A THEORY OF NOISY LINEAR RECURRENT NEURAL NETWORK LEARNING DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Biological neural networks are noisy: their dynamics involve difficult-to-predict fluctuations which are likely due not just to chaos, but also to a variety of low-level biophysical processes. Despite the omnipresence of intrinsic noise, its theoretical consequences are incompletely understood, especially in the context of learning. For example, does noise meaningfully shape representation learning, or is it better viewed as a nuisance factor whose presence or absence changes little about what solution is found and when? Here, we argue that intrinsic noise can significantly modulate representation learning in recurrent neural networks (RNNs), since it can strongly break solution-space degeneracies and in some cases control whether a non-degenerate solution exists at all. We show that this is true for an analytically tractable model of noisy RNN learning with a linear activation function and white noise inputs. We find that, in the so-called rich regime (where weights are initialized with small values), noise strongly modulates the directions in which the read-in, readout, and recurrent weights align. Moreover, we find that noise produces a kind of phase transition: for sufficiently high noise, a non-degenerate solution is not even optimal if parameter scales are limited (e.g., via regularization). These results extend recent analytic work on RNN representation learning and show that noise can act as a fundamental determinant of representations.

1 INTRODUCTION

Recurrent neural networks (RNNs) are extraordinarily successful models of the brain (Vogels et al., 2005; Mante et al., 2013; Rajan et al., 2016; Vyas et al., 2020), and the past two decades of theory and experiment have produced a working understanding of what they learn and how. Their loss landscapes are generally more complex than those of feedforward networks, in part because recurrence can produce exploding and vanishing gradients (Bengio et al., 1994; Pascanu et al., 2013); their solution space is generally highly degenerate (Huang et al., 2025); and they can be biased towards learning low-rank solutions (Schuessler et al., 2020). But a major feature of RNNs that remains poorly understood is the significance of *intrinsic noise*.

Noise is omnipresent in biological neural networks: spike trains and firing rate traces throughout the nervous system involve difficult-to-predict fluctuations (Faisal et al., 2008), which may be a manifestation of chaotic behavior (Sompolinsky et al., 1988; Crisanti & Sompolinsky, 2018) or low-level stochasticity associated with chemical reactions. Although from one perspective noise is a nuisance, from another it provides pressure for networks to find more robust solutions. Noise is routinely used in RNN training (Yang et al., 2019; Driscoll et al., 2024), in part to improve generalization, and there is evidence that noise can drive networks into flatter loss landscape minima (Lim et al., 2021).

From a representation-learning perspective, it may be helpful to view noise as having two potential consequences. First, noise can provide *geometric constraints* on solutions: weights must be oriented in certain directions (i.e., the less noisy directions) and not others, and their overall scales must be chosen to minimize noise. Second, the goal of minimizing noise may compete with that of achieving good task performance; if there is too much noise, one expects that the system may be driven to a trivial solution at which task performance is poor, but there is little or no noise.

Although RNN learning dynamics are difficult to study, an interesting toy model of them has emerged in recent work: *linear* RNNs, especially linear RNNs trained to perform leaky integration of a white noise stimulus (Proca et al., 2025; Bordelon et al., 2025). As Bordelon et al. (2025)

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.

show, when network parameters are initially small, learning has two phases: an *alignment phase*, in which parameter movement concentrates along a specific (initialization-dependent) direction of state space, followed by a *dynamics phase*, in which an eigenvalue 'escapes' from the bulk in order to solve the task.

In this work, we consider an extremely similar theoretical setup, whose primary difference is the presence of (generally anisotropic) intrinsic noise. We show that their theory must be substantially modified even if there is only a small amount of noise, since noise affects both the alignment and dynamics phases, and even whether a nontrivial solution exists at all. Throughout, we focus on characterizing learning in the *rich regime*, as opposed to the *lazy regime*, in part because the rich regime is thought to be more representative of learning in performant RNNs.

Our main contributions are as follows.

- We develop an analytically tractable theory of representation learning in linear RNNs with intrinsic noise. We show that **noise breaks degeneracies of the original objective**, which yields preferred state space directions and read-in/readout scales.
- We characterize how intrinsic noise modifies rich-regime learning dynamics. During the alignment phase, anisotropic noise changes the growth rates of different state-space modes, which **biases learning toward less noisy directions**. Noise also breaks the symmetry between the read-in and readout weights, and in the absence of regularization, drives the input weight scale to infinity and the output weight scale to zero.
- We show that **sufficiently strong noise can produce phase transitions beyond which only a trivial solution is learned**. This is true both when all parameters are trainable and when some parameters are frozen.

2 Mathematical formulation

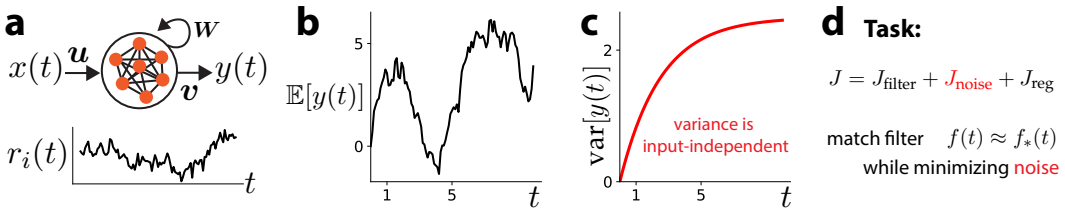

Figure 1: **Theoretical setting.** (**a**) We study a noisy linear RNN which receives a scalar white noise signal $x(t)$ as input and produces an output $y(t)$. The dynamics of individual r_i are noisy (bottom). (**b**) On average, the network filters its input. (**c**) Output variance is input-independent and eventually saturates (assuming the network is stable). (**d**) We assume the network is trained to optimize an objective with three terms: a filter-matching term that incentivizes matching its average behavior to that of a target filter $f_*(t)$, a noise reduction term, and a regularization term.

Our setup largely mirrors that of Bordelon et al. (2025). Throughout this paper, we study a linear RNN with $N \geq 1$ neurons whose instantaneous state is characterized by a vector $\boldsymbol{r} := (r_1, ..., r_N)^T \in \mathbb{R}^N$. We assume dynamics

$$\dot{\boldsymbol{r}} = \boldsymbol{A}\boldsymbol{r} + x(t)\boldsymbol{u} + \boldsymbol{G}\,\boldsymbol{\eta}(t) = -\gamma\boldsymbol{r} + \boldsymbol{W}\boldsymbol{r} + x(t)\boldsymbol{u} + \boldsymbol{G}\,\boldsymbol{\eta}(t) \tag{1}$$

where $x(t)$ is a time-dependent input signal, $\boldsymbol{u} \in \mathbb{R}^N$ are the *read-in weights*, $\gamma > 0$ is the coefficient of the *leak term*, and $\boldsymbol{W} \in \mathbb{R}^{N\times N}$ are the *recurrent weights* (Fig. 1a). To ease notation, we sometimes work with the combination $\boldsymbol{A} := -\gamma\boldsymbol{I} + \boldsymbol{W}$. The remaining term instantiates *intrinsic noise*: $\boldsymbol{G} \in \mathbb{R}^{N\times N}$ controls its amplitude, and $\boldsymbol{\eta}(t) \in \mathbb{R}^N$ is a vector of Gaussian white noise processes. We are most interested in understanding how the structure of $\boldsymbol{G}$ shapes learning.

Because Eq. 1 is linear in $\boldsymbol{r}$ and involves only state-independent additive noise, we can immediately write down the distribution of $\boldsymbol{r}(t)$ conditional on any initial condition, although we will assume $\boldsymbol{r}(0) = \boldsymbol{0}$ for simplicity. We find that $\boldsymbol{r}(t)$ is normally distributed with

$$p(\boldsymbol{r}, t|\boldsymbol{r}_0 = \boldsymbol{0}, 0) = \mathcal{N}\left(\int_0^t e^{\boldsymbol{A}t'}\boldsymbol{u}\,x(t-t')\,dt', \int_0^t e^{\boldsymbol{A}t'}\boldsymbol{S}e^{\boldsymbol{A}^T t'}\,dt'\right) \tag{2}$$

where $\boldsymbol{S} := \boldsymbol{G}\boldsymbol{G}^T$ is, up to a factor of 2, the *diffusion tensor* associated with $\boldsymbol{G}$ (which we take to be positive definite). On average, the network linearly filters its input (Fig. 1b). But because this integration is noisy, $\boldsymbol{S}$-dependent noise accumulates until (provided the eigenvalues of $\boldsymbol{A}$ all have negative real parts) it saturates (Fig. 1c).

2.1 Objective function

Following Bordelon et al. (2025), assume a linear readout determined by *readout weights* $\boldsymbol{v} \in \mathbb{R}^N$:

$$y(t) := \boldsymbol{v}^T \boldsymbol{r}(t) \, . \tag{3}$$

This readout defines a filter $f(t) := e^{-\gamma t}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{u}$ since, on average,

$$\mathbb{E}_{\boldsymbol{r}}[y(t)] = \int_0^t \boldsymbol{v}^T e^{\boldsymbol{A}t'}\boldsymbol{u}\; x(t-t')\; dt' = \int_0^t f(t')\; x(t-t')\; dt' \, . \tag{4}$$

Assume we train some combination of $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{W}$ so that $y(t)$ matches a *target output*

$$y_*(t) := \int_0^t e^{-c_* t'} x(t-t')\; dt' = \int_0^t f_*(t') x(t-t')\; dt' \tag{5}$$

which linearly filters its input $x(t)$ using the *target filter* $f_*(t) := e^{-c_* t}$. Here, $c_* > 0$ controls the time scale on which input is leaky-integrated. Assume we do this using the supervised objective

$$J := \lim_{T\to\infty} \frac{1}{T}\int_0^T \frac{1}{2}\mathbb{E}_{x,\boldsymbol{r}}\left\{\, [y(t) - y_*(t)]^2 \,\right\}\; dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2 \, , \tag{6}$$

which demands that the network output $y(t)$ agrees with $y_*(t)$ on average. Note that we have also included a regularization term whose strength is determined by $\kappa \geq 0$. This objective can be simplified if the input has *white noise* statistics (Proca et al., 2025; Bordelon et al., 2025), i.e., if $\mathbb{E}_x[x(t)] = 0$ and $\mathbb{E}_x[x(t)x(t-t')] = \delta(t-t')$. We find (see Appendix A)

$$\boxed{J(\boldsymbol{u},\boldsymbol{v},\boldsymbol{W}) = \int_0^\infty \frac{1}{2}\left[f(t) - f_*(t)\right]^2 + \frac{e^{-2\gamma t}}{2}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\; dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2} \, . \tag{7}$$

Note that J has three terms: (i) a *filter-matching term* J_{filter}, (ii) a *noise term* J_{noise}, and (iii) a *regularization term* J_{reg} (Fig. 1d). Bordelon et al. (2025) only considered the filter-matching term; we will see that the other two terms strongly modulate learning dynamics.

2.2 Learning dynamics

RNNs are normally trained via backpropagation through time (Werbos, 1990), FORCE (Sussillo & Abbott, 2009), or real-time recurrent learning (Williams & Zipser, 1989; Irie et al., 2024). One virtue of linear RNNs, at least for theory, is that gradients of J with respect to $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{W}$ can be computed exactly, so we can ignore the extra structure associated with those learning rules. Assuming parameters are updated via gradient descent, in the gradient-flow limit in which the learning rate is taken infinitesimally small, we recover ordinary differential equations (ODEs) (see Appendix A)

$$\begin{aligned}
\frac{d}{d\tau}\boldsymbol{u} &= \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}e^{\boldsymbol{W}^T t}\boldsymbol{v}\; dt - \kappa\boldsymbol{u} \\
\frac{d}{d\tau}\boldsymbol{v} &= \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}e^{\boldsymbol{W}t}\boldsymbol{u} - e^{-2\gamma t}e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\; dt - \kappa\boldsymbol{v} \\
\frac{d}{d\tau}\boldsymbol{W} &= \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}\int_0^t e^{\boldsymbol{W}^T t'}\boldsymbol{v}\boldsymbol{u}^T e^{\boldsymbol{W}^T(t-t')}\; dt' dt \\
&\quad - \int_0^\infty e^{-2\gamma t}\int_0^t e^{\boldsymbol{W}^T t'}\boldsymbol{v}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T(t-t')}\; dt' dt
\end{aligned} \tag{8}$$

parameterized by a real-valued 'learning time' $\tau \in [0, \infty)$. These dynamics are generically highly nonlinear and high-dimensional, although we will see shortly that they simplify substantially if the initial weights are assumed to be small.

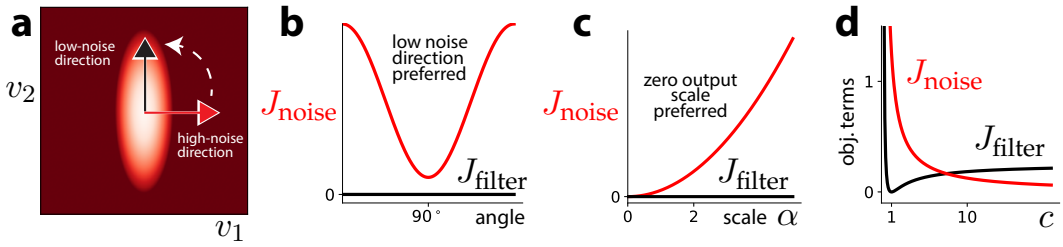

Figure 2: **Noise produces representational pressure.** (**a**) Two representations with the same filter but different readout vectors differ in noise level if $\boldsymbol{S}$ is anisotropic. Pictured is a schematic 2D example with a high-noise direction (horizontal) and a low-noise direction (vertical). One can reduce noise by rotating $\boldsymbol{v}$ (white arrow). (**b**) The noise term J_{noise} in the schematic example as a function of $\boldsymbol{v}$ orientation. J_{filter} is unaffected as long as the filter remains the same. (**c**) J_{noise} and J_{filter} as a function of α (see main text); a zero readout scale is preferred given noise. (**d**) J_{filter} and J_{noise} vs c (see main text) generally have different optima, and their competition can change where the minimum of J is, or whether it has a finite minimum at all.

3 Noise produces representational pressure

One elemental observation is that noise produces *representational pressure*: whereas the noiseless system has many representations compatible with good performance, the noisy system must minimize the corrupting influence of noise, which can greatly reduce the number of good solutions.

The deterministic solution is highly degenerate. In order for $f(t) = e^{-\gamma t}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{u}$ to match $f_*(t) = e^{-c_* t}$, assuming $\boldsymbol{W}$ is diagonalizable, it must be true that (i) $\gamma - c_*$ is an eigenvalue of $\boldsymbol{W}$; (ii) $\boldsymbol{u}$ has a component along one of the corresponding *right* eigenvectors; (iii) $\boldsymbol{v}$ has a component along one of the corresponding *left* eigenvectors; and (iv) $\boldsymbol{u}^T\boldsymbol{v} = 1$. But this yields a solution space with huge parameter degeneracies, including invariance to arbitrary invertible linear transformations. In particular, if $\boldsymbol{T} \in \mathbb{R}^{N\times N}$ is an invertible linear transformation, then $f(t)$ remains invariant to the mapping with $\boldsymbol{v} \to \boldsymbol{T}^T\boldsymbol{v}$, $\boldsymbol{u} \to \boldsymbol{T}^{-1}\boldsymbol{u}$, and $\boldsymbol{W} \to \boldsymbol{T}^{-1}\boldsymbol{W}\boldsymbol{T}$, since

$$\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{u} \xrightarrow{\boldsymbol{T}} \boldsymbol{v}^T\boldsymbol{T}\boldsymbol{T}^{-1}e^{\boldsymbol{W}t}\boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{u} = \boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{u}\,. \tag{9}$$

Interesting special cases include when $\boldsymbol{T}$ is a rotation or a change in read-in/readout scale. Meanwhile, the noise term in our objective does *not* generally respect those degeneracies:

$$\int_0^\infty e^{-2\gamma t}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\,dt \xrightarrow{\boldsymbol{T}} \int_0^\infty e^{-2\gamma t}\boldsymbol{v}^T e^{\boldsymbol{W}t}(\boldsymbol{T}\boldsymbol{S}\boldsymbol{T}^T)e^{\boldsymbol{W}^T t}\boldsymbol{v}\,dt\,. \tag{10}$$

We discuss various interesting consequences of this simple observation below.

Noise picks out preferred state-space directions. If multiple state-space directions could solve the task (i.e., if $\boldsymbol{W}$ has multiple correct eigenvalues), the presence of noise causes *less noisy* directions to be preferred (Fig. 2a, b). For example, if $\boldsymbol{W} = (\gamma - c_*)\boldsymbol{I}$ (i.e., all eigenvalues are correct) and $\boldsymbol{v}^T\boldsymbol{u} = 1$ (so that $f(t) = f_*(t)$), then the noise term is minimized when

$$\boldsymbol{v}_* = \frac{\boldsymbol{S}^{-1}\boldsymbol{u}}{\boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}}\,, \tag{11}$$

i.e., when state-space directions are read out according to how noisy they are, with $\boldsymbol{v}_*$ aligning more with less noisy directions. Essentially this solution is found when only the readout $\boldsymbol{v}$ is trained, i.e., when a 'reservoir computing' approach (Lukoševičius & Jaeger, 2009) is used (see Appendix B).

Noise produces preferred read-in and readout scales. A trivial way to reduce the noise term is to take the scale of the readout weights to zero. This is consistent with the above general observation; if $\boldsymbol{T} = \alpha\boldsymbol{I}$, the filter-matching term remains unchanged but the noise term picks up a factor of α^2, and the overall term is minimized when we take $\alpha \to 0$ (Fig. 2c). This implies, given no regularization ($\kappa = 0$), that it is optimal to take the readout scale to zero and the read-in scale to infinity.

Noise minimization can compete with filter-matching accuracy. The filter-matching term J_{filter} and the noise term J_{noise} generally do not share optima, so their relative size can determine where the fixed points of learning dynamics are, or whether there is any finite fixed point at all. Fig. 2d depicts an example we will consider later in Sec. 7: the balance of the two terms determines whether the recurrent weight scale c converges to a finite value near c_* or diverges to infinity.

4 Noise determines weight alignment in rich regime

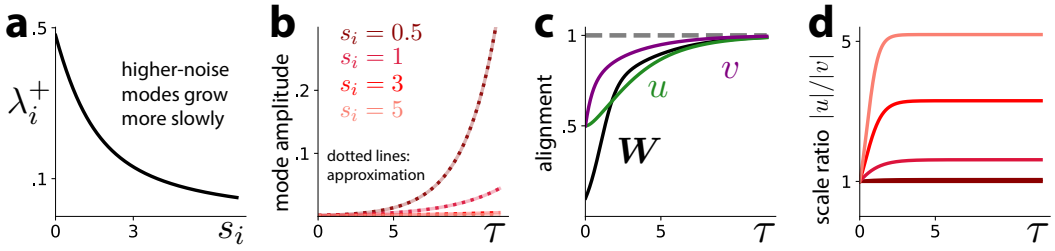

Figure 3: **Noise determines alignment.** (**a**) The dominant contribution to $u_i(\tau)$ and $v_i(\tau)$ grows exponentially quickly at a rate λ_i^+ that decreases as the corresponding $\boldsymbol{S}$ eigenvalue (i.e., noise level) s_i increases, so higher-noise modes grow more slowly. (**b**) Although dynamics involves both growing and decaying contributions, after enough learning time has elapsed, mode amplitudes (solid lines) are well-approximated by only their growing contribution (dashed lines). (**c**) The dot product between $\boldsymbol{u}$ and $\boldsymbol{v}$, and the least noisy eigenvector of $\boldsymbol{S}$, increases during this phase. $\boldsymbol{W}$ aligns in the same direction and becomes approximately rank-one. (**d**) The discrepancy between the scales of $\boldsymbol{u}$ and $\boldsymbol{v}$ grows with learning, and is higher in directions with larger noise levels.

In this section, we begin our analysis of this model's learning dynamics (Eq. 8). We focus on the so-called *rich regime*, in which weights are initialized to small values and low-rank structure emerges. Our analysis here follows the spirit of Bordelon et al. (2025) but differs in its details.

4.1 Dynamics given small initial weights

Assume $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{W}$ are initialized with small values (say, of order $\epsilon > 0$). To leading order in ϵ, gradient descent learning dynamics become (see Appendix A)

$$\frac{d}{d\tau}\boldsymbol{u} \approx \frac{\boldsymbol{v}}{c_* + \gamma} - \kappa \boldsymbol{u} \qquad \frac{d}{d\tau}\boldsymbol{v} \approx \frac{\boldsymbol{u}}{c_* + \gamma} - \frac{\boldsymbol{S}\boldsymbol{v}}{2\gamma} - \kappa \boldsymbol{v} \qquad \frac{d}{d\tau}\boldsymbol{W} \approx \frac{\boldsymbol{v}\boldsymbol{u}^T}{(c_* + \gamma)^2} - \frac{\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{S}}{4\gamma^2} \,. \quad (12)$$

In other words, $\boldsymbol{u}$ and $\boldsymbol{v}$ tend to evolve according to a certain linear system, while $\boldsymbol{W}$ depends only on input from $\boldsymbol{u}$ and $\boldsymbol{v}$. This is a major improvement, since Eq. 8 is in general a highly nonlinear system of ODEs. Note that the presence of $\boldsymbol{S}$ yields dynamics different from those studied by Bordelon et al. (2025); $\boldsymbol{S}$ breaks the symmetry between $\boldsymbol{u}$ and $\boldsymbol{v}$ dynamics, since components of $\boldsymbol{v}$ along high-noise eigenvectors of $\boldsymbol{S}$ are penalized while the analogous components of $\boldsymbol{u}$ are not.

4.2 Weights rotate away from highly noisy state space directions

The alignment-phase ODEs (Eq. 12) simplify if we use the eigenvectors of $\boldsymbol{S}$ as a basis. Letting u_i and v_i denote the components of $\boldsymbol{u}$ and $\boldsymbol{v}$ along an eigenvector whose eigenvalue is s_i, we have

$$\frac{d}{d\tau}u_i = -\kappa u_i + \frac{v_i}{c_* + \gamma} \qquad \frac{d}{d\tau}v_i = \frac{u_i}{c_* + \gamma} - \left(\kappa + \frac{s_i}{2\gamma}\right) v_i \,, \quad (13)$$

i.e., components along different eigenvectors evolve independently. The 2×2 matrix that governs these dynamics has two eigenvalues (see Appendix C):

$$\lambda_i^{\pm} := -\left(\kappa + \frac{s_i}{4\gamma}\right) \pm \sqrt{\frac{s_i^2}{16\gamma^2} + \frac{1}{(c_* + \gamma)^2}} \,. \quad (14)$$

If the regularization strength κ is too large, both eigenvalues are negative, and u_i and v_i collapse to a trivial zero solution. Otherwise, λ_i^+ is positive and λ_i^- is negative, which means that the λ_i^+ component of the solution tends to dominate after an initial transient. More quantitatively,

$$\begin{aligned} u_i(\tau) &\approx \frac{1}{\lambda_i^+ - \lambda_i^-}\left[\frac{v_i(0)}{c_* + \gamma} - u_i(0)(\kappa + \lambda_i^-)\right] e^{\lambda_i^+ \tau} \\ v_i(\tau) &\approx \frac{(c_* + \gamma)(\kappa + \lambda^+)}{\lambda_i^+ - \lambda_i^-}\left[\frac{v_i(0)}{c_* + \gamma} - u_i(0)(\kappa + \lambda_i^-)\right] e^{\lambda_i^+ \tau} . \end{aligned} \tag{15}$$

One interesting feature of this result is that the growth rate λ_i^+ is a *decreasing* function of s_i (Fig. 3a), which implies that components of $\boldsymbol{u}$ and $\boldsymbol{v}$ in noisier directions grow more slowly (Fig. 3b). Because small differences are exponentially amplified, to good approximation $\boldsymbol{u}$ and $\boldsymbol{v}$ align along the *least noisy* eigenvectors of $\boldsymbol{S}$ (Fig. 3c). In particular,

$$\begin{aligned} \boldsymbol{u}(\tau) &\approx u(\tau)\boldsymbol{P}\left[\frac{[\boldsymbol{u}(0) + \boldsymbol{v}(0)]}{2} + \frac{(c_* + \gamma)}{2}\left(\frac{s_{\min}}{4\gamma} + \Delta\right)\boldsymbol{u}(0)\right] \\ \boldsymbol{v}(\tau) &\approx v(\tau)\boldsymbol{P}\left[\frac{[\boldsymbol{u}(0) + \boldsymbol{v}(0)]}{2} + \frac{(c_* + \gamma)}{2}\left(\frac{s_{\min}}{4\gamma} + \Delta\right)\boldsymbol{u}(0)\right] \end{aligned} \tag{16}$$

where $\boldsymbol{P}$ is a projection matrix that projects onto the subspace with the smallest eigenvalue $s_{\min}$ (which may be one-dimensional), and where $\Delta := \sqrt{\frac{s_{\min}^2}{16\gamma^2} + \frac{1}{(c_*+\gamma)^2}} - \frac{1}{c_*+\gamma}$. Note that Eq. 12 implies that $\boldsymbol{W}$ aligns in the same direction as $\boldsymbol{u}$ and $\boldsymbol{v}$ (Fig. 3c). Another interesting feature of this result is that $u_i(\tau)$ generally does not equal $v_i(\tau)$ even if they are initialized to the same value—noise breaks the 'balancing' condition that can be assumed in the noise-free case (Domine et al., 2025; Bordelon et al., 2025). For large s_i, this discrepancy in scale can be quite large (Fig. 3d).

4.3 The outcome of alignment

To very good approximation, we can assume $\boldsymbol{u}$ and $\boldsymbol{v}$ align and that $\boldsymbol{W}$ is a rank-one matrix supported on the same vector. Letting $\boldsymbol{u}_0$ denote that (normalized) vector,

$$\boxed{\boldsymbol{u}(\tau) \approx u(\tau)\boldsymbol{u}_0 \qquad \boldsymbol{v}(\tau) \approx v(\tau)\boldsymbol{u}_0 \qquad \boldsymbol{W}(\tau) \approx [\gamma - c(\tau)]\boldsymbol{u}_0\boldsymbol{u}_0^T} \tag{17}$$

for some training-time-dependent coefficients $u(\tau)$, $v(\tau)$, and $c(\tau)$. Because gradient-flow dynamics does not change parameter directions under this ansatz, to characterize the next phase it suffices to analyze the dynamics of the scalars $u(\tau)$, $v(\tau)$, and $c(\tau)$. In the next few sections, we will use σ^2 to denote the noise level associated with the alignment direction (i.e., to denote the smallest eigenvalue of $\boldsymbol{S}$).

5 Rich regime learning dynamics

Alignment reduces an $N(N+2)$-dimensional problem to a three-dimensional one. Post-alignment learning dynamics correspond to gradient descent on an effective objective (see Appendix D)

$$J = \frac{1}{4c(\tau)}\left[u(\tau)v(\tau) - \frac{2c(\tau)}{c_* + c(\tau)}\right]^2 + \left(\frac{1}{4c_*} - \frac{c(\tau)}{[c_* + c(\tau)]^2}\right) + \frac{\sigma^2 v(\tau)^2}{4c(\tau)} + \frac{\kappa}{2}[u(\tau)^2 + v(\tau)^2] \tag{18}$$

where we will refer to $\sigma^2 > 0$ as the "noise level". Learning dynamics in this regime follow

$$\begin{aligned} \frac{d}{d\tau}u(\tau) &= \frac{v(\tau)}{c_* + c(\tau)} - \frac{u(\tau)v(\tau)^2}{2c(\tau)} - \kappa u(\tau) \\ \frac{d}{d\tau}v(\tau) &= \frac{u(\tau)}{c_* + c(\tau)} - \frac{v(\tau)[\sigma^2 + u(\tau)^2]}{2c(\tau)} - \kappa v(\tau) \\ \frac{d}{d\tau}c(\tau) &= \frac{v(\tau)^2[u(\tau)^2 + \sigma^2]}{4c(\tau)^2} - \frac{u(\tau)v(\tau)}{[c_* + c(\tau)]^2} . \end{aligned} \tag{19}$$

In the remainder of this section, we will assume no regularization ($\kappa = 0$). Let $N := uv$ denote the filter amplitude, so that $f(t) = Ne^{-ct}$. J is obviously minimized when $N = 1$ and $c = c_*$, since

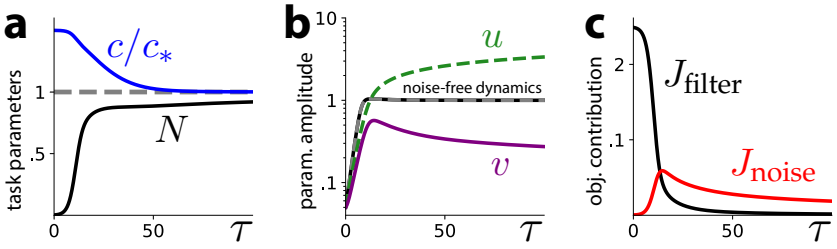

Figure 4: **Rich regime learning dynamics.** For plotting, we take $c_* = 1$, $\sigma^2 = 1$, and $\kappa = 0$. (**a**) In the absence of regularization, learning dynamics drive c toward its target, c_*, and the amplitude $N := uv$ toward 1. (**b**) Unlike the noise-free ($\sigma^2 = 0$) dynamics, noisy dynamics produce asymmetric trajectories for u and v, even if they are initialized to the same value. Asymptotically, $uv \approx 1$, u continues to increase, and v continues to decrease. (**c**) The filter-matching term and noise term are both driven to zero, although in this case the latter occurs on a slower time scale.

these choices set the first two terms to zero. Learning dynamics drive N and c toward those values (Fig. 4a).

A key difference between the noisy and noise-free ($\sigma^2 = 0$) dynamics is that noise breaks the symmetry between u and v. In the noise-free case, if u and v are initialized to the same value—i.e., if one makes a *balancing* assumption (Domine et al., 2025; Bordelon et al., 2025)—$u = v$ holds throughout training. In the noisy case, if there is no regularization, u continues to grow while v continues to shrink (Fig. 4b). In fact, Eq. 19 implies that

$$\frac{d}{d\tau}[u(\tau)^2 - v(\tau)^2] = \frac{\sigma^2}{c(\tau)} v(\tau)^2 > 0\,, \tag{20}$$

which means $u \to \infty$ and $v \to 0$. This makes sense, since taking $N = uv = 1$ and $v \to 0$ is precisely the combination that minimizes both the filter-matching term and the noise term (Fig. 4c).

6 A NOISE-INDUCED PHASE TRANSITION IN RICH REGIME DYNAMICS

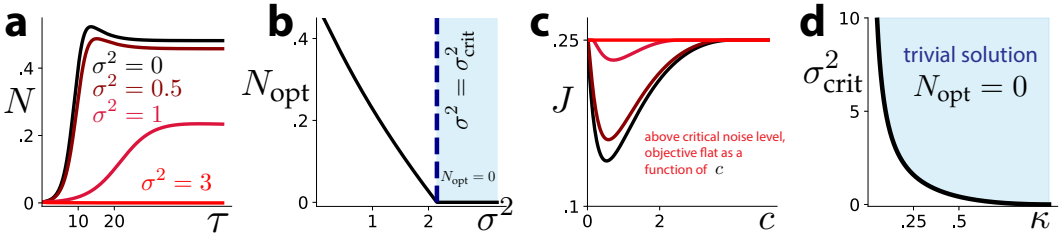

Figure 5: **Excessive noise causes amplitude collapse.** For plotting, we take $c_* = 1$ and $\kappa = 0.2$. (**a**) Below a critical noise level σ^2_{crit}, learning dynamics drives the filter amplitude $N := uv$ toward a nontrivial fixed point. Above that noise level, N is driven to zero. (**b**) The optimal amplitude N_{opt} as a function of noise level. For $\sigma^2 \geq \sigma^2_{\text{crit}}$, the trivial solution with $f(t) = 0$ is optimal. (**c**) Below σ^2_{crit}, J is minimized at a specific c value. Above it, J is insensitive to c since $N = 0$. (**d**) The phase boundary $\sigma^2_{\text{crit}}(\kappa)$. The noise threshold separating trivial and nontrivial solutions depends on the regularization strength; it is higher when κ is small, and $\sigma^2_{\text{crit}} \to \infty$ as $\kappa \to 0$.

The behavior described in the previous section, aside from the asymmetry between u and v dynamics, is not substantially different from what Bordelon et al. (2025) observed in the noise-free case. However, the noisy case becomes significantly different when the regularization strength κ is taken to be nonzero. Without regularization, we saw that the system minimizes noise by taking $u \to \infty$ and $v \to 0$; however, if high u values are penalized, the system can become frustrated. If the noise level is sufficiently high, there is a *phase transition*, and a trivial solution—which minimizes noise at the cost of a collapse in filter-matching performance—becomes optimal.

Below a critical noise level σ^2_{crit} (which depends on the regularization strength κ), the filter amplitude $N := uv$ approaches a nonzero fixed point N_{opt}. When $\kappa = 0$, this equals one, but it decreases as κ increases, since the scales of u and v are penalized by the regularization term. Beyond the critical noise level, dynamics drive N to zero (Fig. 5a), and moreover J is minimized at $N_{\text{opt}} = 0$ (Fig. 5b).

One can show algebraically (see Appendix E) that, after minimizing J over u and v, it becomes

$$J(c) = \frac{1}{4c_*} - c\left[\frac{1}{c_* + c} - \kappa\sqrt{1 + \frac{\sigma^2}{2\kappa c}}\right]_+^2 . \tag{21}$$

where $[...]_+$ denotes rectification (i.e., a ReLU). This simplified objective only has a nontrivial solution if the bracketed quantity is greater than zero; otherwise, J becomes independent of c (Fig. 5c).

We can identify the corresponding phase boundary $\sigma^2_{\text{crit}}(\kappa)$ by determining the noise level at which the bracketed quantity and its derivative equal zero. We find (see Appendix E) that

$$\boxed{\sigma^2_{\text{crit}}(\kappa) = \frac{[1 - q(\kappa)^2]^2}{q(\kappa)}} \tag{22}$$

for $\kappa < 1/c_*$, and $\sigma^2_{\text{crit}} = 0$ for $\kappa \geq 1/c_*$. Here,

$$q(\kappa) := \frac{2}{\sqrt{3}} \sinh\left[\frac{1}{3}\text{arsinh}(3\sqrt{3}\kappa c_*)\right] \qquad \text{and} \qquad \text{arsinh}(x) := \log(x + \sqrt{x^2 + 1}) . \tag{23}$$

The phase boundary is plotted in Fig. 5d. It has a few features worth emphasizing. First, since $q(\kappa) \to 0$ as $\kappa \to 0$, σ^2_{crit} diverges to infinity in this limit; this is consistent with the observation that there is no phase transition without regularization. Second, for small κ, we have the comparatively simple expressions

$$q(\kappa) \approx 2c_*\kappa \qquad\qquad \sigma^2_{\text{crit}}(\kappa) \approx \frac{1}{2c_*\kappa} \tag{24}$$

which qualitatively match Fig. 5d. In summary, if there is too much noise, it is almost as if the system's learning dynamics seek an 'escape route': parameters are pushed into trivial regimes (here, the filter amplitude N is taken to zero) in order to minimize noise at the expense of task accuracy.

7 A SECOND PHASE TRANSITION WHEN SCALE CHANGES ARE IMPOSSIBLE

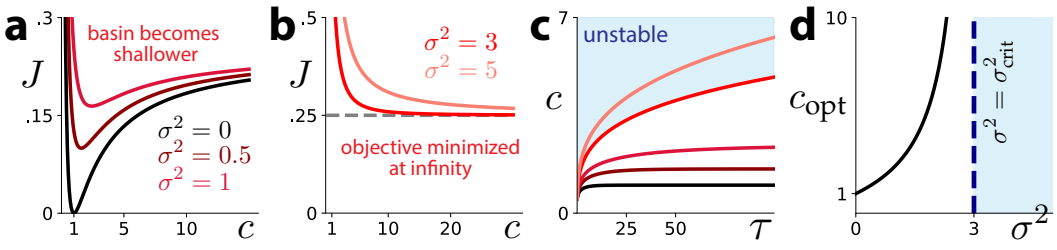

Figure 6: **Excessive noise produces an infinitely fast filter.** $c_* = 1$ assumed throughout. (**a**) When the read-in and readout scales are fixed, J is a function of c only, and for low enough noise it has a single attractor basin at $c = c_{\text{opt}}$. As σ^2 increases, this basin becomes increasingly shallow. (**b**) For $\sigma^2 \geq \sigma^2_{\text{crit}} = 3$, J is a decreasing function of c with no finite minimum. (**c**) Below the noise threshold, dynamics drive c toward the unique stable fixed point; above it, c grows without bound. (**d**) c_{opt} diverges like $1/(\sigma^2_{\text{crit}} - \sigma^2)$ just below the noise threshold.

We just saw how, for sufficiently large intrinsic noise $\sigma^2 > 0$, the system may converge to a trivial solution with zero amplitude, i.e., $uv = 0$. One may wonder whether this behavior is peculiar to our setup—for example, whether it depends on trainable read-in and readout weights. Interestingly, if u and v are frozen, a different type of noise-induced phase transition is possible.

Consider the $u = v = 1$ specialization of the simplified objective (Eq. 18):

$$J = \frac{1}{4c(\tau)}\left[1 - \frac{2c(\tau)}{c_* + c(\tau)}\right]^2 + \frac{1}{4c_*} - \frac{c(\tau)}{[c_* + c(\tau)]^2} + \frac{\sigma^2}{4c(\tau)} = \frac{1+\sigma^2}{4c(\tau)} - \frac{1}{c_* + c(\tau)} + \frac{1}{4c_*} \quad (25)$$

where we have removed the regularization term because it is constant. The corresponding learning dynamics are (see Appendix F)

$$\frac{d}{d\tau}c = \frac{[\sqrt{1+\sigma^2}(c_* + c) + 2c]}{4c^2(c_* + c)^2}[\sqrt{1+\sigma^2}(c_* + c) - 2c] \quad (26)$$

For sufficiently small noise, these dynamics have a stable positive fixed point at

$$c_{\text{opt}} = \frac{\sqrt{1+\sigma^2}\, c_*}{2 - \sqrt{1+\sigma^2}}\,. \quad (27)$$

As one might expect, c_{opt} equals the target rate c_* when there is no noise. However, as σ^2 increases, the corresponding basin of J becomes increasingly shallow (Fig. 6a). Pushing the noise level higher than $\sigma^2_{\text{crit}} = 3$ causes this basin to disappear (Fig. 6b), and for J to be minimized at $c = \infty$. Indeed, Eq. 26 has $dc/d\tau > 0$ for all c if $\sigma^2 \geq 3$, so learning drives c toward infinity (Fig. 6c). Just below σ^2_{crit}, the optimal c value diverges as $1/(\sigma^2_{\text{crit}} - \sigma^2)$ (Fig. 6d). We conclude that, although the precise representational 'escape route' may vary, the system may be driven toward trivial solutions that minimize noise when σ^2 is excessively large.

8 Discussion

We developed a theory of noisy linear RNN learning dynamics and mathematically showed that noise can decisively shape representation learning. Consistent with Bordelon et al. (2025), in the 'rich' regime, where parameters are initialized to small values, a low-rank task (here, matching a single leaky integration time scale) yields an approximately low-rank solution. But our findings differ from those of Bordelon et al. (2025) in three important ways. First, *noise strongly modulates the initial alignment phase*, and $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{W}$ preferentially align along the least noisy eigenvector (or eigenvectors) of $\boldsymbol{S}$, rather than simply in the average direction of the initial $\boldsymbol{u}$ and $\boldsymbol{v}$; this is interesting in part because it makes the network's solution depend much less on its initial condition.

Second, *noise produces asymmetry between the read-in and readout weights*, and in the second phase u tends to become large while v tends to become small. This is true even if u and v are initialized to identical values because noise breaks the *balancing* property that holds in the noise-free case. Finally, *noise produces multiple types of phase transitions*, because minimizing noise can compete with achieving good task performance. If $\boldsymbol{u}$ and $\boldsymbol{v}$ are trainable, we find that excessive noise drives the network toward a solution whose filter amplitude $N := uv$ equals zero; interestingly, this only occurs if regularization constrains the scales of u and v. If only $\boldsymbol{W}$ is trainable, excessive noise drives the system to $f(t) = 0$ in a different way, by making the filter's frequency infinitely fast.

In other machine learning settings, noise is known to shape what types of representations systems learn. Noise in optimization due to using stochastic rather than full-batch gradient descent tends to produce convergence to broader loss landscape minima that generalize better (Smith & Le, 2018; Smith et al., 2020). Noise in diffusion generative modeling may similarly improve generalization, albeit potentially through a different mechanism (Vastola, 2025). Our work suggests that noise shapes RNN representation learning, but does not necessarily produce *better* solutions—only *different* ones. Here, noise reduces solution-space degeneracy by constraining the direction of weight alignment and the relative scales of the read-in and readout weights. If there is too much noise, it is clear that noise has deleterious effects, since it drives the system toward trivial solutions.

Limitations and future work. For tractability's sake, we made a large number of simplifying assumptions: we assumed a *whitened* scalar input; a *linear* network; a simple filter-matching task involving learning *one* time scale; and near-zero parameter initializations that yield low-rank learning dynamics. Although the work we build on makes similar assumptions, it may be possible to relax some of them. For example, it may be possible to study nonlinear networks by linearizing them around a fixed point (either in state space or parameter space), or by exploiting dynamical mean-field theory machinery (Bordelon & Pehlevan, 2023).

AI USE STATEMENT

In this work, we used generative AI tools for brainstorming, literature search, to assist with checking the mathematical correctness of our results, and to assist with code generation. The authors have checked all text and results and take full responsibility for them.

REFERENCES

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. doi: 10.1109/72.279181.

Blake Bordelon and Cengiz Pehlevan. Self-consistent dynamical field theory of kernel evolution in wide neural networks*. *Journal of Statistical Mechanics: Theory and Experiment*, 2023(11): 114009, November 2023. doi: 10.1088/1742-5468/ad01b0. URL https://doi.org/10.1088/1742-5468/ad01b0.

Blake Bordelon, Jordan Cotler, Cengiz Pehlevan, and Jacob A. Zavatone-Veth. Dynamics of learning to integrate in linear recurrent neural networks. *arXiv e-prints*, art. arXiv:2503.18754, March 2025. doi: 10.48550/arXiv.2503.18754.

A. Crisanti and H. Sompolinsky. Path integral approach to random neural networks. *Phys. Rev. E*, 98:062120, Dec 2018. doi: 10.1103/PhysRevE.98.062120. URL https://link.aps.org/doi/10.1103/PhysRevE.98.062120.

Clementine Domine, Nicolas Anguita, Alexandra M Proca, Lukas Braun, Daniel Kunin, Pedro Mediano, and Andrew Saxe. From Lazy to Rich: Exact Learning Dynamics in Deep Linear Networks. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 102485–102536, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/fde1a69a5b6e554b2f1f727197d2651d-Paper-Conference.pdf.

Laura N. Driscoll, Krishna Shenoy, and David Sussillo. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nature Neuroscience*, 27(7):1349–1363, July 2024. ISSN 1546-1726. doi: 10.1038/s41593-024-01668-6. URL https://doi.org/10.1038/s41593-024-01668-6.

A. Aldo Faisal, Luc P. J. Selen, and Daniel M. Wolpert. Noise in the nervous system. *Nature Reviews Neuroscience*, 9(4):292–303, April 2008. ISSN 1471-0048. doi: 10.1038/nrn2258. URL https://doi.org/10.1038/nrn2258.

Ann Huang, Satpreet Harcharan Singh, Flavio Martinelli, and Kanaka Rajan. Measuring and Controlling Solution Degeneracy across Task-Trained Recurrent Neural Networks. 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, Main Conference, pp. 105259–105304. Curran Associates, Inc., 2025. doi: 10.52202/085713-3517. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/979d5a0ecc2cc3653928c5b793fc416b-Paper-Conference.pdf.

Kazuki Irie, Anand Gopalakrishnan, and Jürgen Schmidhuber. Exploring the Promise and Limits of Real-Time Recurrent Learning. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 27575–27598, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/74aec30590e07dbe2e29879f9df14fb2-Paper-Conference.pdf.

Soon Hoe Lim, N. Benjamin Erichson, Liam Hodgkinson, and Michael Mahoney. Noisy Recurrent Neural Networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 5124–5137. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/29301521774ff3cbd26652b2d5c95996-Paper.pdf.

Mantas Lukoševičius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009. ISSN 1574-0137. doi: https://doi.org/10.1016/j.cosrev.2009.03.005. URL https://www.sciencedirect.com/science/article/pii/S1574013709000173.

Valerio Mante, David Sussillo, Krishna V. Shenoy, and William T. Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, November 2013. ISSN 1476-4687. doi: 10.1038/nature12742. URL https://doi.org/10.1038/nature12742.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In Sanjoy Dasgupta and David McAllester (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1310–1318, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL https://proceedings.mlr.press/v28/pascanu13.html.

Alexandra Maria Proca, Clémentine Carla Juliette Dominé, Murray Shanahan, and Pedro A. M. Mediano. Learning dynamics in linear recurrent neural networks. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 49860–49902. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/proca25a.html.

Kanaka Rajan, Christopher D. Harvey, and David W. Tank. Recurrent Network Models of Sequence Generation and Memory. *Neuron*, 90(1):128–142, April 2016. ISSN 0896-6273. doi: 10.1016/j.neuron.2016.02.009. URL https://doi.org/10.1016/j.neuron.2016.02.009.

Friedrich Schuessler, Francesca Mastrogiuseppe, Alexis Dubreuil, Srdjan Ostojic, and Omri Barak. The interplay between randomness and structure during learning in RNNs. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 13352–13362. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/9ac1382fd8fc4b631594aa135d16ad75-Paper.pdf.

Samuel Smith, Erich Elsen, and Soham De. On the generalization benefit of noise in stochastic gradient descent. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9058–9067. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/smith20a.html.

Samuel L. Smith and Quoc V. Le. A bayesian perspective on generalization and stochastic gradient descent. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJij4yg0Z.

H. Sompolinsky, A. Crisanti, and H. J. Sommers. Chaos in random neural networks. *Phys. Rev. Lett.*, 61:259–262, Jul 1988. doi: 10.1103/PhysRevLett.61.259. URL https://link.aps.org/doi/10.1103/PhysRevLett.61.259.

David Sussillo and L.F. Abbott. Generating Coherent Patterns of Activity from Chaotic Neural Networks. *Neuron*, 63(4):544–557, August 2009. ISSN 0896-6273. doi: 10.1016/j.neuron.2009.07.018. URL https://doi.org/10.1016/j.neuron.2009.07.018.

John Vastola. Generalization through variance: how noise shapes inductive biases in diffusion models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 34922–34960, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/564ccf9de3a2a959d3d338889bad9d88-Paper-Conference.pdf.

Tim P. Vogels, Kanaka Rajan, and L.F. Abbott. Neural network dynamics. *Annual Review of Neuroscience*, 28(Volume 28, 2005):357–376, 2005. ISSN 1545-4126. doi: https://doi.org/10.1146/annurev.neuro.28.061604.135637. URL https://www.annualreviews.org/content/journals/10.1146/annurev.neuro.28.061604.135637.

Saurabh Vyas, Matthew D. Golub, David Sussillo, and Krishna V. Shenoy. Computation Through Neural Population Dynamics. *Annual Review of Neuroscience*, 43 (Volume 43, 2020):249–275, 2020. ISSN 1545-4126. doi: https://doi.org/10.1146/annurev-neuro-092619-094115. URL https://www.annualreviews.org/content/journals/10.1146/annurev-neuro-092619-094115.

P.J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990. doi: 10.1109/5.58337.

Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989. doi: 10.1162/neco.1989.1.2.270.

Guangyu Robert Yang, Madhura R. Joglekar, H. Francis Song, William T. Newsome, and Xiao-Jing Wang. Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*, 22(2):297–306, February 2019. ISSN 1546-1726. doi: 10.1038/s41593-018-0310-2. URL https://doi.org/10.1038/s41593-018-0310-2.

A OBJECTIVE FUNCTION AND DERIVATION OF LEARNING DYNAMICS

In this appendix, we simplify our original objective function and derive expressions for the associated gradient descent learning dynamics (in the gradient-flow limit that yields ODEs rather than a discrete-time recurrence).

A.1 SETUP

Recall from Sec. 2 that the objective function has the form

$$\boxed{J := \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2}\mathbb{E}_{x,\boldsymbol{r}} \left\{ \, [y(t) - y_*(t)]^2 \, \right\} \, dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2} \tag{28}$$

where $y(t) = \boldsymbol{v}^T \boldsymbol{r}(t)$ is the RNN's output and

$$y_*(t) := \int_0^t e^{-c_* t'} x(t-t') \, dt' = \int_0^t f_*(t') x(t-t') \, dt' \tag{29}$$

is the target output. If we assume zero initial activity (i.e., $\boldsymbol{r}(0) = \boldsymbol{0}$), then as we mention in Sec. 2,

$$p(\boldsymbol{r}, t | \boldsymbol{r}_0 = \boldsymbol{0}, 0) = \mathcal{N}\left(\int_0^t e^{\boldsymbol{A}t'} \boldsymbol{u} \, x(t-t') \, dt', \int_0^t e^{\boldsymbol{A}t'} \boldsymbol{S} e^{\boldsymbol{A}^T t'} \, dt' \right) . \tag{30}$$

This implies that the output $y(t)$ has statistics

$$\begin{aligned} \mathbb{E}_{\boldsymbol{r}}[y(t)] &= \int_0^t \boldsymbol{v}^T e^{\boldsymbol{A}t'} \boldsymbol{u} \, x(t-t') \, dt' = \int_0^t f(t') \, x(t-t') \, dt' \\ \text{var}_{\boldsymbol{r}}[y(t)] &= \boldsymbol{v}^T \boldsymbol{\Sigma} \boldsymbol{v} \end{aligned} \tag{31}$$

where $f(t) := e^{-\gamma t} \boldsymbol{v}^T e^{\boldsymbol{W}t} \boldsymbol{u}$ is the learned filter and the covariance $\boldsymbol{\Sigma}$ is defined as

$$\boldsymbol{\Sigma} := \int_0^t e^{\boldsymbol{A}t'} \boldsymbol{S} e^{\boldsymbol{A}^T t'} \, dt' . \tag{32}$$

Also recall that the input signal $x(t)$ is assumed to have white noise statistics, i.e., $\mathbb{E}_x[x(t)] = 0$ and $\mathbb{E}_x[x(t)x(t-t')] = \delta(t-t')$.

A.2 SIMPLIFYING THE OBJECTIVE

Decomposing into a bias and variance term. First, average over $\boldsymbol{r}$:

$$\begin{aligned} J &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2}\mathbb{E}_x \left\{ \, [\mathbb{E}_{\boldsymbol{r}}[y(t)] - y_*(t)]^2 + \text{var}_{\boldsymbol{r}}[y(t)] \, \right\} \, dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2 \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2}\mathbb{E}_x \left\{ \, [\mathbb{E}_{\boldsymbol{r}}[y(t)] - y_*(t)]^2 \, \right\} + \frac{1}{2}\text{var}_{\boldsymbol{r}}[y(t)] \, dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2 \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2}\mathbb{E}_x \left\{ \, [\mathbb{E}_{\boldsymbol{r}}[y(t)] - y_*(t)]^2 \, \right\} + \frac{1}{2}\boldsymbol{v}^T \boldsymbol{\Sigma}(t) \boldsymbol{v} \, dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2 \end{aligned} \tag{33}$$

where the second equality uses the fact that variance in $y(t)$ is input-independent.

Simplifying the mean term. The second term is input-independent; it represents noise due to recurrent dynamics. The first term can be written in terms of filters:

$$\begin{aligned} &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2}\mathbb{E}_x \left\{ \int_0^t \int_0^t [f(s) - f_*(s)][f(s') - f_*(s')] \, x(t-s)x(t-s') \right\} \, ds ds' dt \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2} \int_0^t \int_0^t [f(s) - f_*(s)][f(s') - f_*(s')] \, \mathbb{E}_x[x(t-s)x(t-s')] \, ds ds' dt \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2} \int_0^t \int_0^t [f(s) - f_*(s)][f(s') - f_*(s')] \, \delta(s-s') \, ds ds' dt \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \frac{1}{2} \int_0^t [f(s) - f_*(s)]^2 \, ds dt . \end{aligned} \tag{34}$$

This can be simplified with a clever rearrangement. Note,

$$\begin{aligned}\lim_{T\to\infty}\frac{1}{2T}\int_0^T\int_0^t [f(s)-f_*(s)]^2\,dsdt &= \lim_{T\to\infty}\frac{1}{2T}\int_0^T\int_s^T [f(s)-f_*(s)]^2\,dtds \\ &= \lim_{T\to\infty}\frac{1}{2}\int_0^T\left(1-\frac{s}{T}\right)[f(s)-f_*(s)]^2\,ds\,.\end{aligned} \tag{35}$$

If the eigenvalues of $\boldsymbol{A}$ have negative real parts (i.e., if dynamics is stable), then we can drop the s/T term, since it vanishes in the $T\to\infty$ limit. We conclude that

$$J_{\text{filter}} = \frac{1}{2}\int_0^\infty [f(t)-f_*(t)]^2\,dt\,. \tag{36}$$

Simplifying the variance term. The noise term can be rewritten as

$$\begin{aligned}&\lim_{T\to\infty}\frac{1}{2T}\int_0^T \boldsymbol{v}^T\boldsymbol{\Sigma}(t)\boldsymbol{v}\,dt \\ &= \boldsymbol{v}^T\left\{\lim_{T\to\infty}\frac{1}{2T}\int_0^T\int_0^t e^{\boldsymbol{A}s}\boldsymbol{S}e^{\boldsymbol{A}^Ts}\,dsdt\right\}\boldsymbol{v} \\ &= \boldsymbol{v}^T\left\{\lim_{T\to\infty}\frac{1}{2T}\int_0^T\int_s^T e^{\boldsymbol{A}s}\boldsymbol{S}e^{\boldsymbol{A}^Ts}\,dtds\right\}\boldsymbol{v} \\ &= \boldsymbol{v}^T\left\{\lim_{T\to\infty}\frac{1}{2}\int_0^T\left(1-\frac{s}{T}\right)e^{\boldsymbol{A}s}\boldsymbol{S}e^{\boldsymbol{A}^Ts}\,ds\right\}\boldsymbol{v}\,.\end{aligned} \tag{37}$$

Once again, if dynamics are stable, we can drop the s/T term in the infinite T limit. We obtain

$$J_{\text{noise}} = \frac{1}{2}\int_0^\infty \boldsymbol{v}^T e^{\boldsymbol{A}t}\boldsymbol{S}e^{\boldsymbol{A}^Tt}\boldsymbol{v}\,dt\,. \tag{38}$$

Total objective. Adding all of the terms, we obtain the simplified objective

$$\boxed{J(\boldsymbol{u},\boldsymbol{v},\boldsymbol{W}) = \int_0^\infty \frac{1}{2}[f(t)-f_*(t)]^2 + \frac{e^{-2\gamma t}}{2}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^Tt}\boldsymbol{v}\,dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2}\,. \tag{39}$$

A.3 Useful derivatives

A few derivatives are necessary to compute the gradients of J. First, the derivative of the matrix exponential with respect to $\boldsymbol{W}$ has components

$$\frac{\partial}{\partial W_{ij}}e^{\boldsymbol{W}t} = \int_0^t dt'\, e^{\boldsymbol{W}t'}\boldsymbol{e}_i\boldsymbol{e}_j^T e^{\boldsymbol{W}(t-t')}\,. \tag{40}$$

Similarly, the derivative of its transpose is

$$\frac{\partial}{\partial W_{ij}}e^{\boldsymbol{W}^Tt} = \int_0^t dt'\, e^{\boldsymbol{W}^Tt'}\boldsymbol{e}_j\boldsymbol{e}_i^T e^{\boldsymbol{W}^T(t-t')}\,. \tag{41}$$

The derivative of the learned filter $f(t)$ with respect to $\boldsymbol{W}$ has components

$$\frac{\partial}{\partial W_{ij}}f(t) = e^{-\gamma t}\int_0^t dt'\,\left[e^{\boldsymbol{W}^Tt'}\boldsymbol{v}\right]_i\left[e^{\boldsymbol{W}(t-t')}\boldsymbol{u}\right]_j = e^{-\gamma t}\int_0^t dt'\, e^{\boldsymbol{W}^Tt'}\boldsymbol{v}\boldsymbol{u}^T e^{\boldsymbol{W}^T(t-t')}\,. \tag{42}$$

Finally, the derivative of the variance term with respect to $\boldsymbol{W}$ has components

$$\begin{aligned}&\frac{\partial}{\partial W_{ij}}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^Tt}\boldsymbol{v} \\ &= \int_0^t dt'\,\boldsymbol{v}^T e^{\boldsymbol{W}t'}\boldsymbol{e}_i\boldsymbol{e}_j^T e^{\boldsymbol{W}(t-t')}\boldsymbol{S}e^{\boldsymbol{W}^Tt}\boldsymbol{v} + \int_0^t dt'\,\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^Tt'}\boldsymbol{e}_j\boldsymbol{e}_i^T e^{\boldsymbol{W}^T(t-t')}\boldsymbol{v} \\ &= \int_0^t dt'\, e^{\boldsymbol{W}^Tt'}\boldsymbol{v}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T(t-t')} + \int_0^t dt'\, e^{\boldsymbol{W}^T(t-t')}\boldsymbol{v}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^Tt'}\,.\end{aligned} \tag{43}$$

A.4 Derivation of learning dynamics

Using the aforementioned derivatives, we can immediately compute the relevant gradients of J:

$$\begin{aligned}
\frac{\partial J}{\partial \boldsymbol{u}} &= \int_0^\infty [f(t) - f_*(t)] \frac{\partial f(t)}{\partial \boldsymbol{u}} dt + \kappa \boldsymbol{u} \\
&= \int_0^\infty [f(t) - f_*(t)] e^{-\gamma t} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt + \kappa \boldsymbol{u} \\
\frac{\partial J}{\partial \boldsymbol{v}} &= \int_0^\infty [f(t) - f_*(t)] \frac{\partial f(t)}{\partial \boldsymbol{v}} + e^{-2\gamma t} e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt + \kappa \boldsymbol{v} \\
&= \int_0^\infty [f(t) - f_*(t)] e^{-\gamma t} e^{\boldsymbol{W} t} \boldsymbol{u} + e^{-2\gamma t} e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt + \kappa \boldsymbol{v} \\
\frac{\partial J}{\partial \boldsymbol{W}} &= \int_0^\infty [f(t) - f_*(t)] e^{-\gamma t} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{u}^T e^{\boldsymbol{W}^T (t - t')} \, dt' dt \\
&\quad + \int_0^\infty \frac{e^{-2\gamma t}}{2} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{v}^T e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T (t - t')} \, dt' dt + \int_0^\infty \frac{e^{-2\gamma t}}{2} \int_0^t e^{\boldsymbol{W}^T (t - t')} \boldsymbol{v} \boldsymbol{v}^T e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t'} \, dt' dt \, .
\end{aligned} \tag{44}$$

This result immediately implies that our gradient descent learning dynamics have the form

$$\begin{aligned}
\frac{d}{d\tau} \boldsymbol{u} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt - \kappa \boldsymbol{u} \\
\frac{d}{d\tau} \boldsymbol{v} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} e^{\boldsymbol{W} t} \boldsymbol{u} - e^{-2\gamma t} e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt - \kappa \boldsymbol{v} \\
\frac{d}{d\tau} \boldsymbol{W} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{u}^T e^{\boldsymbol{W}^T (t - t')} \, dt' dt \\
&\quad - \int_0^\infty \frac{e^{-2\gamma t}}{2} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{v}^T e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T (t - t')} \, dt' dt - \int_0^\infty \frac{e^{-2\gamma t}}{2} \int_0^t e^{\boldsymbol{W}^T (t - t')} \boldsymbol{v} \boldsymbol{v}^T e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t'} \, dt' dt \, .
\end{aligned} \tag{45}$$

By a change of variables $t' \to t - t'$, one can show that the two noise terms in the $\boldsymbol{W}$ dynamics are identical. This allows the final result to be written in a slightly more compact form:

$$\begin{aligned}
\frac{d}{d\tau} \boldsymbol{u} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt - \kappa \boldsymbol{u} \\
\frac{d}{d\tau} \boldsymbol{v} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} e^{\boldsymbol{W} t} \boldsymbol{u} - e^{-2\gamma t} e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T t} \boldsymbol{v} \, dt - \kappa \boldsymbol{v} \\
\frac{d}{d\tau} \boldsymbol{W} &= \int_0^\infty [f_*(t) - f(t)] e^{-\gamma t} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{u}^T e^{\boldsymbol{W}^T (t - t')} \, dt' dt \\
&\quad - \int_0^\infty e^{-2\gamma t} \int_0^t e^{\boldsymbol{W}^T t'} \boldsymbol{v} \boldsymbol{v}^T e^{\boldsymbol{W} t} \boldsymbol{S} e^{\boldsymbol{W}^T (t - t')} \, dt' dt \, .
\end{aligned} \tag{46}$$

A.5 Learning dynamics given small initial weights

We can expand these learning dynamics to leading order in $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{W}$ (supposing they are all of order $\epsilon > 0$, say). Doing so, we find that

$$\begin{aligned}
\frac{d}{d\tau} \boldsymbol{u} &\approx \int_0^\infty f_*(t) e^{-\gamma t} \boldsymbol{v} \, dt - \kappa \boldsymbol{u} \\
\frac{d}{d\tau} \boldsymbol{v} &\approx \int_0^\infty f_*(t) e^{-\gamma t} \boldsymbol{u} - e^{-2\gamma t} \boldsymbol{S} \boldsymbol{v} \, dt - \kappa \boldsymbol{v} \\
\frac{d}{d\tau} \boldsymbol{W} &\approx \int_0^\infty f_*(t) e^{-\gamma t} \int_0^t \boldsymbol{v} \boldsymbol{u}^T \, dt' dt - \int_0^\infty e^{-2\gamma t} \int_0^t \boldsymbol{v} \boldsymbol{v}^T \boldsymbol{S} \, dt' dt \, .
\end{aligned} \tag{47}$$

We can be more explicit using the fact that $f_*(t) = e^{-c_* t}$. Simplifying yields

$$\begin{aligned}
\frac{d}{d\tau}\boldsymbol{u} &\approx \frac{1}{c_* + \gamma}\boldsymbol{v} - \kappa\boldsymbol{u} \\
\frac{d}{d\tau}\boldsymbol{v} &\approx \frac{1}{c_* + \gamma}\boldsymbol{u} - \frac{1}{2\gamma}\boldsymbol{S}\boldsymbol{v} - \kappa\boldsymbol{v} \\
\frac{d}{d\tau}\boldsymbol{W} &\approx \int_0^\infty f_*(t) e^{-\gamma t} \int_0^t \boldsymbol{v}\boldsymbol{u}^T \, dt' dt - \int_0^\infty e^{-2\gamma t} \int_0^t \boldsymbol{v}\boldsymbol{v}^T \boldsymbol{S} \, dt' dt \\
&\approx \frac{1}{(c_* + \gamma)^2}\boldsymbol{v}\boldsymbol{u}^T - \frac{1}{4\gamma^2}\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{S} \, .
\end{aligned} \tag{48}$$

B RESERVOIR COMPUTING CASE

In this appendix, we work out the learning dynamics of the *reservoir computing* model in which only the readout $\boldsymbol{v}$ is trained. Its primary virtue is that it illustrates, even when learning dynamics are linear, how noise produces representational pressure. For simplicity, we ignore regularization and set $\kappa = 0$.

Recall from Appendix A that $\boldsymbol{v}$ has dynamics

$$\begin{aligned}\frac{d}{d\tau}\boldsymbol{v} &= \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}e^{\boldsymbol{W}t}\boldsymbol{u} - e^{-2\gamma t}e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\, dt \\ &= \int_0^\infty e^{-(c_*+\gamma)t}e^{\boldsymbol{W}t}\boldsymbol{u} - e^{-2\gamma t}e^{\boldsymbol{W}t}\boldsymbol{u}\boldsymbol{u}^T e^{\boldsymbol{W}^T t}\boldsymbol{v} - e^{-2\gamma t}e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\, dt \\ &= \left[\int_0^\infty e^{-(c_*+\gamma)t}e^{\boldsymbol{W}t}\, dt\right]\boldsymbol{u} - \left[\int_0^\infty e^{-2\gamma t}e^{\boldsymbol{W}t}(\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})e^{\boldsymbol{W}^T t}\, dt\right]\boldsymbol{v}\,.\end{aligned} \tag{49}$$

These dynamics are linear, and the matrix that controls which directions of $\boldsymbol{v}$ are learned first is

$$\boldsymbol{M} := \int_0^\infty e^{-2\gamma t}e^{\boldsymbol{W}t}(\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})e^{\boldsymbol{W}^T t}\, dt\,. \tag{50}$$

The fixed point of these dynamics is

$$\boxed{\boldsymbol{v}_* = \left(\int_0^\infty e^{-2\gamma t}e^{\boldsymbol{W}t}(\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})e^{\boldsymbol{W}^T t}\, dt\right)^{-1}\left(\int_0^\infty e^{-(c_*+\gamma)t}e^{\boldsymbol{W}t}\, dt\right)\boldsymbol{u}}\,. \tag{51}$$

Informative simplification. Assume for simplicity that $\boldsymbol{S}$ is diagonal and $\boldsymbol{W} = (\gamma - c_*)\boldsymbol{I}$, i.e., that all state-space directions are compatible with the target filter. In this case,

$$\begin{aligned}\int_0^\infty e^{-(c_*+\gamma)t}e^{\boldsymbol{W}t}\, dt &= \frac{1}{2c_*} \\ \int_0^\infty e^{-2\gamma t}e^{\boldsymbol{W}t}(\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})e^{\boldsymbol{W}^T t}\, dt &= \frac{1}{2c_*}(\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})\end{aligned} \tag{52}$$

so the optimal readout is

$$\boxed{\boldsymbol{v}_* = (\boldsymbol{u}\boldsymbol{u}^T + \boldsymbol{S})^{-1}\boldsymbol{u} = \left(\boldsymbol{S}^{-1} - \frac{\boldsymbol{S}^{-1}\boldsymbol{u}\boldsymbol{u}^T\boldsymbol{S}^{-1}}{1 + \boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}}\right)\boldsymbol{u} = \frac{\boldsymbol{S}^{-1}\boldsymbol{u}}{1 + \boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}}} \tag{53}$$

For a large read-in vector $\boldsymbol{u}$, this approximately equals

$$\boldsymbol{v}_* \approx \frac{\boldsymbol{S}^{-1}\boldsymbol{u}}{\boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}} \tag{54}$$

which weighs each input direction by how noisy it is. This prescription also yields the correct optimal filter, since

$$\boldsymbol{u}^T\boldsymbol{v}_* \approx \frac{\boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}}{\boldsymbol{u}^T\boldsymbol{S}^{-1}\boldsymbol{u}} = 1\,, \tag{55}$$

which implies

$$f(t) = \boldsymbol{v}_*^T e^{\boldsymbol{A}t}\boldsymbol{u} = e^{-c_* t}\boldsymbol{v}_*^T\boldsymbol{u} \approx e^{-c_* t}\,. \tag{56}$$

This approximate solution is intuitive since it is precisely the choice of $\boldsymbol{v}$ that, for this choice of $\boldsymbol{W}$ and $\boldsymbol{u}$, minimizes the quadratic noise term subject to the constraint that $\boldsymbol{u}^T\boldsymbol{v} = 1$.

C ALIGNMENT DYNAMICS

In this appendix, we solve the dynamics associated with the alignment phase (see Sec. 4). Using the small-parameter gradient descent updates from Appendix A, we have

$$\frac{d}{d\tau}\boldsymbol{u} \approx \frac{\boldsymbol{v}}{c_* + \gamma} - \kappa \boldsymbol{u} \qquad \frac{d}{d\tau}\boldsymbol{v} \approx \frac{\boldsymbol{u}}{c_* + \gamma} - \frac{\boldsymbol{S}\boldsymbol{v}}{2\gamma} - \kappa \boldsymbol{v} \qquad \frac{d}{d\tau}\boldsymbol{W} \approx \frac{\boldsymbol{v}\boldsymbol{u}^T}{(c_* + \gamma)^2} - \frac{\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{S}}{4\gamma^2} \,. \tag{57}$$

Because $\boldsymbol{u}$ and $\boldsymbol{v}$ dynamics form a closed system, we will begin by studying $\boldsymbol{u}$ and $\boldsymbol{v}$ only. Once we determine how $\boldsymbol{u}$ and $\boldsymbol{v}$ change with time, we will use them to determine how $\boldsymbol{W}$ changes.

C.1 SOLVING READ-IN AND READOUT DYNAMICS

Naively, $\boldsymbol{u}$ and $\boldsymbol{v}$ together determine a $2N$-dimensional system of ODEs, since

$$\begin{aligned} \frac{d}{d\tau}\boldsymbol{u} &= -\kappa \boldsymbol{u} + \frac{\boldsymbol{v}}{c_* + \gamma} \\ \frac{d}{d\tau}\boldsymbol{v} &= \frac{\boldsymbol{u}}{c_* + \gamma} - \left(\kappa + \frac{\boldsymbol{S}}{2\gamma}\right)\boldsymbol{v} \,. \end{aligned} \tag{58}$$

Note that, to ease notation, we will write equalities instead of $\approx$.

This system substantially simplifies if we work in a basis of eigenvectors of $\boldsymbol{S}$. Along the i-th such eigenvector, if $s_i > 0$ denotes the corresponding eigenvalue,

$$\begin{aligned} \frac{d}{d\tau}u_i &= -\kappa u_i + \frac{v_i}{c_* + \gamma} \\ \frac{d}{d\tau}v_i &= \frac{u_i}{c_* + \gamma} - \left(\kappa + \frac{s_i}{2\gamma}\right) v_i \,. \end{aligned} \tag{59}$$

Relevant eigenvalues and eigenvectors. This is just a two-dimensional system governed by the 2×2 matrix

$$\boldsymbol{M} := \begin{pmatrix} -\kappa & \frac{1}{c_* + \gamma} \\ \frac{1}{c_* + \gamma} & -\left(\kappa + \frac{s_i}{2\gamma}\right) \end{pmatrix} \tag{60}$$

whose characteristic equation is

$$(\lambda + \kappa)\left(\lambda + \kappa + \frac{s_i}{2\gamma}\right) - \frac{1}{(c_* + \gamma)^2} = 0 \,. \tag{61}$$

Equivalently,

$$\lambda^2 + 2\left[\kappa + \frac{s_i}{4\gamma}\right]\lambda + \left[\kappa^2 + \frac{\kappa s_i}{2\gamma} - \frac{1}{(c_* + \gamma)^2}\right] = 0 \,. \tag{62}$$

The solutions of this equation (i.e., the eigenvalues of $\boldsymbol{M}$) are

$$\begin{aligned} \lambda_i^{\pm} &= -\left(\kappa + \frac{s_i}{4\gamma}\right) \pm \sqrt{\left(\kappa + \frac{s_i}{4\gamma}\right)^2 + \frac{1}{(c_* + \gamma)^2} - \kappa^2 - \frac{\kappa s_i}{2\gamma}} \\ &= -\left(\kappa + \frac{s_i}{4\gamma}\right) \pm \sqrt{\frac{s_i^2}{16\gamma^2} + \frac{1}{(c_* + \gamma)^2}} \,. \end{aligned} \tag{63}$$

Note that λ_i^- is always negative, and so corresponds to decay. The other root may be positive or negative depending on the size of the regularization strength $\kappa \geq 0$. Beyond a critical κ value

$$\kappa_{\text{crit}} := \sqrt{\frac{s_{\text{min}}^2}{16\gamma^2} + \frac{1}{(c_* + \gamma)^2}} - \frac{s_{\text{min}}}{4\gamma} \,, \tag{64}$$

where s_{min} is the smallest of the s_i, all modes decay and one obtains only the trivial zero solution for all u_i and v_i. For $\kappa < \kappa_{\text{crit}}$, at least some modes survive.

Assume in what follows that $\kappa < \kappa_{\text{crit}}$. Then at least some λ_i^+ will be positive, which means that u_i and v_i along those directions have components which *grow* with time. In particular, since the eigenvectors of $\boldsymbol{M}$ are

$$\boldsymbol{q}^{\pm} := \begin{pmatrix} 1 \\ (c_* + \gamma)(\kappa + \lambda^{\pm}) \end{pmatrix}, \tag{65}$$

the solution $\boldsymbol{z}(\tau) := (u_i(\tau), v_i(\tau))^T$ equals

$$\boldsymbol{z} = e^{\boldsymbol{M}\tau}\boldsymbol{z}_0 = e^{\boldsymbol{M}\tau}(c_+\boldsymbol{q}^+ + c_-\boldsymbol{q}^-) = c_+\boldsymbol{q}^+ e^{\lambda_i^+\tau} + c_-\boldsymbol{q}^- e^{\lambda_-^i\tau} \tag{66}$$

for some coefficients c_+ and c_-.

Enforcing initial conditions. We can determine c_+ and c_- by choosing them so that $\boldsymbol{z}(0) = (u_i(0), v_i(0))^T$. This requires enforcing

$$\begin{aligned} u_i(0) &= c_+ + c_- \\ v_i(0) &= c_+(c_* + \gamma)(\kappa + \lambda^+) + c_-(c_* + \gamma)(\kappa + \lambda^-) \, . \end{aligned} \tag{67}$$

The first condition implies $c_- = u_i(0) - c_+$. The second implies

$$\begin{aligned} \frac{v_i(0)}{c_* + \gamma} &= c_+(\kappa + \lambda^+) + c_-(\kappa + \lambda^-) \\ &= c_+(\kappa + \lambda^+) + [u_i(0) - c_+](\kappa + \lambda^-) \\ &= u_i(0)(\kappa + \lambda^-) + c_+(\lambda^+ - \lambda^-) \end{aligned} \tag{68}$$

so

$$\boxed{c_+ = \frac{1}{\lambda_i^+ - \lambda_i^-}\left[\frac{v_i(0)}{c_* + \gamma} - u_i(0)(\kappa + \lambda_i^-)\right]}. \tag{69}$$

Because the mode which *grows* as opposed to *shrinks* will dominate after enough time has passed, to good approximation we have

$$\begin{aligned} u_i(\tau) &\approx e^{\lambda_i^+\tau}c_+ = \frac{1}{\lambda_i^+ - \lambda_i^-}\left[\frac{v_i(0)}{c_* + \gamma} - u_i(0)(\kappa + \lambda_i^-)\right] e^{\lambda_i^+\tau} \\ v_i(\tau) &\approx e^{\lambda_i^+\tau}c_+(c_* + \gamma)(\kappa + \lambda^+) = \frac{(c_* + \gamma)(\kappa + \lambda^+)}{\lambda_i^+ - \lambda_i^-}\left[\frac{v_i(0)}{c_* + \gamma} - u_i(0)(\kappa + \lambda_i^-)\right] e^{\lambda_i^+\tau} \end{aligned} \tag{70}$$

for moderate τ.

Note that λ_i^+ is a *decreasing* function of s_i. This implies that $\boldsymbol{u}$ and $\boldsymbol{v}$ will tend to get amplified more quickly along less noisy directions. After enough time has passed, the coefficients along the least noisy direction will tend to dwarf the others, so to good approximation we have

$$\begin{aligned} \boldsymbol{u}(\tau) &\approx \frac{u(\tau)}{2}\boldsymbol{P}\left[\frac{\boldsymbol{v}(0)}{c_* + \gamma} - \boldsymbol{u}(0)(\kappa + \lambda_{\text{min}}^-)\right] \\ \boldsymbol{v}(\tau) &\approx \frac{v(\tau)}{2}\boldsymbol{P}\left[\frac{\boldsymbol{v}(0)}{c_* + \gamma} - \boldsymbol{u}(0)(\kappa + \lambda_{\text{min}}^-)\right] \end{aligned} \tag{71}$$

after the alignment phase, where λ_{min}^- is the negative eigenvalue associated with the least noisy direction (i.e., the smallest s_i, which we can call s_{min}). Here, $\boldsymbol{P}$ is a matrix that projects into the least noisy subspace (which may be of arbitrary dimensionality) and $u(\tau)$ and $v(\tau)$ are mode amplitudes whose precise form is given above. Equivalently, $\boldsymbol{u}(\tau)$ and $\boldsymbol{v}(\tau)$ align in the *least noisy* state-space direction.

Define the quantity

$$\Delta := \sqrt{\frac{s_{\text{min}}^2}{16\gamma^2} + \frac{1}{(c_* + \gamma)^2}} - \frac{1}{c_* + \gamma} \, , \tag{72}$$

which is an increasing function of s_{min}. Δ quantifies the gap between the noise-free solution and the noisy solution. In terms of it, we can write our answer as

$$\begin{aligned} \boldsymbol{u}(\tau) &\approx u(\tau)\boldsymbol{P}\left[\frac{[\boldsymbol{u}(0) + \boldsymbol{v}(0)]}{2} + \frac{(c_* + \gamma)}{2}\left(\frac{s_{\text{min}}}{4\gamma} + \Delta(s_{\text{min}})\right)\boldsymbol{u}(0)\right] \\ \boldsymbol{v}(\tau) &\approx v(\tau)\boldsymbol{P}\left[\frac{[\boldsymbol{u}(0) + \boldsymbol{v}(0)]}{2} + \frac{(c_* + \gamma)}{2}\left(\frac{s_{\text{min}}}{4\gamma} + \Delta(s_{\text{min}})\right)\boldsymbol{u}(0)\right] \, . \end{aligned} \tag{73}$$

C.2 W DYNAMICS

Given $\boldsymbol{u}$ and $\boldsymbol{v}$ dynamics, $\boldsymbol{W}$ dynamics are extremely simple. Because $\boldsymbol{u}$ and $\boldsymbol{v}$ are, to good approximation, rank one, we can write

$$\boldsymbol{u}(\tau) \approx u(\tau)\boldsymbol{u}_0 \qquad\qquad \boldsymbol{v}(\tau) \approx v(\tau)\boldsymbol{u}_0 \tag{74}$$

for some vector $\boldsymbol{u}_0$ (which is an eigenvector of $\boldsymbol{S}$ with eigenvalue σ^2) and scalar amplitudes $u(\tau)$ and $v(\tau)$. Eq. 57 then implies that

$$\frac{d}{d\tau}\boldsymbol{W} \approx \left[\frac{u(\tau)v(\tau)}{(c_* + \gamma)^2} - \frac{\sigma^2 v(\tau)^2}{4\gamma^2}\right] \boldsymbol{u}_0\boldsymbol{u}_0^T \,, \tag{75}$$

or equivalently that

$$\boldsymbol{W}(\tau) \approx [\gamma - c(\tau)]\boldsymbol{u}_0\boldsymbol{u}_0^T \tag{76}$$

since these updates are all in the same direction. $c(\tau)$ is a scalar which determines the overall amplitude of $\boldsymbol{W}$. In other words, $\boldsymbol{W}$ becomes a rank-one matrix aligned in the same direction as $\boldsymbol{u}$ and $\boldsymbol{v}$.

D DERIVATION OF LOW-DIMENSIONAL EFFECTIVE DYNAMICS

In this appendix, we derive the effective learning dynamics associated with the post-alignment phase. Recall from Appendix C and Sec. 4 that the outcome of alignment has

$$\boxed{\boldsymbol{u}(\tau) \approx u(\tau)\boldsymbol{u}_0 \qquad \boldsymbol{v}(\tau) \approx v(\tau)\boldsymbol{u}_0 \qquad \boldsymbol{W}(\tau) \approx [\gamma - c(\tau)]\boldsymbol{u}_0\boldsymbol{u}_0^T} \tag{77}$$

for some (assumed normalized) $\boldsymbol{u}_0$ which is an eigenvector of $\boldsymbol{S}$ with $\boldsymbol{S}\boldsymbol{u}_0 = \sigma^2\boldsymbol{u}_0$. As we found earlier, this ought to be the *least noisy* state-space direction.

We will call this the *alignment ansatz*. We want to show that dynamics are closed under this ansatz, in the sense that it suffices to track the time evolution of the scalars u, v, and c. We will also show that gradient descent dynamics on the full objective become equivalent to gradient descent dynamics on a simpler effective objective.

Note that, under the alignment ansatz,

$$f(t) = \boldsymbol{v}^T e^{\boldsymbol{A}t}\boldsymbol{u} = u(\tau)v(\tau)e^{-c(\tau)t} \, . \tag{78}$$

D.1 DYNAMICS ARE CLOSED UNDER THE ALIGNMENT ANSATZ

$\boldsymbol{u}$ dynamics. General $\boldsymbol{u}$ dynamics have the form

$$\frac{d}{d\tau}\boldsymbol{u} = \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}e^{\boldsymbol{W}^T t}\boldsymbol{v}\, dt - \kappa\boldsymbol{u} \, . \tag{79}$$

Under our ansatz, this simplifies to

$$\begin{aligned}
\frac{d}{d\tau}u(\tau)\boldsymbol{u}_0 &= \left[\int_0^\infty [f_*(t) - f(t)]e^{-c(\tau)t}v(\tau)\, dt - \kappa u(\tau)\right]\boldsymbol{u}_0 \\
&= \left[\int_0^\infty [e^{-c_* t} - u(\tau)v(\tau)e^{-c(\tau)t}]e^{-c(\tau)t}v(\tau)\, dt - \kappa u(\tau)\right]\boldsymbol{u}_0 \\
&= \left[\frac{v(\tau)}{c_* + c(\tau)} - \frac{u(\tau)v(\tau)^2}{2c(\tau)} - \kappa u(\tau)\right]\boldsymbol{u}_0 \, .
\end{aligned} \tag{80}$$

We indeed find that only the scalar $u(\tau)$ changes, as desired.

$\boldsymbol{v}$ dynamics. General $\boldsymbol{v}$ dynamics have the form

$$\frac{d}{d\tau}\boldsymbol{v} = \int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}e^{\boldsymbol{W}t}\boldsymbol{u} - e^{-2\gamma t}e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\, dt - \kappa\boldsymbol{v} \, . \tag{81}$$

Under the ansatz,

$$\begin{aligned}
\frac{d}{d\tau}v(\tau)\boldsymbol{u}_0 &= \left[\int_0^\infty [f_*(t) - f(t)]u(\tau)e^{-c(\tau)t} - \sigma^2 v(\tau)e^{-2c(\tau)t}\, dt - \kappa v(\tau)\right]\boldsymbol{u}_0 \\
&= \left[\int_0^\infty [e^{-c_* t} - u(\tau)v(\tau)e^{-c(\tau)t}]u(\tau)e^{-c(\tau)t} - \sigma^2 v(\tau)e^{-2c(\tau)t}\, dt - \kappa v(\tau)\right]\boldsymbol{u}_0 \\
&= \left[\frac{u(\tau)}{c_* + c(\tau)} - \frac{[u(\tau)^2 + \sigma^2]v(\tau)}{2c(\tau)} - \kappa v(\tau)\right]\boldsymbol{u}_0
\end{aligned} \tag{82}$$

where $\sigma^2 := \boldsymbol{u}_0^T\boldsymbol{S}\boldsymbol{u}_0$ is the noise level in the alignment direction.

$\boldsymbol{W}$ dynamics. General $\boldsymbol{W}$ dynamics have the form

$$\begin{aligned}
\frac{d}{d\tau}\boldsymbol{W} = &\int_0^\infty [f_*(t) - f(t)]e^{-\gamma t}\int_0^t e^{\boldsymbol{W}^T t'}\boldsymbol{v}\boldsymbol{u}^T e^{\boldsymbol{W}^T(t-t')}\, dt' dt \\
&- \int_0^\infty e^{-2\gamma t}\int_0^t e^{\boldsymbol{W}^T t'}\boldsymbol{v}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T(t-t')}\, dt' dt \, .
\end{aligned} \tag{83}$$

Under the ansatz,

$$-\frac{d}{d\tau}c(\tau)\boldsymbol{u}_0\boldsymbol{u}_0^T = u(\tau)v(\tau)\left[\int_0^\infty [f_*(t) - f(t)]\, te^{-c(\tau)t}dt\right]\boldsymbol{u}_0\boldsymbol{u}_0^T \\ - \sigma^2 v(\tau)^2\int_0^\infty te^{-2c(\tau)t}dt\boldsymbol{u}_0\boldsymbol{u}_0^T\ . \tag{84}$$

Simplifying further,

$$-\frac{d}{d\tau}c(\tau)\boldsymbol{u}_0\boldsymbol{u}_0^T = u(\tau)v(\tau)\left[\frac{1}{[c_* + c(\tau)]^2} - \frac{u(\tau)v(\tau)}{4c(\tau)^2}\right]\boldsymbol{u}_0\boldsymbol{u}_0^T \\ - \frac{\sigma^2 v(\tau)^2}{4c(\tau)^2}\boldsymbol{u}_0\boldsymbol{u}_0^T\ , \tag{85}$$

or

$$\frac{d}{d\tau}c(\tau)\boldsymbol{u}_0\boldsymbol{u}_0^T = \left(\frac{v(\tau)^2[u(\tau)^2 + \sigma^2]}{4c(\tau)^2} - \frac{u(\tau)v(\tau)}{[c_* + c(\tau)]^2}\right)\boldsymbol{u}_0\boldsymbol{u}_0^T\ . \tag{86}$$

In all cases, we indeed discover that the dynamics are closed under this ansatz. In summary, because the direction $\boldsymbol{u}_0$ never changes under gradient descent dynamics, it suffices to study the reduced system

$$\begin{aligned}
\frac{d}{d\tau}u(\tau) &= \frac{v(\tau)}{c_* + c(\tau)} - \frac{u(\tau)v(\tau)^2}{2c(\tau)} - \kappa u(\tau) \\
\frac{d}{d\tau}v(\tau) &= \frac{u(\tau)}{c_* + c(\tau)} - \frac{v(\tau)[\sigma^2 + u(\tau)^2]}{2c(\tau)} - \kappa v(\tau) \\
\frac{d}{d\tau}c(\tau) &= \frac{v(\tau)^2[u(\tau)^2 + \sigma^2]}{4c(\tau)^2} - \frac{u(\tau)v(\tau)}{[c_* + c(\tau)]^2}\ .
\end{aligned} \tag{87}$$

D.2 Reduced dynamics are gradient descent on an effective objective

We can make an additional theoretically interesting point about this ansatz: not only is it dynamically closed, but the above ODEs are equivalent to gradient descent on an *effective* objective function. Fortunately, this objective is exactly what one obtains by naively substituting the ansatz into the original objective.

Recall that the full objective equals

$$J(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{W}) = \int_0^\infty \frac{1}{2}\left[f(t) - f_*(t)\right]^2 + \frac{e^{-2\gamma t}}{2}\boldsymbol{v}^T e^{\boldsymbol{W}t}\boldsymbol{S}e^{\boldsymbol{W}^T t}\boldsymbol{v}\, dt + \frac{\kappa}{2}\|\boldsymbol{u}\|^2 + \frac{\kappa}{2}\|\boldsymbol{v}\|^2\ . \tag{88}$$

Substituting our ansatz, it becomes

$$\begin{aligned}
&\frac{1}{2}\int_0^\infty \left\{ \left[uve^{-ct} - e^{-c_* t}\right]^2 + \sigma^2 v^2 e^{-2ct} \right\} dt + \frac{\kappa}{2}(u^2 + v^2) \\
&= \frac{1}{2}\int_0^\infty \left\{ u^2 v^2 e^{-2ct} - 2uve^{-(c_* + c)t} + e^{-2c_* t} + \sigma^2 v^2 e^{-2ct} \right\} dt + \frac{\kappa}{2}(u^2 + v^2) \\
&= \frac{1}{2}\left\{ u^2 v^2 \frac{1}{2c} - 2uv\frac{1}{c_* + c} + \frac{1}{2c_*} + \sigma^2 v^2 \frac{1}{2c} \right\} + \frac{\kappa}{2}(u^2 + v^2)\ .
\end{aligned} \tag{89}$$

Simplifying, we obtain

$$\begin{aligned}
J &= \frac{u(\tau)^2 v(\tau)^2}{4c(\tau)} - \frac{u(\tau)v(\tau)}{c_* + c(\tau)} + \frac{1}{4c_*} + \frac{\sigma^2 v(\tau)^2}{4c(\tau)} + \frac{\kappa}{2}(u^2 + v^2) \\
&= \frac{1}{4c(\tau)}\left[u(\tau)v(\tau) - \frac{2c(\tau)}{c_* + c(\tau)}\right]^2 + \left(\frac{1}{4c_*} - \frac{c(\tau)}{[c_* + c(\tau)]^2}\right) + \frac{\sigma^2 v(\tau)^2}{4c(\tau)} + \frac{\kappa}{2}[u(\tau)^2 + v(\tau)^2]\ .
\end{aligned} \tag{90}$$

It is straightforward to verify that the above ODEs directly correspond to gradient descent on this objective, which establishes our claim.

E CHARACTERIZATION OF AMPLITUDE-COLLAPSE PHASE TRANSITION

In this appendix, we characterize the noise-induced phase transition associated with amplitude collapse (see Sec. 6). The objective J has the form

$$J = \frac{1}{4c}\left(uv - \frac{2c}{c_* + c}\right)^2 + \left(\frac{1}{4c_*} - \frac{c}{(c_* + c)^2}\right) + \frac{\sigma^2 v^2}{4c} + \frac{\kappa}{2}(u^2 + v^2)\,. \tag{91}$$

This amounts to determining the conditions under which J has a nontrivial optimum.

E.1 MINIMIZING J WITH RESPECT TO ITS ARGUMENTS

Let us begin by trying to optimize J with respect to its arguments—although we will find it useful to work in terms of slightly different variables. Define the *filter amplitude* $N := uv$ and the *scale ratio* $r := u/v$. Since $v^2 = N/r$ and $u^2 = Nr$, we can write J as

$$J(r, N, c) = \frac{1}{4c}\left(N - \frac{2c}{c_* + c}\right)^2 + \left(\frac{1}{4c_*} - \frac{c}{(c_* + c)^2}\right) + \frac{\sigma^2 N}{4cr} + \frac{\kappa}{2}N\left(\frac{1}{r} + r\right)\,. \tag{92}$$

Minimizing with respect to r. The derivative of J with respect to r equals

$$\frac{\partial J}{\partial r} = \frac{\kappa}{2}N - \frac{\kappa}{2}N\left(1 + \frac{\sigma^2}{2\kappa c}\right)\frac{1}{r^2} = 0 \tag{93}$$

which implies that

$$r^2 = 1 + \frac{\sigma^2}{2\kappa c} \tag{94}$$

at the minimum. Note that it depends on c but not on N. In a slight abuse of notation, below we will use r to denote the value of the scale ratio at the minimum. Without loss of generality, choose the positive solution so that

$$\boxed{r = \sqrt{1 + \frac{\sigma^2}{2\kappa c}}}\,. \tag{95}$$

Since r is defined as the ratio of u and v, this immediately implies that u and v have the same sign at a nontrivial minimum. Relatedly, it must be true that $N \geq 0$, since $N = u^2/r \geq 0$.

Minimizing with respect to N. The derivative of J with respect to N equals

$$\begin{aligned}\frac{\partial J}{\partial N} &= \frac{1}{2c}\left(N - \frac{2c}{c_* + c}\right) + \frac{\kappa}{2}\left(1 + \frac{\sigma^2}{2\kappa c}\right)\frac{1}{r} + \frac{\kappa}{2}r \\ &= \frac{1}{2c}\left(N - \frac{2c}{c_* + c}\right) + \kappa r\,.\end{aligned} \tag{96}$$

Setting the left-hand side equal to zero, it appears true that

$$N = \frac{2c}{c_* + c} - 2\kappa r c \tag{97}$$

at the minimum. But since our r minimization implied that $N \geq 0$, we actually have

$$N = 2c\left[\frac{1}{c_* + c} - \kappa r\right]_+ \tag{98}$$

where $[\cdots]_+$ denotes rectification (i.e., a ReLU). That is, if the bracketed quantity is negative or zero, J is minimized when $N = 0$. This is a sign of *amplitude collapse*—if $N = 0$ is optimal, it is optimal to take $f(t) = 0$.

If we define the quantity

$$\boxed{h(c) := \frac{1}{c_* + c} - \kappa r}\,, \tag{99}$$

we can write the optimal N as

$$\boxed{N = 2c[h(c)]_+}\,. \tag{100}$$

Rewriting the objective in terms of N. In terms of N, the objective can be written at its minimum as

$$\begin{aligned} J(c) &= c\left(\frac{1}{c_*+c} - [h(c)]_+\right)^2 + \frac{1}{4c_*} - \frac{c}{(c_*+c)^2} + 2\kappa c[h(c)]_+ r(c) \\ &= \frac{1}{4c_*} + c[h(c)]_+\left(-\frac{2}{c_*+c} + [h(c)]_+\right) + 2\kappa c[h(c)]_+ r(c) \\ &= \frac{1}{4c_*} + c[h(c)]_+\left(2\kappa r(c) - \frac{2}{c_*+c} + [h(c)]_+\right) . \end{aligned} \tag{101}$$

Either $h(c) \geq 0$, in which case the bracketed quantity equals $-h(c)$, or $h(c) < 0$, in which case the $[h(c)]_+$ prefactor makes the entire quantity zero. We conclude that

$$\boxed{J(c) = \frac{1}{4c_*} - c[h(c)]_+^2} \tag{102}$$

at the minimum. This immediately tells us that if $h < 0$ for all $c \geq 0$, there is a trivial solution with $N = 0$ and J independent of c (see Fig. 5c).

E.2 Determining the phase boundary

While it appears prohibitively difficult to derive the exact c that minimizes J (since this requires solving a high-order polynomial equation), we can instead determine where there is a transition from a nontrivial solution to a trivial one.

Phase boundary conditions. Eq. 102 implies that a nontrivial solution exists if and only if $h(c) > 0$ for some $c > 0$, since this yields $N > 0$ and a nontrivial value $c > 0$ at the minimum. On the other hand, if $h(c) \leq 0$ for all $c > 0$, then $N = 0$ and the trivial solution is optimal. The phase boundary therefore occurs when the maximum of $h(c)$ just touches zero. Hence, the phase boundary is characterized by the two conditions

$$\boxed{h(c) = 0 \qquad\qquad h'(c) = 0} . \tag{103}$$

Explicitly, we want

$$\begin{aligned} \frac{1}{c_*+c} &= \kappa\sqrt{1+\frac{\sigma^2}{2\kappa c}} \\ \frac{1}{(c_*+c)^2} &= \kappa\frac{\frac{\sigma^2}{2\kappa c^2}}{2\sqrt{1+\frac{\sigma^2}{2\kappa c}}} . \end{aligned} \tag{104}$$

Relatedly, a nontrivial solution only exists when $c_*\kappa < 1$, since for $c_*\kappa \geq 1$,

$$h(c) = \frac{1}{c_*+c} - \kappa r \leq \frac{1}{c_*+c} - \kappa < \frac{1}{c_*} - \kappa \leq 0 \tag{105}$$

for all $c > 0$. In this regime, because $h(c)$ is always nonpositive, $\sigma^2_{\text{crit}} = 0$.

Some algebra. It will be helpful to write σ^2 and κ separately in terms of c. Divide the second equation by the first to obtain

$$\frac{1}{c_*+c} = \frac{\sigma^2}{4\kappa c^2 r^2} . \tag{106}$$

Rearranging,

$$r^2 = 1 + \frac{\sigma^2}{2\kappa c} = \frac{\sigma^2}{4\kappa c^2}(c_*+c) \implies 1 + \frac{\sigma^2}{4\kappa c} = \frac{\sigma^2}{4\kappa c^2}c_* , \tag{107}$$

so

$$\frac{\sigma^2}{2\kappa c} = \frac{2}{c_*/c - 1} . \tag{108}$$

This implies that r can be written in terms of c as

$$r(c)^2 = 1 + \frac{2}{c_*/c - 1} = \frac{c_*/c + 1}{c_*/c - 1} = \frac{c_* + c}{c_* - c} \,. \tag{109}$$

Using this result, the first condition can be rewritten as

$$\kappa^2 = \frac{1}{(c_* + c)^2 r^2} = \frac{c_* - c}{(c_* + c)^3} \,. \tag{110}$$

Let $x := c/c_*$ denote the non-dimensionalized learned rate. In terms of x, our two conditions read

$$\boxed{\kappa^2 = \frac{1}{c_*^2} \frac{1 - x}{(1 + x)^3} \qquad\qquad \sigma^2 = \frac{4 c_* \kappa x^2}{1 - x}} \,. \tag{111}$$

Since both σ^2 and κ can be written in terms of the optimal value of c (or equivalently, x), it suffices to determine this value. We can then write σ^2 in terms of κ, and hence determine the phase boundary.

Solving a cubic equation. Note that the first condition expresses a cubic equation (in terms of c_* and κ) whose solution is x:

$$c_*^2 \kappa^2 (1 + x)^3 = 1 - x \,. \tag{112}$$

Note that it only has a valid solution if $0 \leq x < 1$.

Generic cubic equations are annoying to solve, but they can be made tractable with a clever substitution. Consider the inspired choice

$$\boxed{q := \sqrt{\frac{1 - x}{1 + x}}} \tag{113}$$

which implies that

$$\boxed{x = \frac{1 - q^2}{1 + q^2}} \,. \tag{114}$$

Relatedly,

$$\begin{aligned} 1 + x &= \frac{2}{1 + q^2} \\ 1 - x &= \frac{2q^2}{1 + q^2} \,. \end{aligned} \tag{115}$$

In terms of this new variable q, our cubic equation becomes

$$c_*^2 \kappa^2 = \frac{q^2}{4} (1 + q^2)^2 \,. \tag{116}$$

Since $q \geq 0$ by definition, we can take the square root of both sides to find that

$$2 c_* \kappa = q(1 + q^2) \,, \tag{117}$$

or equivalently that

$$\boxed{q^3 + q - 2\kappa c_* = 0} \,. \tag{118}$$

This is a *depressed cubic equation*. Its discriminant equals

$$\Delta = -4 - 27(-2\kappa c_*)^2 < 0 \,, \tag{119}$$

so it has one real root and two complex conjugate roots. Because a valid solution must be *real*, we are interested in the real root.

Using a standard formula for this root, we find that it equals

$$\boxed{q(\kappa) = \frac{2}{\sqrt{3}} \sinh\left[\frac{1}{3} \text{arsinh}(3\sqrt{3}\kappa c_*)\right]} \tag{120}$$

where

$$\text{arsinh}(x) := \log(x + \sqrt{x^2 + 1}) \,. \tag{121}$$

Determining the critical noise level. Combining this result with our earlier formula for σ^2 in terms of x,

$$\sigma^2 = \frac{4c_*\kappa x^2}{1-x} = 4c_*\kappa\frac{(1-q^2)^2}{2q^2(1+q^2)} \,. \tag{122}$$

Since $2c_*\kappa = q(1+q^2)$, this becomes

$$\boxed{\sigma^2_{\text{crit}} = \frac{(1-q(\kappa)^2)^2}{q(\kappa)}} \,. \tag{123}$$

The final phase boundary result. Summarizing, at the boundary, provided $0 < \kappa c_* \leq 1$,

$$\begin{aligned} q &= \frac{2}{\sqrt{3}}\sinh\left[\frac{1}{3}\text{arsinh}(3\sqrt{3}\kappa c_*)\right] \\ \sigma^2 &= \frac{(1-q^2)^2}{q} \,. \end{aligned} \tag{124}$$

F CHARACTERIZATION OF FREQUENCY-RUNAWAY PHASE TRANSITION

In this appendix, we characterize the 'frequency-runaway' phase transition discussed in Sec. 7 that occurs when u and v are frozen (with $u = v = 1$) but c is trainable.

If u and v are frozen after alignment, the reduced objective becomes

$$J = \frac{1+\sigma^2}{4c(\tau)} - \frac{1}{c_* + c(\tau)} + \frac{1}{4c_*} \tag{125}$$

up to an additive constant. (Because it is constant, we have removed the regularization term). The negative gradient of J equals

$$\begin{aligned} -\frac{\partial J}{\partial c} &= \frac{(1+\sigma^2)}{4c^2} - \frac{1}{(c_*+c)^2} \\ &= \frac{(1+\sigma^2)(c_*+c)^2 - 4c^2}{4c^2(c_*+c)^2} \\ &= \frac{[\sqrt{1+\sigma^2}(c_*+c)+2c][\sqrt{1+\sigma^2}(c_*+c)-2c]}{4c^2(c_*+c)^2} \,, \end{aligned} \tag{126}$$

so c has dynamics

$$\frac{d}{d\tau}c = \frac{[\sqrt{1+\sigma^2}(c_*+c)+2c]}{4c^2(c_*+c)^2}[\sqrt{1+\sigma^2}(c_*+c)-2c] \,. \tag{127}$$

The fixed point of c dynamics can only depend on the second factor in the numerator, since the first factor is always positive for $c \geq 0$ and $c_* > 0$. The second factor equals zero at

$$\boxed{c_{\text{opt}} = \frac{\sqrt{1+\sigma^2}\, c_*}{2 - \sqrt{1+\sigma^2}}} \,, \tag{128}$$

so when c_{opt} is well-defined, it is the fixed point of c dynamics.

Importantly, this quantity blows up when its denominator equals zero, which occurs at the critical noise level $\sigma^2_{\text{crit}} := 3$. Just below that noise level,

$$\boxed{c_{\text{opt}} \approx \frac{8c_*}{\sigma^2_{\text{crit}} - \sigma^2}} \,, \tag{129}$$

so it diverges as $1/(\sigma^2_{\text{crit}} - \sigma^2)$. For $\sigma^2 \geq \sigma^2_{\text{crit}}$, the second factor equals

$$\sqrt{1+\sigma^2}(c_*+c) - 2c = \sqrt{1+\sigma^2}c_* + (\sqrt{1+\sigma^2} - 2)c \,, \tag{130}$$

which is always positive. This means that, beyond the critical noise level, dynamics drive c to infinity. Hence, beyond the threshold, J is a decreasing function of c and has no finite minimum.