\ \Delta C &= dt\, \partial_t C_t + O_T(dt^2),\\ \Delta S &= dt\, \partial_t S_t + O_T(dt^2),\end{aligned}$$

where the derivatives are the right-hand sides of (40)–(42), evaluated at the start of the iteration. The errors are measured in (50). Integrating the continuous equations over the same interval gives these first-order increments with $O_T(dt^2)$ remainders.

Let E_k be the error between the algorithm and the continuous solution at time $k\,dt$, measured in (50). While the iterates remain in a neighborhood of the continuous trajectory on which Assumption 6 holds uniformly, the error satisfies

$$E_{k+1} \le (1 + L_T dt)E_k + C_T dt^2, \qquad E_0 = 0.$$

Summing this recurrence gives $\max_{k\,dt\le T} E_k = O_T(dt)$. For sufficiently small dt, this bound keeps every iterate in that neighborhood up to time T. Linear interpolation gives the same uniform error between steps. The readout bound above holds for the discrete updates as well. Continuity of the forward filter and Lemma 3 transfer this convergence to the features, output, kernels and selected scores.

We next take the width limits. The learned features depend on the sampled lifts, so we compare the finite-width and Gaussian equations along their evolving trajectories. For the lower lift, write the sampled rows as $\mathbf{r}_1, \dots, \mathbf{r}_{p_1}$. Partition the Gaussian label space into measurable cells $V_1, \dots, V_{p_1}$, each of probability $1/p_1$, chosen to minimize

$$\sum_{i=1}^{p_1} \int_{V_i} \|\mathbf{r} - \mathbf{r}_i\|^2 \, d\nu(\mathbf{r}).$$

Set $T_{1,p_1}(\mathbf{r}) = \mathbf{r}_i$ for $\mathbf{r} \in V_i$. This is the quadratic optimal transport map from the Gaussian law to the empirical distribution of the sampled rows. The Gaussian density and finite second moments ensure that such a minimizing partition exists (Villani, 2003, Theorem 2.12). Construct T_{2,p_2} in the same way for the upper lift.

To see why the transport cost tends to zero, divide a bounded box B into finitely many cells of diameter at most ε. In each cell, the fraction of sampled rows converges almost surely to its Gaussian probability by the strong law. Assigning as much Gaussian mass as possible to sampled rows within each cell leaves a vanishing unassigned mass inside B, with squared distance at most ε^2 on the assigned mass. For the remaining assignments, use $\|\mathbf{r} - \mathbf{r}_i\|^2 \le 2\|\mathbf{r}\|^2 + 2\|\mathbf{r}_i\|^2$. Inside B, squared norms are bounded and the unassigned mass vanishes. Outside B, the sampled second moment converges to $\int_{B^c} \|\mathbf{r}\|^2 \, d\nu$ by the strong law. Since the chosen partition minimizes the cost,

$$\limsup_{p_1 \to \infty} \|T_{1,p_1}(\mathbf{r}) - \mathbf{r}\|^2_{L^2(\nu)} \le \varepsilon^2 + 4 \int_{B^c} \|\mathbf{r}\|^2 \, d\nu(\mathbf{r}) \quad \text{almost surely.}$$

Letting the cell diameter tend to zero and the box grow proves convergence. The same argument applies to the upper lift, giving

$$\delta_{1,p_1} := \|T_{1,p_1}(\mathbf{r}) - \mathbf{r}\|_{L^2(\nu)} \to 0, \qquad \delta_{2,p_2} := \|T_{2,p_2}(\eta) - \eta\|_{L^2(\mu)} \to 0$$

almost surely.

Extend each sampled neuron profile to be constant on its cell. Because every cell has probability $1/p_j$, Gaussian integrals of these extensions equal the original empirical averages. Extend C_t on products of lower and upper cells, and S_t on upper cells. These extensions preserve the norms and all update equations. The finite-width and limiting solutions can therefore be compared in the same Hilbert space (50).

At fixed order parameters, replacing $\mathbf{r}$ by $T_{1,p_1}(\mathbf{r})$ changes z_1 by at most $C_T\delta_{1,p_1}$ in L^2. The first-layer kernel and refreshed forward filter satisfy the same bound. Replacing the upper lift changes the upper preactivation by at most $C_T(\delta_{1,p_1} + \delta_{2,p_2})$. Lemma 3 gives this bound for the resulting change

in the selected pair and hence in the update equations. The constants are uniform in width when the response coefficients and lift moments are bounded and Assumption 6 holds uniformly. The initial responses also converge: their coefficients solve the kernel linear system above (46), whose limiting matrix is invertible by assumption.

Let $E(t)$ denote the distance between the two solutions after this identification, in the norm of (50). Integrating the difference of their update equations and applying Grönwall gives

$$\sup_{t\le T} E(t) \le C_T\big(E(0) + \delta_{1,p_1} + \delta_{2,p_2}\big). \tag{51}$$

Choose a neighborhood of the limiting trajectory on which Assumption 6 holds uniformly. The estimate applies until the finite-width solution leaves this neighborhood. Its right-hand side tends to zero, so for sufficiently large widths the solution stays in the neighborhood throughout $[0, T]$.

To obtain the sequential limit, fix p_2 and an admissible set of upper lift rows. The two flows compared in (51) use these same rows, so the upper-lift error is zero. As $p_1 \to \infty$, the remaining error tends to zero almost surely. The limit therefore uses Gaussian expectations in the first layer and empirical averages over the fixed rows in the second. We then let $p_2 \to \infty$ and apply the same estimate to this intermediate evolution, obtaining the equations with Gaussian expectations in both layers. The positive-definite initial limiting kernel and stability under the spectral-gap assumption ensure that all sufficiently large upper samples are admissible almost surely.

Continuity of the forward filter and the maximizing pair gives convergence of the filtered features and selected scores. Cauchy–Schwarz then gives convergence of the output and kernels from their expressions as inner products. Together with (51), these bounds establish almost-sure convergence uniformly on $[0, T]$ in the stated norms. This completes the proof of Theorem C.1. □

D THE EXACT LINEAR REDUCTION

We first derive (13) for the one-hidden-layer linear network $f(x) = \mathbf{W}_2\mathbf{W}_1 x$. Take unnormalized linear activations, unit-rate squared-loss gradient flow, whitened inputs $\mathbb{E}[xx^\top] = I$, and possibly vector-valued targets. Write $S = \mathbb{E}[yx^\top]$. Direct differentiation gives

$$A_\Delta = S - \mathbf{W}_2\mathbf{W}_1, \qquad \dot{\mathbf{W}}_1 = \mathbf{W}_2^\top A_\Delta, \qquad \dot{\mathbf{W}}_2 = A_\Delta \mathbf{W}_1^\top. \tag{52}$$

Thus the incoming row and its matching outgoing column obey the two actions in (6) exactly, with the residual input–output correlation A_Δ as coupling matrix. The prediction term produces saturation as the network fits the target.

Let $S = \sum_i s_i u_i v_i^\top$ and choose orthonormal hidden directions q_i. The aligned initialization

$$\mathbf{W}_1 = \sum_i \alpha_i q_i v_i^\top, \qquad \mathbf{W}_2 = \sum_i \beta_i u_i q_i^\top \tag{53}$$

is preserved: substituting it into (52) and using orthogonality gives

$$\dot{\alpha}_i = \beta_i(s_i - \alpha_i\beta_i), \qquad \dot{\beta}_i = \alpha_i(s_i - \alpha_i\beta_i), \qquad \frac{d}{dt}(\alpha_i^2 - \beta_i^2) = 0. \tag{54}$$

For a balanced positive seed, $m_i = \alpha_i\beta_i$ satisfies $\dot{m}_i = 2m_i(s_i - m_i)$, hence

$$m_i(t) = \frac{s_i}{1 + (s_i/m_i(0) - 1)e^{-2s_i t}}.$$

When $m_i(0) \ll s_i$, each factor initially grows as $e^{s_i t}$, and their product approaches s_i. Alignment is needed for independent scalar modes; an arbitrary initialization also has cross-mode interactions.

Without alignment, linearizing (52) at zero still gives, for each row–column pair,

$$\frac{d}{dt}\begin{pmatrix}\mathbf{w}_{1,j}\\ \boldsymbol{\omega}_{2,j}\end{pmatrix} = \begin{pmatrix}0 & S^\top\\ S & 0\end{pmatrix}\begin{pmatrix}\mathbf{w}_{1,j}\\ \boldsymbol{\omega}_{2,j}\end{pmatrix}.$$

The discarded terms are cubic. The block matrix has eigenvectors $(v_i, \pm u_i)/\sqrt{2}$ with eigenvalues $\pm s_i$, which proves the shared exponential amplification near zero for arbitrary initial directions. This connects the linear dynamics to Theorem 1; the nonlinear surrogate additionally freezes gates and replaces the residual by a label-based approximation.

More than two weight matrices. For $f(x) = \mathbf{W}_D \cdots \mathbf{W}_1 x$, aligned singular directions reduce each mode to layer amplitudes $a_{1i}, \ldots, a_{Di}$. Writing $m_i = \prod_{k=1}^{D} a_{ki}$, its loss is $\frac{1}{2}(s_i - m_i)^2$. Differentiating gives the mode equations of Saxe et al. (2014, Section 2):

$$\dot{a}_{ki} = (s_i - m_i) \prod_{r \neq k} a_{ri}, \qquad \dot{m}_i = (s_i - m_i) \sum_{k=1}^{D} \left(\prod_{r \neq k} a_{ri} \right)^2 . \tag{55}$$

Equal positive initialization across layers is preserved by these equations. Then every $a_{ki} = m_i^{1/D}$,which gives (14). For $D = 2$, this is the balanced two-factor equation $\dot{m}_i = 2m_i(s_i - m_i)$. For $D > 2$, the early growth law is $\dot{m}_i \simeq D s_i m_i^{2-2/D}$. The additional powers arise from multiplying the contributions of all layers. The constant-rate exponential approximation for two weight matrices therefore does not describe the simultaneous growth of all layers in a deeper network.

E THE DEEP STAIRCASE: A THEORETICAL SETTING

We analyze here the deep-staircase model of Section 4. We first define the complete teacher, then specialize the forward and backward spectral operations of Algorithms 1 and 2 to this setting. We next prove the population separation between the forward and label-corrected statistics. We derive a sample guarantee for recovery of the complete first-level latent subspace. The last subsection states a sufficient condition for replacing the oracle sensitivity used in the theorem by a learned upper sensitivity.

E.1 TEACHER AND HIGH-DIMENSIONAL REGIME

Target. Let $\mathbf{x} \sim \mathcal{N}(0, I_d)$ and define three mutually orthogonal first-level blocks

$$\mathbf{h}_U = A_U \mathbf{x} \in \mathbb{R}^{p_U}, \qquad \mathbf{h}_V = A_V \mathbf{x} \in \mathbb{R}^{p_V}, \qquad \mathbf{h}_W = A_W \mathbf{x} \in \mathbb{R}^{p_W}, \tag{56}$$

where the rows of A_U, A_V, A_W are collectively orthonormal. Hence $\mathbf{h}_U, \mathbf{h}_V, \mathbf{h}_W$ are independent standard Gaussian vectors. We write

$$P_U = A_U^\top A_U, \qquad P_H = A_V^\top A_V + A_W^\top A_W \tag{57}$$

for the visible and hidden first-level projectors, and $d_1 = p_U + p_V + p_W$ for the total first-level dimension.

Let $\mathbf{a}_z \in \mathbb{R}^{p_U}$ be a fixed unit vector and $P_q = I_{p_U} - \mathbf{a}_z \mathbf{a}_z^\top$. Define

$$q = \frac{\mathbf{h}_U^\top P_q \mathbf{h}_U - (p_U - 1)}{\sqrt{2(p_U - 1)}}, \qquad z = \mathbf{a}_z^\top \mathbf{h}_U, \qquad S = \operatorname{sign}\left(\mathbf{h}_V^\top C \mathbf{h}_W \right), \qquad \|C\|_F = 1. \tag{58}$$

The standardized target is

$$y = \frac{\tanh\left(\dfrac{q + zS}{\sqrt{2}} + b_0 \right) - \mu_y}{\sigma_y}, \qquad b_0 = \frac{1}{2}, \tag{59}$$

where μ_y and $\sigma_y > 0$ are the population mean and standard deviation of the unnormalized output. Thus Eq. (21) corresponds to $A(\mathbf{h}_U) = q/\sqrt{2}$, $B(\mathbf{h}_U) = z/\sqrt{2}$ and a shifted $\tanh$ link. The numerical experiment uses the corresponding empirical centering and variance normalization on the training labels.

High-dimensional regime. For the high-dimensional regime we take

$$p_U, p_V, p_W \asymp d^\epsilon, \qquad d_1 \asymp d^\epsilon, \qquad 0 < \epsilon < 1, \qquad n \asymp d^\alpha. \tag{60}$$

For the hidden-subspace theorem we set $p_V = p_W = p_h$ and assume that C is full rank with balanced singular values

$$c_- p_h^{-1/2} \leq \sigma_j(C) \leq c_+ p_h^{-1/2}, \qquad 0 < c_- \leq c_+ < \infty. \tag{61}$$

This assumption fixes the scale $\|C\|_F = \Theta(1)$ while excluding hidden directions whose interaction strength vanishes asymptotically relative to the others.

For the population proof only, set $t_\pm = \frac{q \pm z}{\sqrt{2}} + b_0$, and

$$a(q,z) = \frac{\tanh(t_+) + \tanh(t_-) - 2\mu_y}{2\sigma_y}, \qquad b(q,z) = \frac{\tanh(t_+) - \tanh(t_-)}{2\sigma_y}. \tag{62}$$

Then

$$y = a(q,z) + b(q,z)S, \tag{63}$$

with a even and b odd in z. The functions a and b decompose the same teacher for the population proof.

E.2 BackLoFi in the deep-staircase setting

We now specify what the operations of Algorithms 1–2 become in the synthetic setting. The depth and final first-level rank d_1 are fixed to the teacher dimensions, while the planted directions themselves are never given to the learner.

For a scalar sample weight ξ and a representation $\mathbf{z}$, we define

$$\bar{\xi} = \frac{1}{n}\sum_{\mu=1}^{n} \xi_\mu, \qquad \widehat{m}_{\mathbf{z}}[\xi] = \frac{1}{n}\sum_{\mu=1}^{n} (\xi_\mu - \bar{\xi})\mathbf{z}_\mu, \qquad \widehat{C}_{\mathbf{z}}[\xi] = \frac{1}{n}\sum_{\mu=1}^{n} (\xi_\mu - \bar{\xi})\mathbf{z}_\mu \mathbf{z}_\mu^\top. \tag{64}$$

Their population counterparts are

$$m_{\mathbf{z}}[\xi] = \mathbb{E}[(\xi - \mathbb{E}\xi)\mathbf{z}], \qquad C_{\mathbf{z}}[\xi] = \mathbb{E}\left[(\xi - \mathbb{E}\xi)\mathbf{z}\mathbf{z}^\top\right]. \tag{65}$$

At the raw input we abbreviate $\widehat{m}[\xi] = \widehat{m}_{\mathbf{x}}[\xi]$, $\widehat{C}[\xi] = \widehat{C}_{\mathbf{x}}[\xi]$ and similarly for the population quantities. Since $\mathbf{x}$ is isotropic, $C[\xi] = \mathbb{E}[\xi(\mathbf{x}\mathbf{x}^\top - I_d)]$. Before applying a spectral step to a nonlinear hidden representation, the synthetic implementation applies the same empirical batch normalization used in the forward Neural LoFi synthetic setting (Dandi et al., 2026b). The same normalization is applied after a backward correction.

Forward student. Algorithm 1 uses (64) to construct each forward filter and then applies the Gaussian random lift. In the synthetic experiment both forward filters retain rank d_1, and a scalar readout is fitted at the top. We denote the resulting Forward-LoFi predictor by $\widehat{f}_F$ and its exact top-hidden sensitivity by

$$\mathbf{g}^2(\mathbf{x}) = \nabla_{\mathbf{h}_2} \widehat{f}_F(\mathbf{x}), \tag{66}$$

which is the same object as the sensitivity in Eq. (2) for the fitted student.

Backward score selection. At the first boundary, line 4 of Algorithm 2 specializes to

$$\widehat{A}_1 = \widehat{\mathbb{E}}_n\left[y\, \mathbf{g}^2(\mathbf{x})\mathbf{x}^\top\right] = U\Sigma V^\top. \tag{67}$$

Following line 5, each retained left singular vector u_i defines the learned scalar score

$$s_i(\mathbf{x}) = u_i^\top \mathbf{g}^2(\mathbf{x}). \tag{68}$$

This is the synthetic counterpart of the score construction in Eq. (10). The score is learned from the fitted student and need not be a function only of the true visible coordinates.

Modified-label correction. Each retained score defines the modified labels $\widetilde{y}_{\mu,i} = y_\mu s_i(\mathbf{x}_\mu)$. The ordinary second-order LoFi filter is then applied with these labels, yielding a rank-d_1 candidate subspace E_i for each score. For r retained scores, these candidate subspaces are combined with equal weight with the original forward subspace and truncated back to rank d_1, as specified in Algorithm 4. The score rank r is selected by validation.

The corrected lower representation is then followed by the ordinary upper Forward-LoFi step and a refit of the scalar readout. The paired column correction of Algorithm 2 is not used in this synthetic setting, since the quantity of interest is recovery of the missing first-level feature subspace.

The depth and retained first-level rank d_1 are fixed to the teacher dimensions, while the planted directions are never provided to the learner. The population and finite-sample analysis below isolates the same modified-label mechanism with an informative oracle sensitivity and does not analyze the multi-score aggregation used in Algorithm 4.

Algorithm 4 Covariance-row BackLoFi for the deep staircase

Require: Data $\{(\mathbf{x}_\mu, y_\mu)\}_{\mu=1}^n$, total retained rank d_1, candidate score ranks $\mathcal{R}$.

1: Fit the two-layer Forward-LoFi student with Algorithm 1, retaining rank d_1 in the lower filter, and fit its scalar readout $\widehat{f}_F$.
2: Compute the exact fitted sensitivity $\mathbf{g}^2$ by (66).
3: Compute the cross-layer SVD $\widehat{A}_1 = U\Sigma V^\top$ by (67).
4: **for** each candidate $r \in \mathcal{R}$ **do**
5: **for** $i = 1, \dots, r$ **do**
6: $s_i \leftarrow u_i^\top \mathbf{g}^2$ and $\widetilde{y}_i \leftarrow y s_i$.
7: $E_i \leftarrow$ the d_1 leading eigenvectors of $\widehat{C}[\widetilde{y}_i]$ by absolute eigenvalue.
8: **end for**
9: $F_{1,r}^{\mathrm{B}} \leftarrow$ the d_1 leading eigenspace of $F_1 F_1^\top + \sum_{i=1}^r E_i E_i^\top$.
10: Recompute the lower representation with $F_{1,r}^{\mathrm{B}}$, rebuild the upper Forward-LoFi filter, and refit the scalar readout.
11: **end for**
12: Select r by validation error and return the corresponding corrected predictor and rank-d_1 first-layer filter.

E.3 Population separation and signal strengths

We now analyze the population moments in (65). No new estimator is introduced in this subsection.

Lemma 4 (Forward visible gap and hidden cancellation)**.** *For the teacher* (59)*, $m[y] = 0$ and the population second-order operator has the form*

$$C[y] = A_U^\top M_U A_U, \qquad M_U = \kappa_{q,p_U} P_q + \kappa_{z,p_U} \mathbf{a}_z \mathbf{a}_z^\top, \tag{69}$$

where

$$\kappa_{q,p_U} = \sqrt{\frac{2}{p_U - 1}}\, \mathbb{E}[a(q,z)q], \qquad \kappa_{z,p_U} = \mathbb{E}[a(q,z)(z^2-1)]. \tag{70}$$

Moreover,

$$\sqrt{p_U - 1}\, \kappa_{q,p_U} \longrightarrow \frac{\mathbb{E}[\tanh'(b_0 + Z)]}{\sigma_\infty} > 0, \qquad \kappa_{z,p_U} \longrightarrow \frac{\mathbb{E}[\tanh''(b_0 + Z)]}{2\sigma_\infty} \neq 0, \tag{71}$$

where $Z \sim \mathcal{N}(0,1)$ and $\sigma_\infty^2 = \mathrm{Var}(\tanh(b_0 + Z))$. Hence the smallest nonzero absolute eigenvalue satisfies

$$\Delta_U := \min_{\lambda \neq 0} |\lambda(C[y])| \asymp p_U^{-1/2}. \tag{72}$$

In particular, the forward second-order statistic recovers the complete visible U-subspace, while all V/V, W/W, V/W and visible–hidden blocks vanish.

Proof. Averaging (63) over the balanced hidden sign gives $\mathbb{E}[y \mid \mathbf{h}_U] = a(q,z)$. The first moment vanishes because $a(q,z)$ is invariant under $\mathbf{h}_U \mapsto -\mathbf{h}_U$ and $\mathbb{E}[S] = \mathbb{E}[b(q,z)] = 0$ removes the remaining terms. Rotational symmetry in $\mathrm{range}(P_q)$ gives (69)–(70). For the hidden cross-block,

$$\mathbb{E}[y \mathbf{h}_V \mathbf{h}_W^\top] = \mathbb{E}[b(q,z)]\mathbb{E}[S \mathbf{h}_V \mathbf{h}_W^\top] = 0, \tag{73}$$

and the same sign symmetries remove the other hidden and mixed blocks.

The standardized chi-square coordinate q converges to a standard Gaussian independently of z. Since a is bounded and $\mathbb{E}[q^2] = 1$, uniform integrability then reduces the limit of $\mathbb{E}[a(q,z)q]$ to two independent standard Gaussians. A one-dimensional Gaussian integration by parts gives the first limit in (71). Since z is exactly standard Gaussian and independent of q, $\mathbb{E}[a(q,z)(z^2-1)] = \mathbb{E}[\partial_z^2 a(q,z)]$. Taking $p_U \to \infty$ gives the second limit. For $b_0 > 0$ the latter is negative because $b \mapsto \mathbb{E}[\tanh'(b + Z)]$ is strictly decreasing for $b > 0$. The weaker visible eigenvalues are therefore those in $\mathrm{range}(P_q)$, of order $p_U^{-1/2}$. □

Oracle upper sensitivity. To isolate the backward mechanism, suppose temporarily that the visible conditional predictor is known. By (62), $\mathbb{E}[y \mid q, z] = a(q, z)$. Its two coordinate sensitivities are

$$s_q^\star = \frac{\tanh'(t_+) + \tanh'(t_-)}{2\sqrt{2}\,\sigma_y}, \qquad s_z^\star = \frac{\tanh'(t_+) - \tanh'(t_-)}{2\sqrt{2}\,\sigma_y}. \tag{74}$$

These derivatives of the visible conditional predictor with respect to its two second-level coordinates serve as oracle substitutes for the learned score in the recovery theorem.

Lemma 5 (Informative oracle sensitivity)**.** *The q-sensitivity is uninformative for the hidden interaction, while the z-sensitivity has a nonzero limiting coupling:*

$$\mathbb{E}[b(q, z)s_q^\star] = 0, \qquad \theta_{p_U} := \mathbb{E}[b(q, z)s_z^\star] \longrightarrow \theta_\infty < 0. \tag{75}$$

More explicitly, with $T = b_0 + Z$,

$$\theta_\infty = \frac{\mathrm{Cov}(\tanh(T), \tanh'(T))}{2\sqrt{2}\,\sigma_\infty^2}. \tag{76}$$

Proof. The function $b(q, z)$ is odd in z, while $s_q^\star$ is even, which gives the first identity. The score $s_z^\star$ is odd. In the Gaussian limit the variables t_+ and t_- become independent copies of $T = b_0 + Z$, and expanding the product gives (76). Since $\tanh'(T) = 1 - \tanh^2(T)$,

$$\mathrm{Cov}(\tanh(T), \tanh'(T)) = -\,\mathrm{Cov}(\tanh(T), \tanh^2(T)). \tag{77}$$

For $b_0 > 0$, conditioning on $|T| = r$ gives $\mathbb{E}[\tanh(T) \mid |T| = r] = \tanh(r)\tanh(b_0 r)$, which is strictly increasing in r, as is $\tanh^2(r)$. The last covariance is therefore strictly positive and $\theta_\infty < 0$. □

Lemma 6 (Hidden interaction moment)**.** *Let*

$$M_C := \mathbb{E}\left[S\mathbf{h}_V \mathbf{h}_W^\top\right]. \tag{78}$$

Under (61)*, M_C has the same left and right singular vectors as C and*

$$\sigma_{\min}(M_C) \asymp \|M_C\|_{\mathrm{op}} \asymp p_h^{-1/2}. \tag{79}$$

Proof. Condition on $\mathbf{h}_W$. For a standard Gaussian vector G and any nonzero vector a,

$$\mathbb{E}\left[G\,\mathrm{sign}(a^\top G)\right] = \sqrt{\frac{2}{\pi}}\frac{a}{\|a\|_2}. \tag{80}$$

Applying this identity with $a = C\mathbf{h}_W$ yields

$$M_C = \sqrt{\frac{2}{\pi}}\,\mathbb{E}\left[\frac{C\mathbf{h}_W\mathbf{h}_W^\top}{\|C\mathbf{h}_W\|_2}\right]. \tag{81}$$

Write $C = L\,\mathrm{diag}(c_1, \ldots, c_{p_h})R^\top$ and rotate $\mathbf{h}_W = RZ$. Sign symmetry makes the middle matrix diagonal, with ith entry

$$\sqrt{\frac{2}{\pi}}\,c_i\,\mathbb{E}\left[\frac{Z_i^2}{(\sum_j c_j^2 Z_j^2)^{1/2}}\right]. \tag{82}$$

Under (61), the denominator is of order one and the expectation is bounded above and below by constants independent of i and p_h. Each singular value therefore has order $p_h^{-1/2}$. □

Proposition 4 (Population forward/backward separation)**.** *For the oracle score $s_z^\star$, the modified-label population operator has no visible second-order block and is supported on the hidden subspace:*

$$C[ys_z^\star] = A_H^\top \begin{pmatrix} 0 & \theta_{p_U} M_C \\ \theta_{p_U} M_C^\top & 0 \end{pmatrix} A_H, \qquad A_H = \begin{pmatrix} A_V \\ A_W \end{pmatrix}. \tag{83}$$

Consequently its nonzero absolute eigenvalues are the singular values of $\theta_{p_U} M_C$, and its hidden spectral gap obeys

$$\Delta_B := \min_{\lambda \neq 0} |\lambda(C[ys_z^\star])| \asymp p_h^{-1/2}. \tag{84}$$

Together with Lemma 4, this gives the population separation displayed in Eq. (22)*.*

Proof. The visible contribution to the modified label is $a(q,z)s_z^\star$, which is odd in z. Its visible second-order moment therefore vanishes. The hidden term factorizes as

$$\mathbb{E}\left[ys_z^\star \mathbf{h}_V \mathbf{h}_W^\top\right] = \mathbb{E}[b(q,z)s_z^\star]M_C = \theta_{p_U} M_C. \tag{85}$$

The V/V, W/W and mixed visible–hidden blocks vanish by Gaussian sign symmetry. The block matrix in (83) has eigenvalues equal to plus or minus the singular values of $\theta_{p_U} M_C$. Lemmas 5 and 6 then give (84). □

E.4 Finite-sample subspace recovery

We now turn the two population gaps into a finite-sample feature-recovery statement. Let $\widehat{P}_U$ denote the projector onto the p_U eigenvectors of $\widehat{C}[y]$ with largest absolute eigenvalues, and let $\widehat{P}_H$ denote the projector onto the $p_V + p_W$ eigenvectors of $\widehat{C}[ys_z^\star]$ with largest absolute eigenvalues. The result retains the oracle sensitivity $s_z^\star$.

Lemma 7 (Weighted Gaussian covariance concentration)**.** *Let $\xi = \xi(\mathbf{x})$ satisfy $|\xi| \le L$ almost surely and let $\widehat{C}[\xi]$ be defined by* (64) *at $\mathbf{z} = \mathbf{x}$. For $0 < \eta < 1$, set*

$$\gamma_{n,d,\eta} = \sqrt{\frac{d + \log(4/\eta)}{n}} + \frac{d + \log(4/\eta)}{n}. \tag{86}$$

There is a universal constant $c > 0$ such that, with probability at least $1 - \eta$,

$$\|\widehat{C}[\xi] - C[\xi]\|_{\mathrm{op}} \le cL\gamma_{n,d,\eta}. \tag{87}$$

The same statement holds simultaneously for any fixed finite collection of bounded weights after replacing η by a constant multiple of η.

Proof. For a fixed unit vector v, the weighted quadratic form $\xi((v^\top \mathbf{x})^2 - 1)$ is sub-exponential with parameters controlled by L. Bernstein's inequality and a $1/4$-net of the unit sphere give the stated bound for $n^{-1}\sum_\mu \xi_\mu(\mathbf{x}_\mu\mathbf{x}_\mu^\top - I_d)$. The difference between this Hermite-centered matrix and $\widehat{C}[\xi]$ is $\bar{\xi}(n^{-1}\sum_\mu \mathbf{x}_\mu\mathbf{x}_\mu^\top - I_d)$, which has the same order by the standard Gaussian sample-covariance bound. A union bound gives the finite-collection statement. □

Theorem 3 (Oracle-sensitivity recovery in the deep staircase)**.** *Assume the teacher of Section E.1, with $p_V = p_W = p_h$ and* (61)*. Let $\widehat{P}_U$ and $\widehat{P}_H$ be the known-rank spectral projectors defined above. There are constants $c_\star, C_\star > 0$, independent of d, p_U, p_h, n, such that for all sufficiently large p_U, with probability at least $1 - \eta$,*

$$\|\widehat{P}_U - P_U\|_{\mathrm{op}} + \|\widehat{P}_H - P_H\|_{\mathrm{op}} \le C_\star\sqrt{\max\{p_U, p_h\}}\,\gamma_{n,d,\eta}, \tag{88}$$

provided the right-hand side is smaller than $c_\star$. In particular, the complete first-level latent subspace is consistently recovered if

$$\frac{n}{d\max\{p_U, p_V, p_W\}} \longrightarrow \infty. \tag{89}$$

For balanced blocks $p_U, p_V, p_W \asymp d^\epsilon$, this is the sufficient feature-recovery scale

$$n \gg d^{1+\epsilon}, \qquad \textit{equivalently} \qquad \alpha > 1 + \epsilon. \tag{90}$$

Proof. The weights y and $ys_z^\star$ are uniformly bounded. Lemma 7 therefore controls both empirical operators at scale $\gamma_{n,d,\eta}$. Lemma 4 gives a forward spectral gap $\Delta_U \asymp p_U^{-1/2}$. Spectral-projector perturbation, for instance the Davis–Kahan bound (Yu et al., 2015), yields

$$\|\widehat{P}_U - P_U\|_{\mathrm{op}} \lesssim \sqrt{p_U}\,\gamma_{n,d,\eta}. \tag{91}$$

Proposition 4 gives the backward gap $\Delta_B \asymp p_h^{-1/2}$ and the same perturbation argument gives

$$\|\widehat{P}_H - P_H\|_{\mathrm{op}} \lesssim \sqrt{p_h}\,\gamma_{n,d,\eta}. \tag{92}$$

Combining the two estimates proves (88). Condition (89) makes both normalized perturbations vanish. □

Scope of the recovery theorem. Theorem 3 is conditional on an informative oracle sensitivity and proves the recovery of the first level subspace. Accordingly, the scale in (90) is not a prediction for the finite dimensional transition of Algorithm 4 with learned scores. A prediction error theorem would additionally require controlling the learned sensitivity, the upper Forward LoFi refit, and the final readout.

E.5 From oracle to learned upper sensitivities

Theorem 3 isolates the backward mechanism by replacing the score selected through (67) with the informative coordinate sensitivity $s_z^\star$. The implemented estimator instead differentiates the fitted Forward-LoFi student and uses the left singular modes of $\widehat{A}_1$ as in lines 4–5 of Algorithm 2. The following condition is sufficient for a learned score to preserve the hidden population gap.

Proposition 5 (Conditional learned-score transfer)**.** *Condition on a fitted upper student and let $\widehat{s}$ be a normalized learned score. Suppose*

$$\widehat{s} = c\, s_z^\star + e, \qquad |c| \geq c_0 > 0, \tag{93}$$

with uniformly bounded effective labels, and assume

$$\|C[ye]\|_{\mathrm{op}} = o_{\mathbb{P}}(p_h^{-1/2}). \tag{94}$$

Then the population operator $C[y\widehat{s}]$ has a hidden spectral gap of order $p_h^{-1/2}$ with the same limiting hidden subspace as $C[ys_z^\star]$. On an independent estimation sample satisfying (89)*, its known-rank empirical eigenspace consistently recovers P_H.*

Proof. By linearity, $C[y\widehat{s}] = cC[ys_z^\star] + C[ye]$. Proposition 4 gives a nonzero hidden gap of order $p_h^{-1/2}$ for the first term. Condition (94) is negligible on this scale, so spectral perturbation preserves the hidden eigenspace. Conditional on the fitted score, Lemma 7 controls an independent estimation sample and the same Davis–Kahan argument applies. □

For several retained score channels, it is sufficient that their span contain a normalized combination satisfying (93) and (94). The proposition does not analyze the finite sample selection of this informative direction by Algorithm 4. The numerical procedure also reuses the same training samples across the forward fit, score selection and covariance correction, whereas the conditional statement above uses an independent estimation sample.

Finite random feature student. The fitted sensitivity in (66) is computed from the finite random feature Forward LoFi student. Random feature Hermite expansions and their relation to hierarchical spectral recovery are developed for Forward Neural LoFi in Dandi et al. (2026b). We do not prove here that the complete finite width fitted student satisfies (94).

F The implemented algorithm

This appendix gives Algorithm 2 exactly as it is run for the experiments, with the choices that the main text leaves open. The values of every hyperparameter are listed in Appendix H.

Parameterization. A layer is a filter $\mathbf{F}_\ell \in \mathbb{R}^{p_{\ell-1}\times K_\ell}$ followed by a Gaussian lift $\mathbf{R}_\ell \in \mathbb{R}^{p_\ell \times K_\ell}$ with i.i.d. $\mathcal{N}(0,1)$ entries,

$$\mathbf{h}_\ell = \frac{1}{c_\ell}\mathbf{R}_\ell\mathbf{F}_\ell^\top \mathbf{z}_{\ell-1}, \qquad \mathbf{z}_\ell = \frac{\sigma(\mathbf{h}_\ell)}{\sqrt{p_\ell}}, \qquad \mathbf{D}_\ell = \frac{\mathrm{diag}(\sigma'(\mathbf{h}_\ell))}{\sqrt{p_\ell}}, \qquad c_\ell^2 = \widehat{\mathbb{E}}_n\|\mathbf{F}_\ell^\top \mathbf{z}_{\ell-1}\|_2^2, \tag{95}$$

so that $\mathbf{W}_\ell = \mathbf{R}_\ell\mathbf{F}_\ell^\top/c_\ell$. The scale c_ℓ is fitted with the forward network and never refitted. A cutoff k_ℓ stores $K_\ell = k_\ell + 1$ columns, the normalized first moment followed by k_ℓ eigenvectors of $\widehat{\mathbf{C}}_\ell$ deflated against it. Each column carries a tag, FWD, READ or PAIR, according to whether it was installed by forward Neural LoFi, by a row step or by a column step. The rank of every boundary is $r_\ell = \max\{1, \mathrm{round}(fk)\}$ for a rank fraction f. The widths never change: every append replaces r_ℓ existing columns.

Algorithm 5 One pass of Algorithm 2, as implemented.

Require: Network $(\mathbf{F}_\ell, \mathbf{R}_\ell, c_\ell)_{\ell \le L}$ and readout $(\mathbf{a}_L, b)$; rank fraction f, scale β; signal $\in$ {first-order, exact}.

Compute $\mathbf{z}_\ell, \mathbf{D}_\ell$ by (95); $\mathbf{g}^L \leftarrow \mathbf{D}_L \mathbf{a}_L$; keep a copy $(\mathbf{F}_\ell^0, \mathbf{R}_\ell^0)$ of the network
for $\ell = L-1, \dots, 1$ **do**
  $(\mathbf{U}, \mathbf{V}) \leftarrow$ top r_ℓ singular pairs of $\mathbf{A}_\ell = \widehat{\mathbb{E}}_n[y\, \mathbf{g}^{\ell+1} \mathbf{z}_{\ell-1}^\top]$; $\mathbf{s} \leftarrow \mathbf{U}^\top \mathbf{g}^{\ell+1}$ ▷ frozen scores
  $I \leftarrow$ the r_ℓ columns of $\mathbf{F}_\ell$ of lowest importance (97); $\tau_\ell \leftarrow$ mean coordinate RMS of the kept FWD and READ columns
  $\mathbf{F}_\ell[:, I] \leftarrow \mathbf{v}_i\, \beta \tau_\ell / \operatorname{RMS}\langle \mathbf{v}_i, \mathbf{z}_{\ell-1} \rangle$; $\mathbf{R}_\ell[:, I] \leftarrow$ fresh $\mathcal{N}(0,1)$ ▷ row step
  Recompute $\widetilde{\mathbf{z}}_\ell, \mathbf{D}_\ell$; $\mathbf{Q} \leftarrow \widehat{\mathbb{E}}_n[\widetilde{\mathbf{z}}_\ell (y\mathbf{s})^\top]$
  $J \leftarrow$ the r_ℓ columns of $\mathbf{F}_{\ell+1}$ of lowest importance; $\tau_{\ell+1}$ as above
  $\mathbf{F}_{\ell+1}[:, J] \leftarrow \mathbf{q}_i\, \beta \sqrt{p_{\ell+1}}\, \tau_{\ell+1} / \operatorname{RMS}\langle \mathbf{q}_i, \widetilde{\mathbf{z}}_\ell \rangle$; $\mathbf{R}_{\ell+1}[:, J] \leftarrow \mathbf{U}$ ▷ column step
  if $\ell > 1$ **then**
    first-order: $\mathbf{g}^\ell \leftarrow \mathbf{D}_\ell \big(\widehat{\mathbb{E}}_n \big[\frac{1}{c_{\ell+1}} \mathbf{F}_{\ell+1}^0 (\mathbf{R}_{\ell+1}^0)^\top \mathbf{g}^{\ell+1} \big] + \frac{1}{c_{\ell+1}} \mathbf{Q}^0 \mathbf{s} \big)$, $\mathbf{q}_i^0 \propto \frac{1}{c_\ell} \mathbf{R}_\ell^0 (\mathbf{F}_\ell^0)^\top \mathbf{v}_i$
    exact: $\mathbf{g}^\ell \leftarrow \mathbf{D}_\ell \mathbf{W}_{\ell+1}^\top \mathbf{g}^{\ell+1}$ through the corrected $\mathbf{W}_{\ell+1}$, per sample
  end if
end for
Recompute $\mathbf{z}_L$; refit $(\mathbf{a}_L, b)$ by ridge regression

Scales. The only free number is β, the size of an appended coordinate in units of a forward one. It sets the amplitude of every column appended by the row step and by the column step. Write $\operatorname{RMS}\, \xi = (\widehat{\mathbb{E}}_n \xi^2)^{1/2}$ for the root mean square over the training set, and let τ_ℓ be the mean of $\operatorname{RMS}\langle \mathbf{f}_a, \mathbf{z}_{\ell-1} \rangle$ over the surviving columns $\mathbf{f}_a$ of $\mathbf{F}_\ell$ that carry a random lift (FWD and READ). The row step appends $\gamma_i^{\mathrm{row}} \mathbf{v}_i$ to $\mathbf{F}_\ell$ and the column step appends $\gamma_i^{\mathrm{col}} \mathbf{q}_i$ to $\mathbf{F}_{\ell+1}$, with

$$\gamma_i^{\mathrm{row}} = \frac{\beta\, \tau_\ell}{\operatorname{RMS}\langle \mathbf{v}_i, \mathbf{z}_{\ell-1} \rangle}, \qquad \gamma_i^{\mathrm{col}} = \frac{\beta \sqrt{p_{\ell+1}}\, \tau_{\ell+1}}{\operatorname{RMS}\langle \mathbf{q}_i, \widetilde{\mathbf{z}}_\ell \rangle}. \tag{96}$$

In the notation of (11), where the factor $1/c$ of (95) is absorbed, this reads $\mathbf{\Gamma}_\ell^{\mathrm{row}} = \operatorname{diag}(\gamma_i^{\mathrm{row}})/c_\ell$ and $\mathbf{\Gamma}_\ell^{\mathrm{col}} = \operatorname{diag}(\gamma_i^{\mathrm{col}})/c_{\ell+1}$. An appended read coordinate thus has root mean square $\beta\tau_\ell$. A paired column carries the extra factor $\sqrt{p_{\ell+1}}$ because its lift column $\mathbf{u}_i$ has unit norm, while a forward column is lifted by $\mathcal{N}(0,1)$ entries; with it, the paired column contributes to a neuron of layer $\ell+1$ as much as β forward coordinates. Each γ_i is set per column, which equalizes the amplitudes of the retained modes. The columns $\mathbf{q}_i^0$ of the first-order signal are rescaled to the same target as $\gamma_i^{\mathrm{col}} \mathbf{q}_i$. We use $\beta = 0.25$ for the fully connected network and $\beta = 4$ for the convolutional one, each chosen by a local scan at $n = 10^4$.

Which columns are replaced. The widths are fixed, so each append replaces r_ℓ existing columns: the row step replaces the set I of columns of $\mathbf{F}_\ell$, and the column step the set J of columns of $\mathbf{F}_{\ell+1}$. In both cases we remove the r_ℓ columns of lowest importance Ω, and the rule treats forward, read and paired columns alike at every pass. Importance is defined as in forward Neural LoFi, where a direction matters when gradient flow amplifies its coefficients in the rows, but it is computed on the current network, whose layers above are no longer random. For a column a of layer m it is built in three steps.

1. *Coefficients.* Let $\widehat{\mathbf{v}}_a$ be the unit direction of the column and $h_a = \langle \widehat{\mathbf{v}}_a, \mathbf{z}_{m-1} \rangle$ its coordinate. Neuron j reads it with the coefficient $c_{j,a} = (\mathbf{R}_m)_{ja} \|\mathbf{v}_a\| / c_m$, and sees the effective label $y\, \langle \boldsymbol{\omega}_j, \mathbf{g}^{m+1} \rangle$, where $\boldsymbol{\omega}_j$ is the column of $\mathbf{W}_{m+1}$ that reads it.
2. *Rate.* Project the row update (4) on $\widehat{\mathbf{v}}_a$, with the label in place of the residual and the gate expanded to first degree, $\sigma'(h) \simeq d_m + e_m h$ with $(d_m, e_m) = (\widehat{\mathbb{E}}_n \sigma'(h), \widehat{\mathbb{E}}_n h\sigma'(h))$. This gives $\dot{c}_{j,a} \propto e_m\, \lambda_j(a)\, c_{j,a} + d_m\, \ell_j(a)$, with the drivers λ_j, ℓ_j given in (97).
3. *Importance.* Take the root mean square of this rate over the neurons of the layer:

$$\begin{gathered} \lambda_j(a) = \big\langle \boldsymbol{\omega}_j, \widehat{\mathbb{E}}_n[y\, \widehat{\mathbf{g}}\, h_a^2] \big\rangle, \qquad \ell_j(a) = \big\langle \boldsymbol{\omega}_j, \widehat{\mathbb{E}}_n[y\, \widehat{\mathbf{g}}\, h_a] \big\rangle, \\ \Omega(a) = \operatorname{RMS}_j \big(e_m\, \lambda_j(a)\, c_{j,a} + d_m\, \ell_j(a) \big). \end{gathered} \tag{97}$$

A column whose coefficients no neuron would move can thus be removed at the least cost. In the rate, the term in λ_j is the covariance channel, multiplicative in the coefficient, and the term in ℓ_j the spike channel, additive; only the diagonal of the driver is kept, since the drive of $\widehat{\mathbf{v}}_a$ by the other directions a neuron reads averages out over neurons. The backward signal enters through $\widehat{\mathbf{g}} = \bar{\mathbf{g}} + \mathbf{U}\mathbf{U}^\top(\mathbf{g}^{m+1} - \bar{\mathbf{g}})$, its mean plus its sample dependence along the retained modes of the boundary above, the only part that reaches the neurons with a coherent sign. When the backward signal carries no sample dependence, $\widehat{\mathbf{g}} = \bar{\mathbf{g}}$, $\lambda_j(a)$ is $\langle \boldsymbol{\omega}_j, \bar{\mathbf{g}} \rangle \, \widehat{\mathbb{E}}_n[y h_a^2]$, and Ω orders the covariance directions by the signed Rayleigh quotient that forward Neural LoFi uses, weighted by how strongly the neurons read them. For the top layer $\boldsymbol{\omega}_j = (\mathbf{a}_L)_j$ and $\widehat{\mathbf{g}} \equiv 1$.

Convolutional network. The network is conv1 (3×3, no filter), conv2 (3×3, filter k_c over channels) with pooling, and a fully connected layer (filter k_3). At the boundary conv2 — FC, $\mathbf{A}$ pairs the FC sensitivity with the spatial mean of conv1's output; the row step replaces channel directions of conv2, each with a fresh random 3×3 kernel, and the column step replaces FC filter columns by $\mathbf{Q}$ computed on the pooled map, with all $H'W'$ lift rows of a paired channel equal to $\mathbf{u}_i$ and the target divided by $\sqrt{H'W'}$ to match a forward channel. Conv1 has no filter, so the boundary conv1 — conv2 performs the column step only, with locations treated as samples and $\mathbf{u}_i$ placed on the centre tap. The backward signal is carried through the pooling by its adjoint, and the importance is computed per filter and tap. Every statistic is streamed over the training set.

Numerics. The top r_ℓ singular pairs are computed exactly when $r_\ell + 32$ exceeds half the smaller dimension of $\mathbf{A}_\ell$ and by a randomized SVD otherwise (oversampling 32, four power iterations, fixed seed). Statistics are chunked averages over the training set, accumulated in double precision.

F.1 Variants tested and not retained

We think it is useful to report approaches that did not work, so that others need not try them again. The following variants were implemented and run on the fully connected network, and none was retained.

Gate-only signal ($\mathbf{g}^\ell = \mathbf{D}_\ell \mathbf{1}$): no gain over forward Neural LoFi, as expected, since its read-directions stay in the span of the forward filter (Section 2).

Mean-field and exact signals: the same test error as the first-order signal within seed noise; the first-order signal is kept because it needs no propagation through the installed blocks.

Common column scale (the mean or the largest paired column set to $\beta\sqrt{p_{\ell+1}}\tau_{\ell+1}$, relative amplitudes kept): no improvement over forward Neural LoFi; only the per-column scale did improve.

Scale τ over all columns: paired columns carry $\sqrt{p}$-larger coordinates, so including them inflates τ by a factor of about 6 per pass, and the later appends with it.

Weakest swap (the lowest Rayleigh quotients $|\widehat{\mathbb{E}}_n[y h_a^2]|$): backward columns score high, so passes accumulate them and consume the forward columns.

Replace-own swap (each pass rewrites the slots of the previous one): no gain beyond the first pass.

Update of the whole block $\ell + 1$ (forward columns refitted on $\widetilde{\mathbf{z}}_\ell$ before the swap): the same test error, within 0.1 points at every n.

Dense interpolation ($\mathbf{W} \leftarrow (1-\alpha)\mathbf{W} + \alpha$ estimate, $\alpha = 0.1$, no swap): the same test error, within 0.1 points at every n.

G Additional numerical simulations

G.1 Number of passes at fixed width

Figure 4 shows the test error of the fully connected network against the number of passes of Algorithm 2 at fixed width. With $r_\ell = 5\%$, 7%, 10% and 12% of k_ℓ the error decreases with the passes to the same minimum, reached when about half of the filter columns have been replaced (pass 4 at 12%, pass 5 at 10%, pass 7–8 at 7%, pass 9–10 at 5%); further passes exhaust the forward columns

and the error rises quickly. With $r_\ell = 3\%$ the error is still decreasing slowly after ten passes, and with $r_\ell = 2\%$ it stays at the forward level.

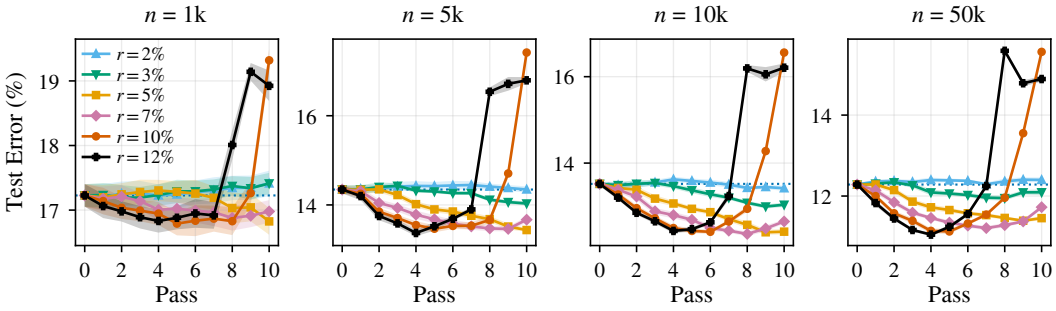

Figure 4: Test error of the fully connected network against the number of passes of Algorithm 2 at fixed width, for six rank fractions r_ℓ/k_ℓ and four training-set sizes n. Pass 0 is forward Neural LoFi (dotted line). Each point is the best filter width k_ℓ for its n, rank fraction and pass, selected by the mean over 10 seeds; bands are one standard error. Settings in Appendix H.

G.2 What successive passes select

Figure 5 shows, for each pass, the fraction of its new read and write directions that lies in the span of those installed by the earlier passes. For the read directions it stays above 0.97, against at most 0.02 for a random subspace: each pass selects again the same read subspace and installs it with fresh lifts. Successive passes therefore resample random features within the same subspace. The error in Figure 4 consequently depends on the fraction of columns replaced and is insensitive to the per-pass step r_ℓ.

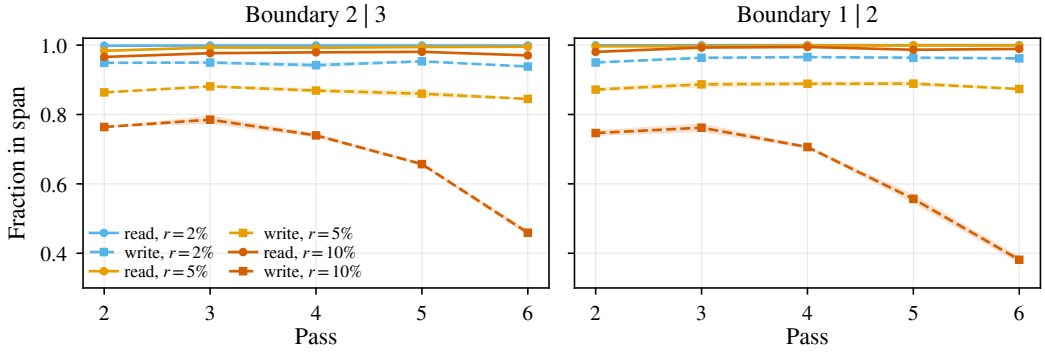

Figure 5: Fraction of the read (full) and write (dashed) directions selected at a pass inside the span of those installed by the earlier passes, $\|V_t^\top Q_{<t}\|_F^2/r_\ell$, at the two boundaries of the fully connected network ($k_\ell = 512$, $n = 10{,}000$, mean and standard error over 3 seeds). A random subspace gives at most 0.02.

G.3 Damped updates

The fixed-width version of Algorithm 2 used in Figure 3 replaces at each pass the r_ℓ forward columns of lowest importance by the corrected ones, so that the passes consume the forward columns (Appendix G.1). Figure 6 tests a rule that consumes nothing: the layers are kept dense and the two updates of (11) are applied as damped averages,

$$\mathbf{W}_\ell \leftarrow (1-\alpha)\,\mathbf{W}_\ell + \alpha\,\mathbf{R}_\ell^{\leftarrow}\mathbf{\Gamma}_\ell^{\mathrm{row}}\mathbf{V}_\ell^\top, \qquad \mathbf{W}_{\ell+1} \leftarrow (1-\alpha)\,\mathbf{W}_{\ell+1} + \alpha\,\mathbf{U}_\ell\mathbf{\Gamma}_\ell^{\mathrm{col}}\mathbf{Q}_\ell^\top, \tag{98}$$

with $\alpha = 0.1$, the same estimates, no column replaced and the readout refitted after each pass. With $r_\ell = 25\%$ of k_ℓ the error decreases monotonically over about 25 passes to 18.7%, 16.9%, 13.6%,

12.6% and 11.0% at $n = 500$, 1,000, 5,000, 10,000 and 50,000, within 0.2 points of the minimum of the swap (18.7%, 16.8%, 13.4%, 12.4% and 11.1%) and never significantly below it, then rises slowly, by 0.15–0.25 points at pass 50; with $r_\ell = 10\%$ the descent is slower (minimum at passes 30–48 for $n \geq 5{,}000$) and stops 0.2 points higher. The damped rule reaches essentially the same minimum error as the swap over more passes, followed by a gradual rise. This is consistent with the identity $(1-\alpha)\mathbf{W} + \alpha\mathbf{E} = (1-\alpha)\big(\mathbf{W} + \frac{\alpha}{1-\alpha}\mathbf{E}\big)$: the network is positively homogeneous and the ridge readout absorbs the factor $1-\alpha$, so the damped rule is the update (11) with the scale β replaced by $\beta\alpha/(1-\alpha)$, applied at every pass. Both methods use the same direction-selection rule.

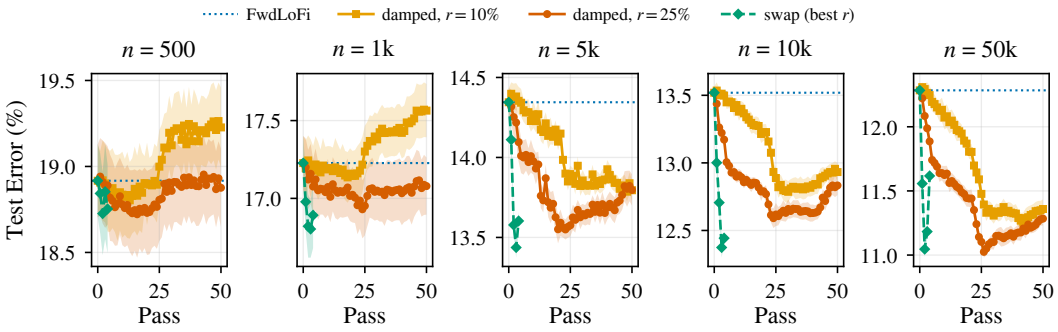

Figure 6: Test error of the fully connected network against the number of passes of the damped update (98) with $\alpha = 0.1$, at two rank fractions r_ℓ/k_ℓ, and of the fixed-width swap of Figure 3 at its best rank fraction (dashed), for five training-set sizes n. Pass 0 is forward Neural LoFi (dotted line). Each point is the best filter width k_ℓ for its n, rank fraction and pass, selected by the mean over 10 seeds; bands are one standard error. Settings in Appendix H.

G.4 Gradient-descent training curves

Figures 7 and 8 show the test error of full-batch gradient descent along training for the learning rates that do not diverge, on the fully connected and on the convolutional network. On the fully connected network the rate 0.1 is the best at every n. On the convolutional network the rate 0.003 is the best up to $n = 5{,}000$ and the largest stable rate, 0.005, from $n = 10{,}000$ upwards; the rate 0.001 is behind at every n. At large n the error is still decreasing at the last step on both networks, so the gradient-descent curves of Figure 3 are limited by the step budget there (3,000 steps on the fully connected network, 5,000 on the convolutional one).

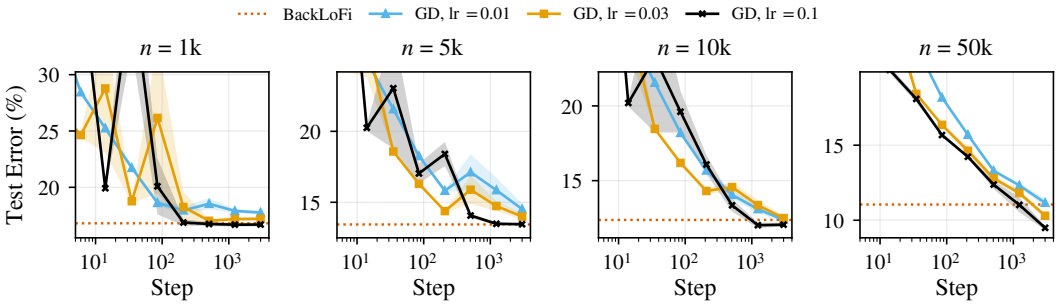

Figure 7: Test error of full-batch gradient descent on the fully connected network against the step, for three learning rates; mean and standard error over 5 seeds. Dotted: BackLoFi of Figure 3.

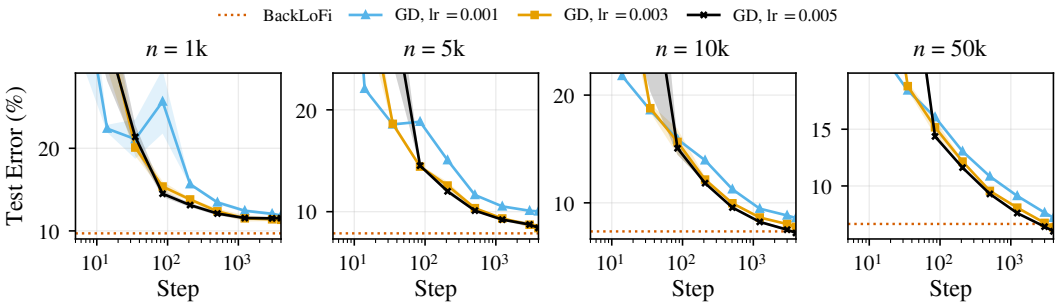

Figure 8: Test error of full-batch gradient descent on the convolutional network against the step, for three learning rates at every n; mean and standard error over 3 seeds for the rates 0.003 and 0.005, one seed for 0.001 at $n \geq 5{,}000$. Dotted: BackLoFi of Figure 3.

H HYPERPARAMETERS OF THE FIGURES

The image experiments share one protocol: binary CIFAR-10 (vehicles $+1$, animals -1), a training sub-sample of size n, the 10^4 test images, a ridge readout with leave-one-out penalty among 50 log-spaced values in $[10^{-6}, 10^6]$, test error as the fraction of wrong signs, means over seeds with every selection made on the seed mean. Algorithmic terms are defined in Appendix F.

Figure 2. Deep-staircase teacher of Appendix E, with $d = 256$, $(p_U, p_V, p_W) = (6, 5, 5)$, $d_1 = 16$, interaction strength 1, and output shift $b_0 = 1/2$. Training labels are standardized using the training-set mean and standard deviation, which are then kept fixed for validation and test data. The validation and test sizes are respectively 16,384 and 32,768.

The spectral estimators follow Algorithm 4. The two random-feature widths are $(512, 1024)$ and the retained ranks are $(16, 16)$. The activation is $\sigma(t) = \mathrm{ReLU}(t) - 1/\sqrt{2\pi}$, hidden biases are zero, and the Gaussian lifts are kept fixed between the forward and corrected fits. Layer normalizations are estimated on the training set. The score rank is selected from $r \in \{0, 1, 2, 4, 6\}$ by validation MSE. The final linear ridge readout has an unpenalized intercept, with regularization selected from $\{10^{-8}, 10^{-7}, \ldots, 10^{-1}, 1\}$ by validation MSE. The spectral curves use $\alpha \in \{1.0, 1.1, \ldots, 3.0\}$, $n = \mathrm{round}(256^{\alpha})$, and seeds $\{1, 2, 4, 5, 6\}$.

The gradient-descent baselines use two hidden layers of widths $(512, 1024)$ with the same centered-ReLU activation. AdamW is used with batch size $\min(n, 1024)$, weight decay 10^{-4}, gradient clipping at norm 5, and learning rate selected from $\{3 \times 10^{-4}, 10^{-3}\}$ by validation. The end-to-end baseline is trained for $T = 1000$ minibatch updates. For layer-wise training, $T/2$ updates are used at each of the two stages. The best validation checkpoint is retained, followed by the same linear ridge refit as for the spectral estimators. The GD curves use $\alpha \in \{1.0, 1.2, \ldots, 3.0\}$ and seeds $\{1, 2, 4\}$.

Prediction is evaluated by normalized test MSE. Feature recovery is measured from the final rank-16 first-layer row space. Curves report the mean over seeds and shaded regions one standard error of the mean.

Figure 3. $n \in \{500, 1\mathrm{k}, 2\mathrm{k}, 5\mathrm{k}, 10\mathrm{k}, 20\mathrm{k}, 50\mathrm{k}\}$. *Left:* $L = 3$ fully connected ReLU layers, $p_\ell = 5{,}000$, 10 seeds. FwdLoFi: $k_\ell \in \{32, \ldots, 2{,}048\}$ (powers of two), best per n. BackLoFi (1st ord): fixed width, $r_\ell/k_\ell \in \{5, 10, 15, 20, 25\}\%$, $P \leq 4$ passes, $\beta = 0.25$, best (k_ℓ, r_ℓ, P) per n; exact: $r_\ell/k_\ell \in \{10, 20, 25\}\%$, $k_\ell \geq 64$. RF: identity filters. Ridge: on the 3,072 pixels. GD: same widths, full batch, MSE on ± 1 labels, no momentum or weight decay, 3,000 steps, rates $\{0.01, 0.03, 0.1, 0.2\}$; curve at rate 0.1 (best on average over n) and its best recorded step, 5 seeds. *Right:* conv 3×3 with 4,096 channels (no filter), conv 3×3 with 4,096 channels (filter k_c), 4×4 max pooling, fully connected 5,000 (filter k_3), ReLU, 5 seeds. FwdLoFi: $k_c \in \{64, \ldots, 1{,}024\}$, $k_3 \in \{32, \ldots, 1{,}024\}$. BackLoFi ($1^{\mathrm{st}}$ ord): $k_3 \leq 512$, ranks $\mathrm{round}(f \min(k_c, k_3))$ and $\mathrm{round}(f k_c)$ at the two boundaries, $f \in \{5, 10, 15, 20, 25\}\%$, $P \leq 4$, $\beta = 4$; exact: $k_c, k_3 \in \{128, \ldots, 1{,}024\}$, $f \in \{10, 20, 25\}\%$, $P \leq 3$. RF: identity filters. GD: same widths, full batch, MSE on ± 1

labels, no momentum or weight decay, $5{,}000$ steps, bfloat16 autocast (a replica without it at $n = 500$ gives the same curve within the seed noise), rates $\{0.001, 0.003, 0.005\}$ (0.01 diverges at the third step for every n); 3 seeds for 0.003 and 0.005 at every n and for 0.001 up to $n = 2\text{k}$, one seed above; the curve uses, at each n, the best of the rates run with three seeds, selected on the seed mean, at its best recorded step (0.003 up to $n = 5\text{k}$, 0.005 above). Shaded bands are one standard error over seeds. *Backward signal:* compared on the grid of the exact arm, the first-order signal of (12) and the exact signal backpropagated through the corrected network give the same error on the fully connected network within the standard error at every n; on the convolutional network the first-order signal is lower by about 0.3 points for $2{,}000 \le n \le 10{,}000$ (0.4 at $n = 500$) and equal at $n = 50{,}000$. *Reference values at* $n = 50{,}000$*:* fully connected, RF 13.9%, FwdLoFi 12.3%, BackLoFi 11.1%, GD 9.5%; convolutional, RF 8.4%, FwdLoFi 7.8%, BackLoFi 6.7%, GD 5.7%.

Figure 4. Fully connected network and BackLoFi (1st ord) of Figure 3, with $r_\ell/k_\ell \in \{2, 3, 5, 7, 10, 12\}\%$, $P = 1, \dots, 10$, $\beta = 0.25$, $k_\ell \in \{32, \dots, 2{,}048\}$ (best per point), $n \in \{1\text{k}, 5\text{k}, 10\text{k}, 50\text{k}\}$, 10 seeds.

Figure 5. Fully connected network, first-order signal, $k_\ell = 512$, $n = 10\text{k}$, $r_\ell/k_\ell \in \{2, 5, 10\}\%$, 6 passes, $\beta = 0.25$; 3 seeds.

Figure 6. Fully connected network of Figure 3 with dense layers and the damped updates (98), first-order signal, $\alpha = 0.1$, $r_\ell/k_\ell \in \{5, 10, 25\}\%$ (5% not shown: it stays within 0.1 points of FwdLoFi for $n \le 1\text{k}$ and descends more slowly than 10% above), $P = 1, \dots, 50$, $\beta = 0.25$, $k_\ell \in \{32, \dots, 2{,}048\}$ (best per point), $n \in \{500, 1\text{k}, 5\text{k}, 10\text{k}, 50\text{k}\}$, 10 seeds. The forward scale τ entering $\mathbf{\Gamma}$ is measured with the forward-phase filters on the current features; both updates are applied at each boundary in turn, so a hidden layer between two boundaries is damped twice per pass; a layer whose pre-activation scale drifts by more than a factor 8 would be rescaled by a power of two (never triggered). Swap reference: the 4-pass arm of Figure 3 at its best r_ℓ for each n.

Figure 7. GD runs of Figure 3, rates $\{0.01, 0.03, 0.1\}$, $n \in \{1\text{k}, 5\text{k}, 10\text{k}, 50\text{k}\}$, 5 seeds.

Figure 8. Convolutional network of Figure 3 with trainable layers of the same widths, random initialization, full batch (gradient accumulated over the whole training set), MSE on ± 1 labels, no momentum or weight decay, $5{,}000$ steps, bfloat16 autocast; rates $\{0.001, 0.003, 0.005\}$ (0.01 diverges) at every n; seeds as in the caption.