High-dimensional neuronal activity from low-dimensional latent dynamics: a solvable model

Valentin Schmutz*
University College London
WC1E 6BT London, UK
v.schmutz@ucl.ac.uk

Ali Haydaroglu*
University College London
WC1E 6BT London, UK
ali.haydaroglu.20@ucl.ac.uk

Shuqi Wang
École Polytechnique Fédérale de Lausanne
1015 Lausanne, Switzerland
shuqi.wang@epfl.ch

Yixiao Feng
Shanghai Jiao Tong University
200240 Shanghai, China
yf2887@nyu.edu

Matteo Carandini
University College London
WC1E 6BT London, UK
m.carandini@ucl.ac.uk

Kenneth D. Harris
University College London
WC1E 6BT London, UK
kenneth.harris@ucl.ac.uk

Abstract

Computation in recurrent networks of neurons has been hypothesized to occur at the level of low-dimensional latent dynamics, both in artificial systems and in the brain. This hypothesis seems at odds with evidence from large-scale neuronal recordings in mice showing that neuronal population activity is high-dimensional. To demonstrate that low-dimensional latent dynamics and high-dimensional activity can be two sides of the same coin, we present an analytically solvable recurrent neural network (RNN) model whose dynamics can be exactly reduced to a low-dimensional dynamical system, but generates an activity manifold that has a high linear embedding dimension. This raises the question: Do low-dimensional latents explain the high-dimensional activity observed in mouse visual cortex? Spectral theory tells us that the covariance eigenspectrum alone does not allow us to recover the dimensionality of the latents, which can be low or high, when neurons are nonlinear. To address this indeterminacy, we develop Neural Cross-Encoder (NCE), an interpretable, nonlinear latent variable modeling method for neuronal recordings, and find that high-dimensional neuronal responses to drifting gratings and spontaneous activity in visual cortex can be reduced to low-dimensional latents, while the responses to natural images cannot. We conclude that the high-dimensional activity measured in certain conditions, such as in the absence of a stimulus, is explained by low-dimensional latents that are nonlinearly processed by individual neurons.

1 Introduction

The mammalian cortex comprises a large number of neurons, which, in principle, should allow it to use a high-dimensional neural code to represent sensory, motor, and cognitive information. Nevertheless, multi-neuronal recordings in nonhuman primates [1–4] have suggested that cortical populations perform computations by approximating low-dimensional dynamical systems [5, 6], with neuronal firing rates lying on a low-dimensional "neural manifold" [7]. In support of this hypothesis,

*equal contribution

low-dimensional dynamics have been inferred from multi-neuronal recordings through a wide variety of methods [8–23]; they spontaneously emerge in recurrent neural networks (RNNs) trained to solve behavioral tasks [2, 24–34]; and they appear in several theoretical models of noise-robust neuronal population dynamics [35–38]. A result that might at first sight challenge the low-dimensional dynamical systems hypothesis is that visual cortical population activity in mice has high linear dimension [39, 40] with shared neuronal covariance having a heavy-tailed eigenspectrum (see also [41] and [42] for recordings in cerebellum and across cortex, respectively). In particular, the shared covariance eigenspectrum has a power-law tail with an exponent close to 1 ($\alpha \approx 1.04$) [39] for responses to natural images and an exponent of $\alpha \approx 1.14$ for spontaneous activity [40]. Are these two views on the dimensionality of population activity compatible? Namely, can a low-dimensional dynamical system produce a neural manifold that has a high linear embedding dimension?

Here, we first construct a solvable RNN model that reconciles the low- and high-dimensional perspectives on population activity by carefully disambiguating the *linear* dimension of the system *before* and *after* the neurons' nonlinearity, which we refer to as the pre- and post-activation dimension, respectively. This dichotomy refines the usual distinction between linear and "intrinsic" dimension [39, 43, 44], since the intrinsic dimension of a system is the same before and after any continuous, injective nonlinearity. Using the notions of pre- and post-activation linear dimensions, we show that our RNN can be exactly reduced to a low-dimensional dynamical system in the space of pre-activations, making the pre-activations low-dimensional. Then, we show that these latent dynamics generate high-dimensional post-activation activity that has a power-law covariance eigenspectrum. (In this work, dimension will always refer to linear dimension, unless stated otherwise.)

Before analyzing experimental recordings, we revisit the spectral theory of infinite-width neural networks (random feature kernels) [45–47] to quantitatively relate the pre-activation dimension, the neuronal activation function, and the post-activation covariance eigenspectrum. This three-way relationship tells us that high-dimensional activity is consistent with both low- and high-dimensional pre-activations. To uncover the pre-activation dimension of high-dimensional activity in visual cortex, we perform two-photon calcium recordings of tens of thousands of neurons from mouse visual cortex, and infer the pre-activation dimension using the Neural Cross-Encoder (NCE), an interpretable, nonlinear latent variable modeling method which models the activity of each neuron as a simple linear-nonlinear readout of low-dimensional latents. NCE reveals that both the responses to drifting gratings and spontaneous activity can be well approximated by low-dimensional pre-activations, but that responses to natural images cannot. This suggests that the encoding of natural images in visual cortex is already high-dimensional in the space of pre-activations.

2 Solvable RNN Model

To demonstrate how high-dimensional post-activations can arise from low-dimensional pre-activation dynamics, we first present a solvable RNN model whose autonomous dynamics is low-dimensional in the space of pre-activations, but high-dimensional in the space of post-activations, with the post-activations producing a power-law covariance eigenspectrum.

We consider an RNN consisting of N rate-units (neurons). The pre-activation x_i of neuron i evolves according to

$$\dot{x}_i = -x_i + \frac{1}{N}\sum_{j=1}^{N} W_{ij}\phi(x_j), \tag{1}$$

where W_{ij} denotes the synaptic weight from neuron j to neuron i, and $\phi : \mathbb{R} \to \mathbb{R}_{\geq 0}$ is a nonlinear activation function converting the pre-activations into post-activations (firing rates). To define the weights W_{ij}, we randomly place neurons on a ring [48–50] by assigning to each neuron i an independent and uniformly distributed angle $\theta_i \in [0, 2\pi)$ (Fig. 1A). The weights W_{ij} are then given by the following shifted cosine function:

$$W_{ij} := J\cos(\theta_i - \theta_j - \Delta). \tag{2}$$

The shift Δ in Eq. (2) makes the weights asymmetric, with neurons sending their strongest excitatory output to neurons located at an angle Δ counter-clockwise (Fig. 1B). To make the model solvable, we assume that the activation function ϕ is the Heaviside step function Θ, i.e.,

$$\phi(x) = \Theta(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ 0, & \text{if } x < 0. \end{cases} \tag{3}$$

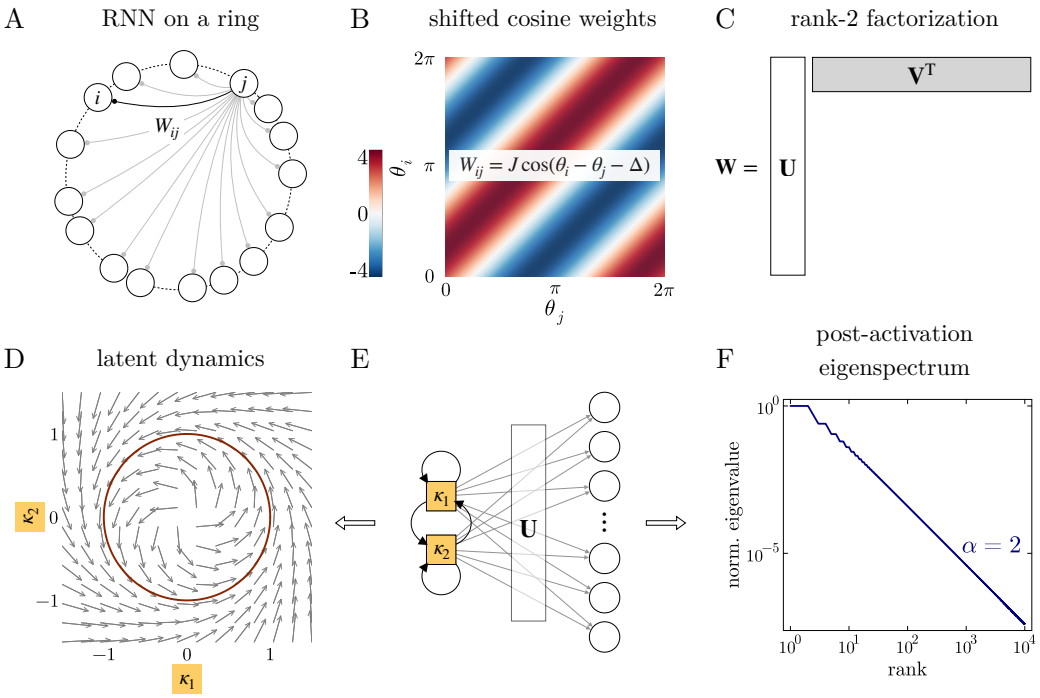

Figure 1: **Solvable RNN model.** (**A**) Schematic of the RNN model. (**B**) Shifted cosine function defining the position-dependent synaptic weights W_{ij}. (**C**) Factorization of weight matrix $\mathbf{W}$ as the product of a 2-column matrix $\mathbf{U}$ and two-row matrix $\mathbf{V}^{\mathrm{T}}$. (**D**) Vector field of the dynamics of the latent variables, κ_1 and κ_2, in the large-network limit. The latent dynamics produces a stable limit cycle on the unit circle. (**E**) Graphical representation of the RNN's effective dynamics. (**F**) Post-activation covariance eigenspectrum. The eigenvalues follow a power-law with decay exponent $\alpha = 2$. They are normalized such that the largest eigenvalue is 1. The activation function ϕ used is the step function defined in Eq. (3). Shown in D and F are theoretical values given by Eqs. (5) and (8), respectively, which closely match simulations of large networks (not shown).

2.1 Low-dimensional pre-activation dynamics

The RNN model defined above has low-dimensional pre-activation dynamics because the weight matrix $\mathbf{W}$ has rank 2. Indeed, using an elementary trigonometric identity,[2] $\mathbf{W}$ can be factorized as the outer product $\mathbf{W} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$ of two $N \times 2$-matrices (Fig. 1C), with

$$\mathbf{U} := \begin{pmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \vdots & \vdots \\ \cos(\theta_N) & \sin(\theta_N) \end{pmatrix} \quad \text{and} \quad \mathbf{V} := J \begin{pmatrix} \cos(\theta_1 + \Delta) & \sin(\theta_1 + \Delta) \\ \vdots & \vdots \\ \cos(\theta_N + \Delta) & \sin(\theta_N + \Delta) \end{pmatrix}.$$

Then, following Beiran et al. [51], we can reduce the N-dimensional system, Eq. (1), to a 2-dimensional system describing the dynamics of the latent variables $\boldsymbol{\kappa} := \mathbf{U}^{\dagger}\mathbf{x}$, where † denotes the pseudoinverse and $\mathbf{x}$ the N-dimensional vector of pre-activations $(x_1, \ldots, x_N)^{\mathrm{T}}$. The dynamics of the latent variables follows

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \tfrac{1}{N}\mathbf{v}^{\mathrm{T}}\phi(\mathbf{U}\boldsymbol{\kappa}),$$

where ϕ is applied element-wise to the N-dimensional vector of pre-activations $\mathbf{U}\boldsymbol{\kappa} = \mathbf{x}$. In the equation above, the vector $\phi(\mathbf{U}\boldsymbol{\kappa}) = \phi(\mathbf{x})$ represents the joint post-activations (firing rates) of the N neurons.

Taking the number of neurons $N \to \infty$ yields a neural field limit [52] where the sum over neurons becomes an integral over the ring,

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \int_{\theta=0}^{2\pi} \mathbf{v}(\theta)\phi\left(\mathbf{u}(\theta) \cdot \boldsymbol{\kappa}\right) \frac{\mathrm{d}\theta}{2\pi}, \tag{4}$$

[2] $\cos(\alpha - \beta) = \cos(\alpha)\cos(\beta) + \sin(\alpha)\sin(\beta)$

with $\mathbf{u}(\theta) := (\cos(\theta), \sin(\theta))^{\mathrm{T}}$ and $\mathbf{v}(\theta) := J(\cos(\theta + \Delta), \sin(\theta + \Delta))^{\mathrm{T}}$. Equation (4) describes the dynamics of the latent variables $\boldsymbol{\kappa}$ as the solution to a 2-dimensional dynamical system whose vector field involves an integral over the "circuit structure" [52] (the ring).

Since ϕ is the step function, the integral over the ring in Eq. (4) can be solved, and we obtain the solvable 2-dimensional dynamical system,

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|}, \tag{5}$$

when $J := \pi\sqrt{2}$ and $\Delta := \pi/4$ (derivation presented in Appendix A). Equation (5) generates a stable limit cycle over the unit circle (Fig. 1D), which implies that the latent variables $\boldsymbol{\kappa}$ will eventually rotate on the unit circle indefinitely. In Appendix C, we provide other examples of low-rank RNNs for which the latent dynamics can be expressed in a tractable form similar to Eq. (5).

Neuronal activity, modeled here as the post-activations $\phi(\mathbf{U}\boldsymbol{\kappa})$, are simple *linear-nonlinear* readouts of latent variables $\boldsymbol{\kappa}$ (Fig. 1E). Hence, we have effectively reduced the dynamics of the RNN, Eq. (1), to a 2-dimensional latent dynamical system. We will say that neuronal activity is a *linear-nonlinear* function of latent variables if it is given by the composition of a linear mapping ($\mathbf{U}$) and an element-wise, nondecreasing nonlinear mapping (ϕ).

2.2 Post-activations produce a power-law eigenspectrum

Since the dynamics of the latent variables is solvable in the large-network limit, and rotates on the unit circle, we can compute the correlation between the post-activations of two neurons. For any pair of neurons i and j, with positions θ_i and θ_j on the ring, respectively, the correlation of their post-activations C_{ij} is, in the long-recording limit, given by

$$C_{ij} = \frac{2}{\pi}\left(\pi - |\theta_i - \theta_j|\right) - 1, \tag{6}$$

where $|\theta_i - \theta_j| := \cos^{-1}(\cos(\theta_i - \theta_j))$ is the absolute angle difference θ_i and θ_j (derivation presented in Appendix B).

We can find the eigenvalue spectrum of post-activations by noting that as the number of recorded neurons $M \to \infty$, the eigenvalues of the $M \times M$ correlation matrix $\mathbf{C}^{(M)}$ defined by Eq. (6) converge to the eigenvalues of an integral operator. Since the angles θ_i are independently and uniformly sampled on circle $[0, 2\pi)$, the correlation matrix $\mathbf{C}^{(M)}$ is a so-called Euclidean random matrix [53], that is, a matrix whose entries are given by the pairwise distances between randomly sampled points in a given space. Writing $\lambda_1^{(M)} \geq \lambda_2^{(M)} \geq \cdots \geq \lambda_N^{(M)}$ for the ranked eigenvalues of the matrix $\mathbf{C}^{(M)}$, random matrix theory [54, 55] tells us that, as $M \to \infty$, the scaled eigenvalues $\{\lambda_n^{(M)}/M\}_{n=1}^{M}$, converge (in a ℓ_2 sense) to the eigenvalues of the integral operator,

$$f(\theta) \mapsto \int_{\theta'=0}^{2\pi} \left[\frac{2}{\pi} \underbrace{\left(\pi - |\theta - \theta'|\right)}_{\text{0-th arc-cosine kernel}} - 1\right] f(\theta') \frac{\mathrm{d}\theta'}{2\pi}. \tag{7}$$

The eigenvalues of this integral operator can be computed analytically. In Eq. (7), we have highlighted the presence of the 0-th arc-cosine kernel $k_0(\theta, \theta') := \pi - |\theta - \theta'|$ of Cho and Saul [45], which is well-known in machine learning, and whose eigenvalues have been computed in [46]. In short, by the rotational invariance of k_0, we have that, for any positive integer m, the functions $\theta \mapsto \cos(m\theta)$ and $\theta \mapsto \sin(m\theta)$ are orthogonal eigenfunctions of the operator Eq. (7) sharing the same eigenvalue,

$$\frac{2}{\pi^2} \int_{\theta=0}^{\pi} (\pi - \theta) \cos(m\theta) \mathrm{d}\theta = \frac{2}{\pi^2} \frac{1 - (-1)^m}{m^2}.$$

Using this result, we obtain that the ranked eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots$ of the operator Eq. (7) are given by

$$\lambda_{2m-1} = \lambda_{2m} = \frac{4}{\pi^2} m^{-2}, \qquad \forall m \in \mathbb{N}^*, \tag{8}$$

that is, eigenvalues come in identical pairs that decay exactly as a power law with decay exponent $\alpha = 2$ (Fig. 1F). Hence, post-activations are high-dimensional in the sense that their covariance eigenspectrum has a heavy tail [39].

In summary, this solvable model shows that low-dimensional dynamics in the space of pre-activations can generate high-dimensional post-activations. The heavy tail of the covariance eigenspectrum implies that post-activations are not confined to any finite-dimensional linear subspace. Formally, the smallest vector space containing the post-activations generated by our model has the same size as the infinite-dimensional reproducing kernel Hilbert space associated with the kernel k_0. We stress that, in this model, the heavy tail of the post-activation eigenspectrum is not due to noise, since we used a deterministic, non-chaotic RNN. Also, all the results presented above remain exact if the rate-units in Eq. (1) are replaced by linear-nonlinear-Poisson neurons, as spike noise cancels out in the limits we consider [38, 52].

3 Post-activation eigenspectrum depends on pre-activation dimension and activation function

To shed light on the relationship between the post-activation eigenspectrum, pre-activation dimension, and the activation function ϕ, we now turn to a more general setup, which allows us to relax some of the strong assumptions of the solvable model (Fig. 1E). First, we allow the number of latent variables d to be greater than 2, assuming that the latent variables, denoted by $\mathbf{z}$, are uniformly distributed on the unit sphere $\mathbb{S}^{d-1}$ in $\mathbb{R}^d$. We assume that the pre-activations of the network are determined by passing the latent activity through a $N \times d$ feedforward weight matrix $\mathbf{U}$ with i.i.d. standard normal entries. In this setup (Fig. 2A), we call d the *pre-activation dimension*, as it sets the linear dimensionality of the pre-activations. In the solvable model of Sec. 2.2, for example, the pre-activation dimension was $d = 2$ (Fig. 1E). Finally, we replace the step function, Eq. (3), by the general rectified power activation function

$$\phi_{p,c}(x) := [\max(0, x + c)]^p, \tag{9}$$

where the activation parameter $p \in \mathbb{R}_{\geq 0}$ is a nonnegative real value and the bias $c \in \mathbb{R}$. (By convention, $\phi_{0,c}(x) := \Theta(x + c)$.)

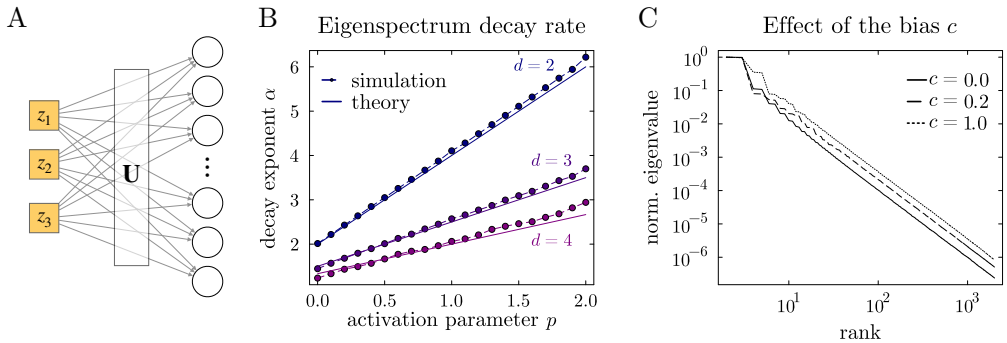

Figure 2: **Eigenspectrum of the random feature kernel** (11)**.** (**A**) Schematic of the feedforward setup (in the case of three input variables). (**B**) Comparison between simulations of the eigenspectrum decay rate α and the theoretical value predicated by Conjecture 1, for $p \in [0, 2]$ and $d = 2, 3, 4$. Up to finite-size effects (number of neurons $N = 2 \cdot 10^4$ and number of inputs $T = 10^4$), simulations match the theoretical prediction $\alpha = 1 + \frac{2p+1}{d-1}$. (In this plot, $c = 0$.) (**C**) Simulations of the eigenspectrum for various bias parameter c, while keeping p and d fixed. (In this plot, $p = 0$ and $d = 2$, which gives $\alpha = 2$. Eigenvalues are normalized as in Fig. 1F.)

This setup can be analyzed within the framework of random feature kernels (see [56, Sec. 9.5]). Denoting μ_{d-1} the uniform probability measure on the sphere $\mathbb{S}^{d-1}$, let us take T independent latent variable samples $\mathbf{z}_1, \ldots, \mathbf{z}_T$ from μ_{d-1}, and define the $N \times T$ post-activation matrix $\mathbf{A}^{(N,T)} := (\phi_{p,c}(\mathbf{U}\mathbf{z}_1), \ldots, \phi_{p,c}(\mathbf{U}\mathbf{z}_T))$. In the limits $N \to \infty$ and $T \to \infty$ taken successively, the covariance eigenspectrum of $\mathbf{A}^{(N,T)}$ converges (when properly scaled) to the eigenvalue spectrum of the integral operator

$$f(\mathbf{z}) \mapsto \int_{\mathbb{S}^{d-1}} K_{p,c,d}(\mathbf{z}, \mathbf{z}') f(\mathbf{z}') \mathrm{d}\mu_{d-1}(\mathbf{z}'), \tag{10}$$

where $K_{p,c,d} : \mathbb{S}^{d-1} \times \mathbb{S}^{d-1} \to \mathbb{R}$ is the *random feature kernel*

$$K_{p,c,d}(\mathbf{z}, \mathbf{z}') := \mathbb{E}_{\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)}[\phi_{p,c}(\boldsymbol{\xi} \cdot \mathbf{z})\phi_{p,c}(\boldsymbol{\xi} \cdot \mathbf{z}')] \tag{11}$$

(see [46, 56] or Appendix D for more details).

Known theoretical results [46] and simulations suggest the following conjecture:

Conjecture 1. *For any $p \in \mathbb{R}_{\geq 0}$, $c \in \mathbb{R}$, and any integer $d \geq 2$, the ranked eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots$ of the integral operator* (10) *obey the following power-law decay:*

$$\lambda_n \asymp n^{-\alpha} \quad \textit{with} \quad \alpha = 1 + \frac{2p+1}{d-1}, \tag{12}$$

where $a_n \asymp b_n$ means $\lim_{n \to +\infty} a_n / b_n = C \in (0, +\infty)$.

This conjecture goes beyond, but is consistent with, known theoretical results for the special cases $p = 0$ (step function) and $p = 1$ (ReLU) with $c = 0$ [46]. To the best of our knowledge, Conjecture 1 is not a straightforward consequence of any existing result in theoretical machine learning [47, 57, 58] or harmonic analysis [59–62], hence our presentation of Eq. (12) as a conjecture. Note that when the activation parameter p is an integer, $\phi_{p,c}$ is p-times weakly differentiable, that is, the first p weak derivatives[3] of $\phi_{p,c}$ are all locally integrable. This, and the fact that the bias c does not affect the decay rate, suggest a further extension of the conjecture to more general activation functions, with p replaced by the weak differentiability of the activation function.

We tested Conjecture 1 numerically by performing PCA on large post-activation matrices $\mathbf{A}^{(N,T)}$. The linear and continuous dependence of the decay exponent α on the activation parameter p predicted by Eq. (12) was confirmed in simulations (Fig. 2B). Simulations also confirmed that the bias c of the activation function does not affect the decay rate (Fig. 2C), a fact already mentioned in [63, 64].

To summarize, the spectral theory of random feature kernels suggests a three-way relationship between the power-law tail exponent of the post-activation eigenspectrum, the pre-activation dimension, and the activation function. This relationship should hold when we can consider the neurons as linear-nonlinear functions of the latent vector, with weights that vary randomly and independently between neurons. The relationship suggests that, when high-dimensional neuronal activity (modeled here as post-activations) is observed [39, 40], two scenarios are possible: high-dimensional activity could arise from nonlinear transformation of low-dimensional latent states, or it could reflect pre-activations that are already high-dimensional. To distinguish these two scenarios, we propose, in what follows, a method for inferring the pre-activation dimension of neuronal activity in experimental recordings.

4 Latent Variable Modeling of Neuronal Recordings

4.1 Experimental Data

We conducted large-scale volumetric two-photon microscopy on awake, adult mice during visual stimulation and spontaneous activity. We targeted primary and higher visual cortices with a Light Beads Microscope [65], and extracted deconvolved activity traces for 19,223 $\pm$ 2,948 neurons using Suite3D [66] as described in Appendix F. Recordings were performed in three stimulus conditions: (1) responses to 320 full-field drifting grating stimuli with 2-14 repeats each; (2) responses to 1866 natural images with 2 repeats each; (3) spontaneous activity in the absence of stimuli for 10-15 minutes.

4.2 Neural Cross-Encoder (NCE)

The Neural Cross-Encoder (NCE) divides neurons randomly into two sets: a source set and a target set. It predicts the activity $\mathbf{b}_t$ of the target set from the source set $\mathbf{a}_t$ via a non-linear readout of a set of latents, $\mathbf{z}_t$ (Fig. 3A). NCE uses a multi-layer feedforward encoder $\mathcal{E}$ that ends in a bottleneck layer whose activity $\mathbf{z}_t = \mathcal{E}(\mathbf{a}_t)$ represents a low-dimensional latent state estimated from the source neurons. The reason to use this rather than an autoencoder, which predicts one set of neurons from themselves, is to discard variability that is not shared across neurons. The NCE we used here has a

[3]The k-th weak derivative of a function $f : \mathbb{R} \to \mathbb{R}$ is the defined as the function $g \in L^1_{\text{loc}}$ that satisfies $\int_{\mathbb{R}} \varphi(x) g(x) \mathrm{d}x = (-1)^k \int_{\mathbb{R}} \varphi^{(k)}(x) f(x) \mathrm{d}x$, for all $\varphi \in C_c^\infty(\mathbb{R})$.

single power-ReLU output layer, matching the setup of section 3, and a 3-layer encoder allowing flexible estimation of latent variables, so that the number of latent variables can be readily interpreted as the pre-activation dimension. We train NCE with stochastic gradient descent on source-target activity pairs as described in Appendix G. When all nonlinearities are removed, NCE becomes equivalent to Reduced Rank Regression [67]. When predicting stimulus-driven activity, we pair the activity of source and target neurons on different repeats of the same stimulus, to also discard shared variability that is not related to the stimulus [39].

Linear-nonlinear readout. The predicted activity of a set of target neurons at time t, $\mathbf{b}_t \in \mathbb{R}^B$, is a weighted sum of the latent variables, $\mathbf{z}_t \in \mathbb{R}^d$, passed through a nondecreasing nonlinearity:

$$\mathbf{b}_t = \phi_{p,\mathbf{c}}(\mathbf{U}\mathbf{z}_t) + \mathbf{r}. \tag{13}$$

Here, $\phi_{p,\mathbf{c}}$ is the rectified power activation function, defined in Eq. 9, with a power parameter p that is constant across neurons, a pre-activation bias that varies across neurons encoded by an N-dimensional vector $\mathbf{c}$, and a post-activation added bias $\mathbf{r}$ to account for non-zero baseline firing rates. The decoder parameters $\{p, \mathbf{c}, \mathbf{U}, \mathbf{r}\}$ are learned alongside the encoder parameters of $\mathcal{E}$. The fact that the decoder, Eq. (13), has a single-layer is crucial as it allows us to interpret the latents ($\mathbf{z}_t$) as linear factors of the observed neurons' pre-activations ($\mathbf{U}\mathbf{z}_t$). It is this constrained decoder that allows us to infer the pre-activation dimension of neuronal activity; in comparison, a multi-layer decoder as in used in [13, 23] would infer something closer to the intrinsic dimension of neuronal activity, which is not our goal.

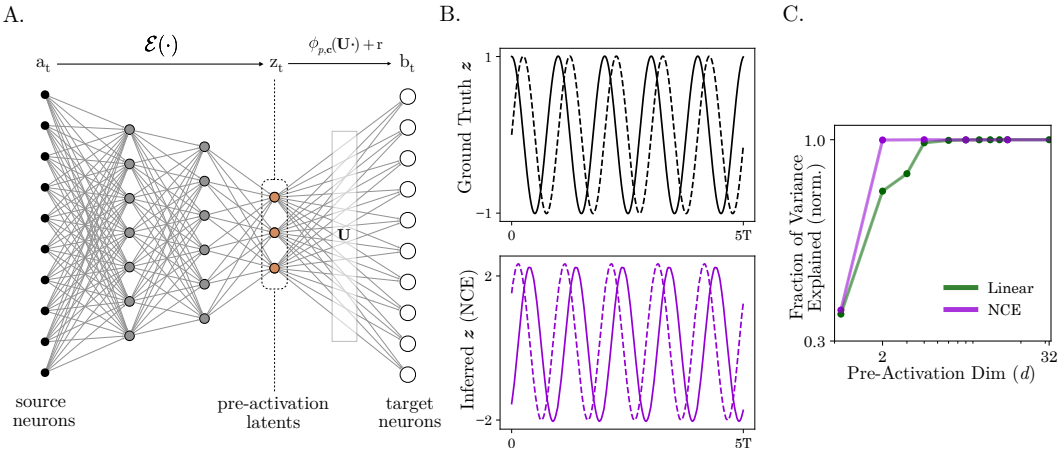

Figure 3: **Validation of the Neural Cross Encoder** (**A**) Schematic of the NCE. **B** Simulated pre-activation latents of a two-dimensional toy model (top), and the inferred pre-activation latents using NCE (bottom). **C** Reconstruction score as a function of pre-activation dimensionality for Linear (RRR) and NCE models on simulated data.

4.3 Results

NCE identifies the latent dimensionality of simulated data. To validate that NCE can identify the pre-activation latent variables, we test it on simulated data generated from the toy model in Sec. 2 with a ReLU readout ($p = 1$) and $d = 2$. NCE recovers the true latents up to a scaling and a shift (Fig 3B). Moreover, NCE can explain all of the variance in the population with only two pre-activation dimensions, while the corresponding linear model (Reduced Rank Regression) requires more dimensions (Fig 3C).

Pre-activation dimension is low for grating responses, high for natural image responses. We first consider the pre-activation dimensional of visual stimulus responses. To ensure that the NCE focused on the stimulus responses, and not correlated ongoing activity such as spontaneous activity or encoding of movements, the the activity of the target cells and the source cells were taken from different repeats of the same stimuli. In the case of drifting gratings, for which we know there is

a low-dimensional latent variable (the grating orientation), an NCE model with low-dimensional pre-activations accurately predicts neuronal responses (Fig. 4A,B). NCE requires fewer dimensions (5.5 ± 1.2, mean $\pm$ std) than the corresponding linear model (13.9 ± 3.0) to predict 95% of the explainable variance (defined as the maximum variance explained across all d for a given model type). On the other hand, NCE models with low pre-activation dimension are not sufficient to predict responses to natural images (Fig. 4C,D), requiring 93.9 ± 6.0 dimensions to reach the threshold, suggesting that natural images produce high-dimensional representations in the space of pre-activations. Linear models only account for a smaller fraction of the total variance (Fig. 4D), and therefore underestimate the dimensionality of natural image responses (48.0 ± 13.1).

Spontaneous activity has low pre-activation dimension. Spontaneous activity is well predicted by NCE with low pre-activation dimension (Fig. 4E,F). Across all recordings, spontaneous activity of 1000 target neurons has an estimated pre-activation dimension d of 7.0 ± 1.0 (mean $\pm$ std.), somewhat larger than grating responses but substantially lower than natural images responses (Fig. 4G,H). The linear model finds a similar dimension (7.5 ± 2.0), though its performance deteriorates at high pre-activation dimensions due to overfitting, while the NCE performance remains consistent (Fig. 4F).

These results indicate that visual cortex activity can be modeled as a linear-nonlinear transformation of a latent vector, which is low-dimensional for grating responses and spontaneous activity, but high-dimensional for natural image responses. In the case of grating responses, we find latents that resemble the sine and cosine of the stimulus angle (Fig. 4A)–this is what one would expect to find if neurons follow the canonical model of simple cells in visual cortex [68]. On the other hand, during spontaneous activity, the dynamics of the latent variables are correlated with the running speed of the mouse, and are perhaps related to its arousal state (see Appendix G.3).

5 Summary of Technical Contributions and Previous Works

Latent dynamics of low-rank RNNs Low-rank RNNs are tractable models of how the brain can perform computations through low-dimensional population dynamics [36, 51, 69–72]. In particular, the dynamics of certain low-rank RNNs, in the large-network limit, reduce to that of "effective circuits", i.e., dynamical systems describing the evolution of the latent variables [51, 52, 73]. A limitation of these effective dynamical systems is that the expression of the vector field involves an integral over the distribution of weights (the "circuit structure" [52]), making them somewhat opaque and costly to solve numerically. In this work (Sec. 2.1 and Appendix C), we prove that, in several special cases, the integral over the weight distribution can be solved, yielding simple exact equations for the latent dynamics.

Eigenvalue decay of random feature kernels In the infinite-width limit, two-layer neural networks with random input weights behave like *random feature kernels* that depend on the distribution of input weights and the activation function of the neurons in the hidden layer [56, Sec. 9.5]. This functional perspective can be generalized to deep networks [45, 74] and constitute the basis of the Neural Tangent Kernel formalism for studying learning dynamics [75]. When the activation function is the ReLU, the decay rate of the eigenvalues has been proven to be polynomial [46, 47, 58, 64, 76, 77] (see [57, 61, 63, 78, 79] for general results on dot-product kernels). In this work, we propose a simple formula that links the power-law exponent of the eigenvalue decay rate, the power of the rectified-power activation function, and the input dimensionality. This formula, which goes beyond known results [46, 47, 64], is presented as a conjecture that we test in simulations.

Latent variable modeling of neuronal activity While most latent variable models of neuronal activity were originally developed for electrophysiological recordings [8–23], some are tailored for calcium recordings [80–82]. These models vary in their mechanistic interpretability: The inferred latents are either abstract variables, for example when the model's mapping from latents to neuronal activity involves a multi-layer neural network [13, 23], or they can be interpreted as linear factors of the neurons' pre-activations, as in [9, 10]. With nonlinear dimensionality reduction methods such as CEBRA [83] or Rastermap [84], latent variables are also abstract as there is no explicit mapping going from the latents to neuronal activity. In this work (Sec. 4), we developed NCE, a latent variable model for calcium recordings that models neuronal activity as an interpretable linear-nonlinear readout of latent variables. NCE also uses a cross-encoding scheme, which allows it to discard variability

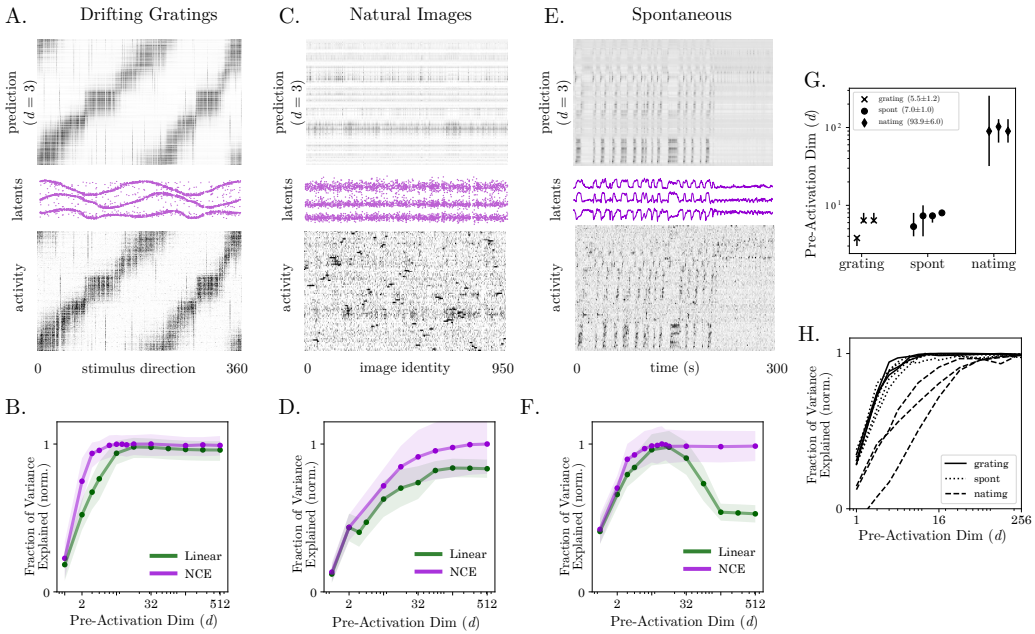

Figure 4: **Pre-activation dimension in visual cortex depends on stimulus condition.** (**A**) Rasters of true (bottom) and predicted (top) neuronal activity, and estimated latents (middle) for 1000 example neurons using an NCE with pre-activation dimension $d = 3$, for drifting grating responses. (**B**) Fraction of variance explained as a function of pre-activation dimension for an example session of grating responses. Plotted values are the mean across source/target selections for NCE (purple) and Reduced Rank Regression (green). Shaded regions are the standard deviation across source/target selections. All experiments contain 500 source and 1000 target neurons, and all scores are computed on a held-out test set. (**C,D**) Same as **A,B** for natural image responses. (**E,F**) Same as **A,B** for spontaneous activity. (**G**) Estimated pre-activation dimension d sufficient to capture 95% of explainable variance in the NCE across the three conditions. Each point is the mean for a single imaging session, error bars indicate the min/max across source/target selections. (**H**) Fraction of variance explained by NCE models with varying pre-activation dimension (log scale) across the three conditions. Each line represents a single imaging session.

not shared across neurons. We demonstrate that NCE is capable of identifying a low-dimensional pre-activation space even when the recorded neuronal activity has high linear dimension.

6 Discussion

Dimensionality of neural systems The solvable RNN model we proposed produces cyclic population dynamics that is low-dimensional in the space of pre-activations, and high-dimensional in the space of post-activations (firing rates). Thus, an RNN can produce trajectories that are simultaneously low-dimensional and high-dimensional, depending on the variables being considered. In this work, we focused on the notion of *linear* dimensionality and adopted an infinite-dimensional Hilbert space formalism borrowed from kernel methods to characterize the linear dimensionality of firing rate trajectories in the large-network limit. Of course, the "intrinsic" dimension of neuronal activity is equal to 1, since the dynamics is periodic; this highlights the important distinction between intrinsic and linear (or "embedding") dimension of neuronal activity (see also [43, 44]). The type of high-dimensional activity our model produces is computationally relevant: It can be exploited by a downstream readout neuron to represent arbitrary periodic functions (see also [85]), or, following the random readout approach of [86, 87], it can be used to represent a Gaussian process prior over periodic functions (see Appendix E).

Two definitions of high-dimensional neuronal activity have been studied in neuroscience. Throughout this work, we define high-dimensional neuronal activity to mean a covariance eigenspectrum with a heavy tail that decays strictly faster that $1/n$ [39]. This definition is well-suited for systems

generating activity whose pairwise correlations do not converge to zero in the limit of large network size. In contrast, random chaotic RNNs—solvable models whose pairwise correlations do converge to zero—produce a different form of high-dimensional activity, where the eigenspectrum decays slower than $1/n$ [88], reflecting noise that is not shared between neurons. While a comprehensive comparison of these two types of high-dimensional activity is beyond the scope of this work, we mention that the latter type is relevant when one wants to study the noisiness of neuronal responses [89].

Pre-activations and subthreshold membrane potentials We developed NCE to disentangle the linear dimension of neuronal activity and of the neurons' pre-activations. We show that even when activity is high-dimensional, it can be well-explained with low-dimensional pre-activations in the case of grating responses and spontaneous activity. From a biological point of view, how should we interpret the pre-activations inferred by NCE? If one assumes that the link between synaptic integration and neuronal firing is well approximated by a simple nonlinear activation function in cortical neurons, as NCE does, one could argue that pre-activations represent estimates of the neurons' synaptic inputs or subthreshold membrane potentials. This is an experimental prediction that large-scale voltage imaging of neuronal populations [90] may make testable in the near future.

Limitations Training NCE, which is a non-convex optimization problem, can be challenging when neural data is limited. Due to experimental constraints, the duration of imaging experiments is limited to $3.5\,\mathrm{h}$, which yields only a few thousand training examples per session for tens of thousands of neurons; thus, optimization can get stuck in local minima. To facilitate training, we limit to fitting only 1000 highly responsive neurons at a time, and use tools such as data augmentation and pretraining as described in Appendix G. The fact that we were not able to accurately predict neuronal responses to natural images with a low-dimensional NCE model does not necessarily exclude the possibility for such a model to exist, and one could possibly find it with a larger training set. Note that this work did not revisit the problem of how to estimate the tail of the shared covariance eigenspectrum from neuronal recordings, which is discussed in [39, 40, 91, 92].

Acknowledgments and Disclosure of Funding

The authors thank Louis Pezon for useful discussions; Michael Krumin, Bex Terry and Charu B. Reddy for experimental support; Kimberly Ren for feedback on the manuscript. This work was funded by UKRI (Frontier Award EP/X022366/1 to MC), BBSRC (grant BB/W019884/1 to MC), the National Institutes of Health BRAIN initiative (grant U01NS126057 to MC), the Wellcome Trust (Investigator Award 223144/Z/21/Z to MC and KDH), and the ERC (101097874 to KDH). MC holds the GlaxoSmithKline / Fight for Sight Chair in Visual Neuroscience. AH is supported by a studentship from the Gatsby Charitable Foundation (GAT3755) and the Wellcome Trust (219627/Z/19/Z). VS is supported by a Royal Society Newton International Fellowship (NIF\R1\231927) and a fellowship from the Swiss National Science Foundation (grant no. 222150).

References

[1] Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487(7405):51–56, 2012.

[2] 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, 2013.

[3] Abigail A Russo, Sean R Bittner, Sean M Perkins, Jeffrey S Seely, Brian M London, Antonio H Lara, Andrew Miri, Najja J Marshall, Adam Kohn, Thomas M Jessell, et al. Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966, 2018.

[4] Evan D Remington, Devika Narain, Eghbal A Hosseini, and Mehrdad Jazayeri. Flexible sensorimotor computations through rapid reconfiguration of cortical dynamics. *Neuron*, 98(5): 1005–1019, 2018.

[5] Krishna V Shenoy, Maneesh Sahani, and Mark M Churchland. Cortical control of arm movements: a dynamical systems perspective. *Annual Review of Neuroscience*, 36(1):337–359, 2013.

[6] Saurabh Vyas, Matthew D Golub, David Sussillo, and Krishna V Shenoy. Computation through neural population dynamics. *Annual Review of Neuroscience*, 43(1):249–275, 2020.

[7] Juan A Gallego, Matthew G Perich, Lee E Miller, and Sara A Solla. Neural manifolds for the control of movement. *Neuron*, 94(5):978–984, 2017.

[8] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of Neurophysiology*, 102(1):614–635, 2009.

[9] Jakob H Macke, Lars Buesing, John P Cunningham, Byron M Yu, Krishna V Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

[10] Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, L. F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, 2018. doi: 10.1038/s41592-018-0109-9.

[11] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. In *International conference on machine learning*, pages 1726–1734. PMLR, 2019.

[12] Joshua Glaser, Matthew Whiteway, John P Cunningham, Liam Paninski, and Scott Linderman. Recurrent switching dynamical systems models for multiple interacting neural populations. In *Advances in Neural Information Processing Systems*, volume 33, pages 14867–14878, 2020.

[13] Ding Zhou and Xue-Xin Wei. Learning identifiable and interpretable latent models of high-dimensional neural activity using pi-VAE. *Advances in neural information processing systems*, 33:7234–7247, 2020.

[14] Timothy D Kim, Thomas Z Luo, Jonathan W Pillow, and Carlos D Brody. Inferring latent dynamics underlying neural population activity via neural differential equations. In *International Conference on Machine Learning*, pages 5551–5561. PMLR, 2021.

[15] Marine Schimel, Ta-Chu Kao, Kristopher T Jensen, and Guillaume Hennequin. iLQR-VAE : control-based learning of input-driven dynamics with applications to neural data. In *International Conference on Learning Representations*, 2022.

[16] Mohammad Reza Keshtkaran, Andrew R Sedler, Raeed H Chowdhury, Raghav Tandon, Diya Basrai, Sarah L Nguyen, Hansem Sohn, Mehrdad Jazayeri, Lee E Miller, and Chethan Pandarinath. A large-scale neural network training framework for generalized estimation of single-trial population dynamics. *Nature Methods*, 19(12):1572–1577, 2022.

[17] Adrian Valente, Jonathan W Pillow, and Srdjan Ostojic. Extracting computational mechanisms from neural data using low-rank RNNs. In *Advances in Neural Information Processing Systems*, volume 35, pages 24072–24086, 2022.

[18] Shuqi Wang, Valentin Schmutz, Guillaume Bellec, and Wulfram Gerstner. Mesoscopic modeling of hidden spiking neurons. In *Advances in Neural Information Processing Systems*, volume 35, pages 23566–23579, 2022.

[19] Mikhail Genkin, Krishna V Shenoy, Chandramouli Chandrasekaran, and Tatiana A Engel. The dynamics and geometry of choice in premotor cortex. *bioRxiv*, 2023.

[20] Timothy Doyeon Kim, Thomas Zhihao Luo, Tankut Can, Kamesh Krishnamurthy, Jonathan W. Pillow, and Carlos D. Brody. Flow-field inference from neural data using deep recurrent networks. *bioRxiv*, 2023. doi: 10.1101/2023.11.14.567136.

[21] Aditi Jha, Diksha Gupta, Carlos Brody, and Jonathan W Pillow. Disentangling the roles of distinct cell classes with cell-type dynamical systems. In *Advances in Neural Information Processing Systems*, volume 37, pages 33668–33690, 2024.

[22] Christopher Langdon and Tatiana A Engel. Latent circuit inference from heterogeneous neural responses during cognitive tasks. *Nature Neuroscience*, pages 1–11, 2025.

[23] Adam Gosztolai, Robert L Peach, Alexis Arnaudon, Mauricio Barahona, and Pierre Vandergheynst. MARBLE: interpretable representations of neural population dynamics using geometric deep learning. *Nature Methods*, pages 1–9, 2025.

[24] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Computation*, 25(3):626–649, 2013.

[25] David Sussillo, Mark M Churchland, Matthew T Kaufman, and Krishna V Shenoy. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, 18(7):1025–1033, 2015.

[26] Guillaume Hennequin, Tim P Vogels, and Wulfram Gerstner. Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, 82(6): 1394–1406, 2014.

[27] Federico Carnevale, Victor de Lafuente, Ranulfo Romo, Omri Barak, and Néstor Parga. Dynamic control of response criterion in premotor cortex during perceptual detection under temporal uncertainty. *Neuron*, 86(4):1067–1077, 2015.

[28] Jonathan A Michaels, Benjamin Dann, and Hansjörg Scherberger. Neural population dynamics during reaching are better explained by a dynamical system than representational tuning. *PLoS Computational Biology*, 12(11):e1005175, 2016.

[29] Alexander Rivkind and Omri Barak. Local dynamics in trained recurrent neural networks. *Physical Review Letters*, 118(25):258101, 2017.

[30] Joao Barbosa, Rémi Proville, Chris C Rodgers, Michael R DeWeese, Srdjan Ostojic, and Yves Boubenec. Early selection of task-relevant features through population gating. *Nature Communications*, 14(1):6837, 2023.

[31] Christopher Langdon, Mikhail Genkin, and Tatiana A Engel. A unifying perspective on neural manifolds and circuits for cognition. *Nature Reviews Neuroscience*, 24(6):363–377, 2023.

[32] Daniel Durstewitz, Georgia Koppe, and Max Ingo Thurm. Reconstructing computational system dynamics from neural data with recurrent neural networks. *Nature Reviews Neuroscience*, 24 (11):693–710, 2023.

[33] Laura N Driscoll, Krishna Shenoy, and David Sussillo. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nature Neuroscience*, 27(7):1349–1363, 2024.

[34] Harsha Gurnani, Weixuan Liu, and Bingni W Brunton. Feedback control of recurrent dynamics constrains learning timescales during motor adaptation. *bioRxiv*, pages 2024–05, 2024.

[35] Ashok Litwin-Kumar and Brent Doiron. Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nature Neuroscience*, 15(11):1498–1505, 2012. doi: 10.1038/nn.3220.

[36] Francesca Mastrogiuseppe and Srdjan Ostojic. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.

[37] Brian DePasquale, David Sussillo, LF Abbott, and Mark M Churchland. The centrality of population-level factors to network computation is demonstrated by a versatile approach for training spiking networks. *Neuron*, 111(5):631–649, 2023.

[38] Valentin Schmutz, Johanni Brea, and Wulfram Gerstner. Emergent rate-based dynamics in duplicate-free populations of spiking neurons. *Physical Review Letters*, 134(1):018401, 2025.

[39] Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris. High-dimensional geometry of population responses in visual cortex. *Nature*, 571(7765): 361–365, 2019.

[40] Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Charu Bai Reddy, Matteo Carandini, and Kenneth D Harris. Spontaneous behaviors drive multidimensional, brainwide activity. *Science*, 364(6437):eaav7893, 2019.

[41] Frederic Lanore, N Alex Cayco-Gajic, Harsha Gurnani, Diccon Coyle, and R Angus Silver. Cerebellar granule cell axons support high-dimensional representations. *Nature Neuroscience*, 24(8):1142–1150, 2021.

[42] Jason Manley, Sihao Lu, Kevin Barber, Jeffrey Demas, Hyewon Kim, David Meyer, Francisca Martínez Traub, and Alipasha Vaziri. Simultaneous, cortex-wide dynamics of up to 1 million neurons reveal unbounded scaling of dimensionality with neuron number. *Neuron*, 112 (10):1694–1709, 2024.

[43] Mehrdad Jazayeri and Srdjan Ostojic. Interpreting neural computations by examining intrinsic and embedding dimensionality of neural activity. *Current Opinion in Neurobiology*, 70:113–120, 2021.

[44] Mark D Humphries. Strong and weak principles of neural dimension reduction. *Neurons, Behavior, Data analysis, and Theory*, 5(2):1–28, 2021.

[45] Youngmin Cho and Lawrence Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

[46] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(19):1–53, 2017.

[47] Alberto Bietti and Francis Bach. Deep equals shallow for ReLU networks in kernel regimes. In *International Conference on Learning Representations*, 2021.

[48] William Skaggs, James Knierim, Hemant Kudrimoti, and Bruce McNaughton. A model of the neural basis of the rat's sense of direction. In *Advances in Neural Information Processing Systems*, volume 7, 1994.

[49] Rani Ben-Yishai, R Lev Bar-Or, and Haim Sompolinsky. Theory of orientation tuning in visual cortex. *Proceedings of the National Academy of Sciences*, 92(9):3844–3848, 1995.

[50] Kechen Zhang. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *Journal of Neuroscience*, 16(6):2112–2126, 1996.

[51] Manuel Beiran, Alexis Dubreuil, Adrian Valente, Francesca Mastrogiuseppe, and Srdjan Ostojic. Shaping dynamics with multiple populations in low-rank recurrent networks. *Neural Computation*, 33(6):1572–1615, 2021.

[52] Louis Pezon, Valentin Schmutz, and Wulfram Gerstner. Linking neural manifolds to circuit structure in recurrent networks. *bioRxiv*, pages 2024–02, 2024.

[53] Marc Mézard, Giorgio Parisi, and Anthony Zee. Spectra of euclidean random matrices. *Nuclear Physics B*, 559(3):689–701, 1999.

[54] Vladimir Koltchinskii and Evarist Giné. Random matrix approximation of spectra of integral operators. *Bernoulli*, 6(1):113–167, 2000. ISSN 1350-7265,1573-9759. doi: 10.2307/3318636.

[55] Charles Bordenave. Eigenvalues of euclidean random matrices. *Random Structures & Algorithms*, 33(4):515–532, 2008. doi: https://doi.org/10.1002/rsa.20228.

[56] Francis Bach. *Learning theory from first principles*. MIT press, 2024.

[57] Meyer Scetbon and Zaid Harchaoui. A spectral analysis of dot-product kernels. In *International Conference on Artificial Intelligence and Statistics*, pages 3394–3402. PMLR, 2021.

[58] Yicheng Li, Zixiong Yu, Guhan Chen, and Qian Lin. On the eigenvalue decay rates of a class of neural-network related kernel functions defined on general domains. *Journal of Machine Learning Research*, 25(82):1–47, 2024.

[59] Thomas Kühn. Eigenvalues of integral operators with smooth positive definite kernels. *Archiv der Mathematik*, 49:525–534, 1987.

[60] M Castro and V Menegatto. Eigenvalue decay of positive integral operators on the sphere. *Mathematics of Computation*, 81(280):2303–2317, 2012.

[61] Douglas Azevedo and Valdir Antonio Menegatto. Sharp estimates for eigenvalues of integral operators generated by dot product kernels on the sphere. *Journal of Approximation Theory*, 177:57–68, 2014.

[62] Thaís Jordão and V Menegatto. Estimates for fourier sums and eigenvalues of integral operators via multipliers on the sphere. *Proceedings of the American Mathematical Society*, 144(1): 269–283, 2016.

[63] Blake Bordelon and Cengiz Pehlevan. Population codes enable learning from few examples by shaping inductive bias. *Elife*, 11:e78606, 2022.

[64] Marjorie Xie, Samuel P Muscinelli, Kameron Decker Harris, and Ashok Litwin-Kumar. Task-dependent optimal representations for cerebellar learning. *Elife*, 12:e82914, 2023.

[65] Jeffrey Demas, Jason Manley, Frank Tejera, Kevin Barber, Hyewon Kim, Francisca Martínez Traub, Brandon Chen, and Alipasha Vaziri. High-speed, cortex-wide volumetric recording of neuroactivity at cellular resolution using light beads microscopy. *Nature Methods*, 18(9): 1103–1111, 2021.

[66] Ali Haydaroğlu, Tinya Chang, Andrew Landau, Michael Krumin, Sam Dodgson, Liad J. Baruchin, Maria Cozan, Jingkun Guo, David Meyer, Charu Bai Reddy, Jian Zhong, Na Ji, Sylvia Schröder, Kenneth D. Harris, Alipasha Vaziri, and Matteo Carandini. Suite3D: Volumetric cell detection for two-photon microscopy. *bioRxiv*, page 2025.03.26.645628, 2025.

[67] Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5(2):248–264, 1975.

[68] Matteo Carandini. What simple and complex cells compute. *The Journal of Physiology*, 577(Pt 2):463–466, December 2006. ISSN 0022-3751. doi: 10.1113/jphysiol.2006.118976.

[69] Friedrich Schuessler, Alexis Dubreuil, Francesca Mastrogiuseppe, Srdjan Ostojic, and Omri Barak. Dynamics of random recurrent networks with correlated low-rank structure. *Physical Review Research*, 2(1):013111, 2020.

[70] Alexis Dubreuil, Adrian Valente, Manuel Beiran, Francesca Mastrogiuseppe, and Srdjan Ostojic. The role of population structure in computations through neural dynamics. *Nature Neuroscience*, 25(6):783–794, 2022.

[71] Yue Wan and Robert Rosenbaum. High-dimensional dynamics in low-dimensional networks. *arXiv preprint arXiv:2504.13727*, 2025.

[72] Francesca Mastrogiuseppe, Joana Carmona, and Christian Machens. Stochastic activity in low-rank recurrent neural networks. *bioRxiv*, pages 2025–04, 2025.

[73] Romain Veltz and Olivier Faugeras. Local/global analysis of the stationary solutions of some neural field equations. *SIAM Journal on Applied Dynamical Systems*, 9(3):954–998, 2010.

[74] Jaehoon Lee, Jascha Sohl-dickstein, Jeffrey Pennington, Roman Novak, Sam Schoenholz, and Yasaman Bahri. Deep neural networks as gaussian processes. In *International Conference on Learning Representations*, 2018.

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

[76] Amnon Geifman, Abhay Yadav, Yoni Kasten, Meirav Galun, David Jacobs, and Basri Ronen. On the similarity between the laplace and neural tangent kernels. In *Advances in Neural Information Processing Systems*, volume 33, pages 1451–1461, 2020.

[77] Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[78] Alex Smola, Zoltán Ovári, and Robert C Williamson. Regularization with dot-product kernels. In *Advances in Neural Information Processing Systems*, volume 13, 2000.

[79] Ha Quang Minh, Partha Niyogi, and Yuan Yao. Mercer's theorem, feature maps, and smoothing. In *International Conference on Computational Learning Theory*, pages 154–168. Springer, 2006.

[80] Luke Y Prince, Shahab Bakhtiari, Colleen J Gillon, and Blake A Richards. Parallel inference of hierarchical latent dynamics in two-photon calcium imaging of neuronal populations. *bioRxiv*, pages 2021–03, 2021.

[81] Feng Zhu, Harrison A Grier, Raghav Tandon, Changjia Cai, Anjali Agarwal, Andrea Giovannucci, Matthew T Kaufman, and Chethan Pandarinath. A deep learning framework for inference of single-trial neural population dynamics from calcium imaging with subframe temporal resolution. *Nature Neuroscience*, 25(12):1724–1734, 2022.

[82] Tze Hui Koh, William E Bishop, Takashi Kawashima, Brian B Jeon, Ranjani Srinivasan, Yu Mu, Ziqiang Wei, Sandra J Kuhlman, Misha B Ahrens, Steven M Chase, et al. Dimensionality reduction of calcium-imaged neuronal population activity. *Nature Computational Science*, 3(1): 71–85, 2023.

[83] Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, May 2023.

[84] Carsen Stringer, Lin Zhong, Atika Syeda, Fengtong Du, Maria Kesa, and Marius Pachitariu. Rastermap: a discovery method for neural population recordings. *Nature Neuroscience*, 28(1): 201–212, 2025.

[85] Soledad Gonzalo Cogno, Horst A Obenhaus, Ane Lautrup, R Irene Jacobsen, Claudia Clopath, Sebastian O Andersson, Flavio Donato, May-Britt Moser, and Edvard I Moser. Minute-scale oscillatory sequences in medial entorhinal cortex. *Nature*, 625(7994):338–344, 2024.

[86] Radford M Neal. Priors for infinite networks. *Bayesian learning for neural networks*, pages 29–53, 1996.

[87] Christopher Williams. Computing with infinite networks. In *Advances in Neural Information Processing Systems*, volume 9, 1996.

[88] David G Clark, LF Abbott, and Ashok Litwin-Kumar. Dimension of activity in random neural networks. *Physical Review Letters*, 131(11):118401, 2023.

[89] Gengshuo John Tian, Ou Zhu, Vinay Shirhatti, Charles M Greenspon, John E Downey, David J Freedman, and Brent Doiron. Neuronal firing rate diversity lowers the dimension of population covariability. *bioRxiv*, pages 2024–08, 2024.

[90] Jian Zhong, Ryan G. Natan, Qinrong Zhang, Justin S. J. Wong, Christoph Miehl, Krishnashish Bose, Xiaoyu Lu, François St-Pierre, Su Guo, Brent Doiron, Kevin K. Tsia, and Na Ji. Faced 2.0 enables large-scale voltage and calcium imaging in vivo. page 2025.03.06.641784, 2025.

[91] Dean A. Pospisil and Jonathan W. Pillow. Revisiting the high-dimensional geometry of population responses in visual cortex. *bioRxiv*, 2024. doi: 10.1101/2024.02.16.580726.

[92] Marius Pachitariu, Lin Zhong, Alexa Gracias, Amanda Minisi, Crystall Lopez, and Carsen Stringer. A critical initialization for biological neural networks. *bioRxiv*, pages 2025–01, 2025.

[93] Tsai-Wen Chen, Trevor J. Wardill, Yi Sun, Stefan R. Pulver, Sabine L. Renninger, Amy Baohan, Eric R. Schreiter, Rex A. Kerr, Michael B. Orger, Vivek Jayaraman, Loren L. Looger, Karel Svoboda, and Douglas S. Kim. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, 2013.

[94] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv*, (arXiv:1503.02531), 2015. arXiv:1503.02531.

[95] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, (arXiv:1412.6980), 2017. arXiv:1412.6980.

[96] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, page 1026–1034. IEEE, 2015.

Appendix

A Derivation of the latent dynamics Eq. (5) from Eq. (4)

To go from Eq. (4) to Eq. (5), we only have to solve the integral

$$\int_{\theta=0}^{2\pi} \mathbf{v}(\theta)\phi(\mathbf{u}(\theta)\cdot\boldsymbol{\kappa})\frac{\mathrm{d}\theta}{2\pi},$$

with $\mathbf{u}(\theta) = (\cos\theta, \sin\theta)^{\mathrm{T}}$ and $\mathbf{v}(\theta) = J(\cos(\theta+\Delta), \sin(\theta+\Delta))^{\mathrm{T}}$. Using the polar change of coordinate

$$\boldsymbol{\kappa} = \|\boldsymbol{\kappa}\| \begin{pmatrix} \cos\vartheta \\ \sin\vartheta \end{pmatrix}, \tag{14}$$

we have

$$\begin{aligned}\int_{\theta=0}^{2\pi} \mathbf{v}(\theta)\phi(\mathbf{u}(\theta)\cdot\boldsymbol{\kappa})\frac{\mathrm{d}\theta}{2\pi} &= J\int_{\theta=0}^{2\pi} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \phi\left(\|\boldsymbol{\kappa}\| \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \cdot \begin{pmatrix} \cos\vartheta \\ \sin\vartheta \end{pmatrix}\right) \frac{\mathrm{d}\theta}{2\pi} \\ &= J\int_{\theta=0}^{2\pi} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \phi\left(\|\boldsymbol{\kappa}\|\cos(\vartheta-\theta)\right) \frac{\mathrm{d}\theta}{2\pi}.\end{aligned}$$

Recalling that ϕ is the Heaviside step function Θ, we have

$$\begin{aligned}J\int_{\theta=0}^{2\pi} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \phi\left(\|\boldsymbol{\kappa}\|\cos(\vartheta-\theta)\right) \frac{\mathrm{d}\theta}{2\pi} &= J\int_{\theta=0}^{2\pi} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \Theta\left(\|\boldsymbol{\kappa}\|\cos(\vartheta-\theta)\right) \frac{\mathrm{d}\theta}{2\pi} \\ &= \frac{J}{2\pi}\int_{\theta=\vartheta-\pi/2}^{\vartheta+\pi/2} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \mathrm{d}\theta.\end{aligned}$$

The integral above can be solved, which yields

$$\frac{J}{2\pi}\int_{\theta=\vartheta-\pi/2}^{\vartheta+\pi/2} \begin{pmatrix} \cos(\theta+\Delta) \\ \sin(\theta+\Delta) \end{pmatrix} \mathrm{d}\theta = \frac{J}{2\pi}\begin{bmatrix} \sin(\theta+\Delta) \\ -\cos(\theta+\Delta) \end{bmatrix}_{\theta=\vartheta-\pi/2}^{\vartheta+\pi/2} = \frac{J}{\pi}\begin{pmatrix} \sin(\vartheta+\frac{\pi}{2}+\Delta) \\ -\cos(\vartheta+\frac{\pi}{2}+\Delta) \end{pmatrix}.$$

Using the sum and difference trigonometric identities,

$$\begin{aligned}\sin(\vartheta+\tfrac{\pi}{2}+\Delta) &= \sin(\tfrac{\pi}{2}+\Delta)\cos\vartheta + \cos(\tfrac{\pi}{2}+\Delta)\sin\vartheta, \\ -\cos(\vartheta+\tfrac{\pi}{2}+\Delta) &= -\cos(\tfrac{\pi}{2}+\Delta)\cos\vartheta + \sin(\tfrac{\pi}{2}+\Delta)\sin\vartheta,\end{aligned}$$

we can write

$$\frac{J}{\pi}\begin{pmatrix} \sin(\vartheta+\frac{\pi}{2}+\Delta) \\ -\cos(\vartheta+\frac{\pi}{2}+\Delta) \end{pmatrix} = \frac{J}{\pi}\begin{pmatrix} \sin(\frac{\pi}{2}+\Delta) & \cos(\frac{\pi}{2}+\Delta) \\ -\cos(\frac{\pi}{2}+\Delta) & \sin(\frac{\pi}{2}+\Delta) \end{pmatrix} \underbrace{\begin{pmatrix} \cos\vartheta \\ \sin\vartheta \end{pmatrix}}_{=\boldsymbol{\kappa}/\|\boldsymbol{\kappa}\|}$$

where we revert the polar change of coordinate Eq. (14). Plugging in $J = \pi\sqrt{2}$ and $\Delta = \pi/4$, we obtain

$$\frac{J}{\pi}\begin{pmatrix} \sin(\frac{\pi}{2}+\Delta) & \cos(\frac{\pi}{2}+\Delta) \\ -\cos(\frac{\pi}{2}+\Delta) & \sin(\frac{\pi}{2}+\Delta) \end{pmatrix} \frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|} = \sqrt{2}\begin{pmatrix} \sin(\frac{3\pi}{4}) & \cos(\frac{3\pi}{4}) \\ -\cos(\frac{3\pi}{4}) & \sin(\frac{3\pi}{4}) \end{pmatrix} \frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|} = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|}.$$

B Derivation of pairwise correlations Eq. (6)

First, we compute the time-average and the standard deviation the post-activation of neuron i,

$$\phi\left(\begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} \cdot \boldsymbol{\kappa}\right),$$

assuming that the latent variables $\boldsymbol{\kappa}$ rotate at constant speed on the unit circle (which is the asymptotic behavior of the dynamical system Eq. (5)). Using again the polar change of coordinate Eq. (14) and recalling that $\phi = \Theta$ (the Heaviside step function), we have

$$\phi\left(\begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} \cdot \boldsymbol{\kappa}\right) = \Theta\left(\begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} \cdot \begin{pmatrix} \cos\vartheta \\ \sin\vartheta \end{pmatrix}\right) = \Theta(\cos(\theta_i - \vartheta)).$$

The time-average of the post-activation of the neuron i is

$$\bar{r}_i := \int_{\vartheta=0}^{2\pi} \Theta(\cos(\theta_i - \vartheta)) \frac{\mathrm{d}\vartheta}{2\pi} = \int_{\vartheta=\theta_i-\pi/2}^{\theta_i+\pi/2} \frac{\mathrm{d}\vartheta}{2\pi} = \frac{1}{2}. \tag{15}$$

Similarly, the variance the post-activation of neuron i is

$$\sigma_i^2 := \int_{\vartheta=0}^{2\pi} \Theta(\cos(\theta_i - \vartheta))^2 \frac{\mathrm{d}\vartheta}{2\pi} - \bar{r}_i^2 = \int_{\vartheta=\theta_i-\pi/2}^{\theta_i+\pi/2} \frac{\mathrm{d}\vartheta}{2\pi} - \frac{1}{4} = \frac{1}{4}. \tag{16}$$

The correlation between any pair of neurons i and j is

$$C_{ij} := \frac{1}{\sigma_i \sigma_j} \int_{\vartheta=0}^{2\pi} [\Theta(\cos(\theta_i - \vartheta)) - \bar{r}_i][\Theta(\cos(\theta_j - \vartheta)) - \bar{r}_j] \frac{\mathrm{d}\vartheta}{2\pi}. \tag{17}$$

Developing the product in the integrand above, we see that it only remains to compute the integral

$$\int_{\vartheta=0}^{2\pi} \Theta(\cos(\theta_i - \vartheta))\Theta(\cos(\theta_j - \vartheta)) \frac{\mathrm{d}\vartheta}{2\pi}.$$

Using the change of variable $\vartheta = \theta_i + \frac{\pi}{2} - \tilde{\vartheta}$, we have

$$\begin{aligned}\int_{\vartheta=0}^{2\pi} \Theta(\cos(\theta_i - \vartheta))\Theta(\cos(\theta_j - \vartheta)) \frac{\mathrm{d}\vartheta}{2\pi} &= \int_{\tilde{\vartheta}=0}^{2\pi} \Theta(\cos(\tilde{\vartheta} - \tfrac{\pi}{2}))\Theta(\cos(\tilde{\vartheta} + \theta_j - \theta_i - \tfrac{\pi}{2})) \frac{\mathrm{d}\tilde{\vartheta}}{2\pi} \\ &= \int_{\tilde{\vartheta}=0}^{\pi} \Theta(\cos(\tilde{\vartheta} + \theta_j - \theta_i - \tfrac{\pi}{2})) \frac{\mathrm{d}\tilde{\vartheta}}{2\pi}.\end{aligned}$$

Without loss of generality, we assume that $\theta_j - \theta_i \geq 0$ (otherwise, permute the indices j and i). We then distinguish two cases: If $\theta_j - \theta_i \leq \pi$,

$$\int_{\tilde{\vartheta}=0}^{\pi} \Theta(\cos(\tilde{\vartheta} + \theta_j - \theta_i - \tfrac{\pi}{2})) \frac{\mathrm{d}\tilde{\vartheta}}{2\pi} = \int_{\tilde{\vartheta}=\theta_j-\theta_i}^{\pi} \frac{\mathrm{d}\tilde{\vartheta}}{2\pi} = \pi - \theta_j - \theta_i,$$

if $\theta_j - \theta_i > \pi$,

$$\int_{\tilde{\vartheta}=0}^{\pi} \Theta(\cos(\tilde{\vartheta} + \theta_j - \theta_i - \tfrac{\pi}{2})) \frac{\mathrm{d}\tilde{\vartheta}}{2\pi} = \int_{\tilde{\vartheta}=0}^{\theta_j-\theta_i-\pi} \frac{\mathrm{d}\tilde{\vartheta}}{2\pi} = \theta_j - \theta_i - \pi.$$

However, by the definition of the absolute angle difference $|\theta_i - \theta_j| := \cos^{-1}(\cos(\theta_i - \theta_j))$, we know that

$$\pi - |\theta_j - \theta_i| = \begin{cases} \pi - \theta_j - \theta_i, & \text{if } \theta_j - \theta_i \leq \pi, \\ \theta_j - \theta_i - \pi, & \text{if } \theta_j - \theta_i > \pi. \end{cases}$$

Hence,

$$\int_{\vartheta=0}^{2\pi} \Theta(\cos(\theta_i - \vartheta))\Theta(\cos(\theta_j - \vartheta)) \frac{\mathrm{d}\vartheta}{2\pi} = \pi - |\theta_j - \theta_i|, \tag{18}$$

and we can verify that the equality above is invariant upon permutation of the indices i and j. Using Eqs. (15), (16), and (18) in Eq. (17), we obtain

$$C_{ij} = \frac{2}{\pi} \left(\pi - |\theta_i - \theta_j|\right) - 1.$$

C Random low-rank RNNs with tractable latent dynamics

The model analyzed in Sec. 2 is not the only model for which a simple expression for the latent dynamics can be obtained. To illustrate this fact, we re-use the general definition of RNN dynamics, Eq. (1), but consider here the class of low-rank RNNs with rank-d weights matrices $\mathbf{W}$ given by

$$\mathbf{W} = \mathbf{U}\,\mathbf{M}\,\mathbf{U}^{\mathrm{T}}, \tag{19}$$

where $\mathbf{U}$ is an $N \times d$ random matrix with i.i.d. standard normal entries and $\mathbf{M} \in \mathbb{R}^{d \times d}$ is an arbitrary "overlap" matrix [51]. As explained in [51], thanks to the fact that the "patterns" $\mathbf{U}$ are

Gaussian, the dynamics of the d-dimensional latent variable vector $\boldsymbol{\kappa} := \mathbf{U}^\dagger \mathbf{x}$ can be described, in the large-network limit $N \to \infty$, by the dynamical system

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \mathbb{E}_{Y\sim\mathcal{N}(0,1)}\left[Y\phi(Y\|\boldsymbol{\kappa}\|)\right]\mathbf{M}\frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|}. \tag{20}$$

In the present work, we go one step further and ask whether the expectation $\mathbb{E}_{Y\sim\mathcal{N}(0,1)}\left[Y\phi(Y\|\boldsymbol{\kappa}\|)\right]$ can be solved, in order to yield simpler expressions for the latent dynamics. The following proposition shows that the answer is positive when the activation function ϕ is either (i) a step function, (ii) a ReLU function, or (iii) a Gaussian cumulative distribution function

Proposition 1. *(i) For any bias $b \in \mathbb{R}$, if ϕ is the step function $\phi = \Theta(x+b)$, Eq.* (20) *reduces to*

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \frac{1}{\sqrt{2\pi}}\exp\left(-\tfrac{b^2}{2\|\boldsymbol{\kappa}\|^2}\right)\mathbf{M}\frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|}.$$

(ii) For any bias $b \in \mathbb{R}$, if ϕ is the ReLU function $\phi = \max(0, x+b)$, Eq. (20) *reduces to*

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \frac{1}{2}\left(1 + \operatorname{erf}\left(\tfrac{b}{\sqrt{2}\|\boldsymbol{\kappa}\|}\right)\right)\mathbf{M}\boldsymbol{\kappa}. \tag{21}$$

(iii) For any mean $\mu \in \mathbb{R}$ and variance $\sigma^2 \geq 0$, if ϕ is the Gaussian c.d.f. $\phi(x) = \frac{1}{\sqrt{2\pi}\sigma}\int_{t=-\infty}^{x}\exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right)\mathrm{d}t$, Eq. (20) *reduces to*

$$\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \frac{e^{-\mu^2/(2(\sigma^2+\|\boldsymbol{\kappa}\|^2))}}{\sqrt{2\pi(\sigma^2+\|\boldsymbol{\kappa}\|^2)}}\mathbf{M}\frac{\boldsymbol{\kappa}}{\|\boldsymbol{\kappa}\|}.$$

The proposition shows the nontrivial effect of the activation function has on the latent dynamics. For example, let us take again $d = 2$. When ϕ is the step function with bias $b = 0$, and if $\mathbf{M} = \sqrt{2\pi}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}$, we recover the latent dynamics Eq. (5), which generates a stable limit cycle. In contrast, when ϕ is the ReLU function with bias $b = 0$, Eq. (21) becomes the linear system $\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \mathbf{M}\boldsymbol{\kappa}$, which can generate rotational dynamics but cannot generate a stable limit cycle.

Proof. The prove these statements, it suffices to solve the integral,

$$\mathbb{E}_{Y\sim\mathcal{N}(0,1)}\left[Y\phi(Y\|\boldsymbol{\kappa}\|)\right] = \int_{y=-\infty}^{\infty} y\,\phi(y\|\boldsymbol{\kappa}\|)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y,$$

for the three different functions ϕ.

For *(i)*,

$$\int_{y=-\infty}^{\infty} y\,\Theta(y\|\boldsymbol{\kappa}\|+b)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y = \int_{y=-\frac{b}{\|\boldsymbol{\kappa}\|}}^{\infty} y\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y = \frac{1}{\sqrt{2\pi}}\exp\left(-\tfrac{b^2}{2\|\boldsymbol{\kappa}\|^2}\right).$$

For *(ii)*,

$$\int_{y=-\infty}^{\infty} y\max(0, y\|\boldsymbol{\kappa}\|+b)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y = \int_{y=-\frac{b}{\|\boldsymbol{\kappa}\|}}^{\infty} y(y\|\boldsymbol{\kappa}\|+b)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y,$$

and we integrate by parts,

$$\begin{aligned}
&\int_{y=-\frac{b}{\|\boldsymbol{\kappa}\|}}^{\infty} y(y\|\boldsymbol{\kappa}\|+b)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y\\
&\quad= \underbrace{\left[-(y\|\boldsymbol{\kappa}\|+b)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\right]_{y=-\frac{b}{\|\boldsymbol{\kappa}\|}}^{\infty}}_{=0} + \|\boldsymbol{\kappa}\|\int_{y=-\frac{b}{\|\boldsymbol{\kappa}\|}}^{\infty}\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\mathrm{d}y\\
&\quad= \frac{1}{2}\left(1+\operatorname{erf}\left(\tfrac{b}{\sqrt{2}\|\boldsymbol{\kappa}\|}\right)\right)\|\boldsymbol{\kappa}\|.
\end{aligned}$$

For *(iii)*, we first integrate by parts,

$$\int_{y=-\infty}^{\infty} y\,\phi(y\|\boldsymbol{\kappa}\|)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\,\mathrm{d}y = \int_{y=-\infty}^{\infty} \phi'(y\|\boldsymbol{\kappa}\|)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\,\mathrm{d}y$$
$$= \int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y\|\boldsymbol{\kappa}\|-\mu)^2}{2\sigma^2}\right)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\,\mathrm{d}y.$$

Then, we complete the square,

$$\int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y\|\boldsymbol{\kappa}\|-\mu)^2}{2\sigma^2}\right)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\,\mathrm{d}y$$
$$= \frac{1}{\sqrt{2\pi}}\int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\|\boldsymbol{\kappa}\|^2y^2-2\mu\|\boldsymbol{\kappa}\|y+\mu^2+\sigma^2y^2}{2\sigma^2}\right)\mathrm{d}y$$
$$= \frac{1}{\sqrt{2\pi}}\int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\sigma^2+\|\boldsymbol{\kappa}\|^2}{2\sigma^2}\left(y-\frac{\mu\|\boldsymbol{\kappa}\|}{\sigma^2+\|\boldsymbol{\kappa}\|^2}\right)^2+\frac{\mu^2\|\boldsymbol{\kappa}\|^2}{2\sigma^2(\sigma^2+\|\boldsymbol{\kappa}\|^2)}-\frac{\mu^2}{2\sigma^2}\right)\mathrm{d}y$$
$$= \frac{1}{\sqrt{2\pi(\sigma^2+\|\boldsymbol{\kappa}\|^2)}}\underbrace{\int_{y=-\infty}^{\infty}\frac{\sqrt{\sigma^2+\|\boldsymbol{\kappa}\|^2}}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\sigma^2+\|\boldsymbol{\kappa}\|^2}{2\sigma^2}\left(y-\frac{\mu\|\boldsymbol{\kappa}\|}{\sigma^2+\|\boldsymbol{\kappa}\|^2}\right)^2\right)\mathrm{d}y}_{=1}$$
$$\times\exp\left(-\frac{\mu^2}{2(\sigma^2+\|\boldsymbol{\kappa}\|^2)}\right).$$

□

D From the eigenvalues of the activity matrix $\mathbf{A}^{(N,T)}$ to the eigenvalues of the random feature kernel operator Eq. (10)

First, we recall that the eigenvalues $\lambda_1^{(N,T)} \geq \lambda_2^{(N,T)} \geq \dots$ of the "covariance matrix",

$$\mathbf{C}^{(N,T)} := \frac{1}{T}\mathbf{A}^{(N,T)}(\mathbf{A}^{(N,T)})^{\mathrm{T}},$$

are related to the eigenvalues $\eta_1^{(N,T)} \geq \eta_2^{(N,T)} \geq \dots$ of the "Gram matrix",

$$\mathbf{G}^{(N,T)} := \frac{1}{N}(\mathbf{A}^{(N,T)})^{\mathrm{T}}\mathbf{A}^{(N,T)}.$$

Indeed, using the singular value decomposition of $\mathbf{A}^{(N,T)}$, one can easily verify that

$$\frac{\lambda_n^{(N,T)}}{N} = \frac{\eta_n^{(N,T)}}{T}, \qquad \forall\, n \in [\![1, \min(N,T)]\!].$$

By the law of large numbers, in the limit $N \to \infty$, the entries of the Gram matrix $\mathbf{G}^{(N,T)}$ converge to an expectation corresponding to the random feature kernel Eq. (11): For any $1 \leq s, t \leq T$,

$$\mathbf{G}_{s,t}^{(N,T)} = \frac{1}{N}\sum_{i=1}^{N}\phi_{p,c}(\mathbf{u}_i\mathbf{z}_s)\phi_{p,c}(\mathbf{u}_i\mathbf{z}_s) \xrightarrow[N\to\infty]{a.s.} \mathbb{E}_{\boldsymbol{\xi}\sim\mathcal{N}(0,1)}[\phi_{p,c}(\boldsymbol{\xi}\cdot\mathbf{z}_s)\phi_{p,c}(\boldsymbol{\xi}\cdot\mathbf{z}_t)] = K_{p,c,d}(\mathbf{z}_s,\mathbf{z}_t),$$

where $\mathbf{u}_i$ above denotes the i-th row of the $N \times d$ random matrix $\mathbf{U}$. Hence, as $N \to \infty$, the eigenvalues $\eta_n^{(N,T)}$ of the Gram matrix $\mathbf{G}^{(N,T)}$ converge to the eigenvalues $\eta_n^{(T)}$ of the kernel matrix

$$\mathbf{K}^{(T)} := (K_{p,c,d}(\mathbf{z}_s,\mathbf{z}_t))_{1\leq s,t\leq T}.$$

Furthermore, we have that

$$\lim_{N\to\infty}\frac{\lambda_n^{(N,T)}}{N} = \frac{\eta_n^{(T)}}{T}.$$

Finally, by a theorem from Koltchinskii and Giné [54, Theorem 3.1], the scaled eigenvalues $\eta_n^{(T)}/T$ of the of the kernel matrix $\mathbf{K}^{(T)}$ converge (in a ℓ_2 sense) to the eigenvalues λ_n of the integral operator Eq. (10). In conclusion, we have that, for any given n,

$$\lim_{T\to\infty}\lim_{N\to\infty}\frac{\lambda_n^{(N,T)}}{N} = \lambda_n.$$

E Gaussian processes from RNNs

As in Appendix B, we start from an infinite-size network where the latent variables $\boldsymbol{\kappa}$ rotate at constant speed on the unit circle. Using again the change of coordinate Eq. (14), the post-activation (firing rate) of a neuron with location θ_i is

$$\Theta(\cos(\theta_i - \vartheta)),$$

where ϑ is the angle of the latent variable $\boldsymbol{\kappa}$. Also, from Eq. (5), one can easily deduce that the norm of the velocity vector is always $\sqrt{\cos(\vartheta)^2 + \sin(\vartheta)^2} = 1$, which implies that the cycle time period of the system is 2π.

Taking M independently and uniformly sampled neurons on the circle, we consider the random projection readout

$$g^{(M)}(\vartheta) := \frac{1}{\sqrt{M}} \sum_{i=1}^{M} \tilde{w}_i \Theta(\cos(\theta_i - \vartheta)),$$

where the readout weights $\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_M$ are *i.i.d.* standard normal variables that are also independent from the random angles θ_i. Since the readout $g^{(M)}$ is a weighted sum of 2π-periodic functions, $g^{(M)}$ is itself a 2π-periodic function.

As the number of randomly sampled neurons M tends to infinity, the random readout $g^{(M)}$ converges in law to a Gaussian process on the circle $\mathbb{S}^1$. To see this, we first compute the mean and covariance of the 2π-periodic function $g^{(M)}$, as $M \to \infty$. For the mean, we have

$$\mathbb{E}\left[\frac{1}{\sqrt{M}} \sum_{i=1}^{M} \tilde{w}_i \Theta(\cos(\theta_i - \vartheta))\right] = \frac{1}{\sqrt{M}} \sum_{i=1}^{M} \underbrace{\mathbb{E}\left[\tilde{w}_i\right]}_{=0} \mathbb{E}\left[\Theta(\cos(\theta_i - \vartheta))\right] = 0,$$

for any M. For the covariance, we have, we have, for any pair of angles ϑ and ϑ',

$$\mathbb{E}\left[\frac{1}{\sqrt{M}} \sum_{i=1}^{M} \tilde{w}_i \Theta(\cos(\theta_i - \vartheta)) \frac{1}{\sqrt{M}} \sum_{i=1}^{M} \tilde{w}_i \Theta(\cos(\theta_i - \vartheta'))\right] = \frac{1}{M} \sum_{i=1}^{M} \Theta(\cos(\theta_i - \vartheta))\Theta(\cos(\theta_i - \vartheta')),$$

using that $E[\tilde{w}_i{}^2] = 1$ and $E[\tilde{w}_i \tilde{w}_j] = 0$ when $i \neq j$. By the law of large numbers,

$$\frac{1}{M} \sum_{i=1}^{M} \Theta(\cos(\theta_i - \vartheta))\Theta(\cos(\theta_i - \vartheta')) \xrightarrow[M\to\infty]{a.s.} \mathbb{E}_{\theta\sim\mathcal{U}([0,2\pi[)}[\Theta(\theta - \vartheta)\Theta(\theta - \vartheta')],$$

and using the same line of arguments we used to derive Eq. (18) in Appendix B, we obtain

$$\mathbb{E}_{\theta\sim\mathcal{U}([0,2\pi[)}[\Theta(\theta - \vartheta)\Theta(\theta - \vartheta')] = \pi - |\vartheta - \vartheta'| = k_0(\vartheta, \vartheta').$$

Then, by the multivariate central limit theorem, we have that, for any finite sequence of angles $\vartheta_1, \vartheta_2, \ldots, \vartheta_m$,

$$(g^{(M)}(\vartheta_1), g^{(M)}(\vartheta_2), \ldots, g^{(M)}(\vartheta_m))^{\mathrm{T}} \xrightarrow[M\to\infty]{\mathcal{L}} \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \tag{22}$$

where

$$\boldsymbol{\Sigma} := \begin{pmatrix} \pi & \pi - |\vartheta_1 - \vartheta_2| & \ldots & \pi - |\vartheta_1 - \vartheta_m| \\ \pi - |\vartheta_2 - \vartheta_1| & \pi & \ldots & \pi - |\vartheta_2 - \vartheta_m| \\ \vdots & \vdots & \ddots & \vdots \\ \pi - |\vartheta_m - \vartheta_1| & \pi - |\vartheta_m - \vartheta_2| & \ldots & \pi \end{pmatrix}.$$

By the Kolmogorov extension theorem, there exists a Gaussian process on $\mathbb{S}^1$, $g \sim \mathcal{GP}(\mathbf{0}, k_0)$, such that $\mathbf{X}$ can be replaced by $(g(\vartheta_1), g(\vartheta_2), \ldots, g(\vartheta_m))^{\mathrm{T}}$ in Eq. (22), for any finite sequence of angles $\theta_1, \theta_2, \ldots, \theta_m$. Hence, we have shown that the random projection readout $g^{(M)}$ defines a random 2π-periodic function that has the same law as a Gaussian process as $M \to \infty$.

In summary, the above arguments show that our solvable RNN model can be used to draw samples from a periodic Gaussian process. In other words, it can represent a Gaussian process prior over periodic functions. This indicates that the Gaussian process framework originally developed for infinite-width feedforward neural networks [86, 87] also applies, in certain cases, to large networks with recurrent dynamics.

F *In vivo* experimental methods

Neuronal Recordings Transgenic mice expressing the GCamP6s calcium indicator [93] in all excitatory cortical neurons (CamKII x Ai162) were implanted with a $4\,\mathrm{mm}$ imaging window over visual cortex. Mice were headfixed under a custom Light Beads Microscope [65], modified with an optical stabilization module. A volumetric field of view of $2 \times 2 \times 0.3\ \mathrm{mm}^3$ over primary and higher visual areas was imaged with 2-3 $\mu\mathrm{m}$ lateral and $20\,\mu\mathrm{m}$ axial voxel spacing, with volume rates between 4.1-$4.5\ \mathrm{Hz}$. During recordings, mice were not anesthetized and were free to run on a low-resistance treadmill.

Visual Stimulation Visual stimuli were delivered via three screens that covered approximately $270°$ horizontally and $70°$ vertically of the visual field of the mouse. During drifting grating experiments, full-field drifting gratings with a temporal frequency of $2.0\,\mathrm{Hz}$ and spatial frequency of 0.04 cycles/degree were presented for $2\,\mathrm{s}$, with an inter-stimulus interval (ISI) drawn from a uniform distribution between 2-3 s. A sequence of 720 stimuli was presented with random directions drawn from a uniform distribution of integer angles ranging from $0°$ to $359°$. The same set of 720 stimuli was repeated in a new random order in the second half of the experiment. For natural image experiments, 1866 unique stimuli were presented, each composed of a mosaic of three distinct natural images. The images were chosen from the same set presented in [39]. Each stimulus was presented for $0.8\,\mathrm{s}$ with a random ISI between 0.9-$1.3\ \mathrm{s}$. Stimuli were presented in two shuffled sequences such that each stimulus was presented twice. For sessions of spontaneous activity, no stimuli were displayed and the screens were grey. Mice were awake, free to run (as in the other sessions) for 10 minutes of imaging.

Data preprocessing To detect cells and extract neuronal activity from volumetric fluorescence movies, we used Suite3D [66], an analysis pipeline for motion-correction, cell detection and signal extraction from volumetric two-phtoon data. After semi-automated curation, we analyzed activity from 19,223 $\pm$ 2,948 (mean $\pm$ std) neurons per experiment, across 7 experiments (3 spontaneous activity, 3 natural images, 1 drifting gratings, across three mice). The extracted cell fluorescence was deconvolved after neuropil substraction. Deconvolved traces were re-sampled at $5\,\mathrm{Hz}$ and aligned to stimulus onset events. For stimulus response sessions, we projected out the activity along the top 30 principal components (PCs) of spontaneous activity (computed on a spontaneous session recorded consecutively from the same mouse), similar to what was done in [39]. To compute stimulus responses, neuronal activity within a response window was averaged ($0.5\,\mathrm{s}$ for images, $2.0\,\mathrm{s}$ for gratings) to produce a single value per neuron per stimulus presentation. Deconvolved spontaneous activity was resampled using linear interpolation at 5 Hz.

G Neural Cross-Encoder

G.1 Datasets

For all three data conditions (gratings, natural images and spontaneous activity), we compared the performance of the (nonlinear) NCE model to (linear) Reduced Rank Regression on the same datasets. For results in Fig. 4 we used datasets with 500 source neurons and 1,000 target neurons. For stimulus-driven conditions (gratings and images), the source neurons were chosen to be the 500 neurons with the most reliable stimulus-evoked responses, quantified by the fraction of stimulus-related variance defined in Ref. [39]. 1,000 target neurons were chosen from the 2,000 non-source neurons with the highest stimulus-related variance. For the spontaneous condition, the target and source neurons were selected at random from the full population. We repeated the model fit for each condition and session five times with random initializations and neuron selections. In all cases, source and target neuron sets were non-overlapping.

To reduce the impact of non-stimulus related activity in stimulus-driven conditions, we used the fact that each stimulus was presented multiple times, and trained the model to predict the activity of the target neurons on one repeat of a stimulus from the activity of the source neurons on a different repeat, using all possible permutations. For example, if the same stimulus was shown at times t_1, t_2, the network was trained to minimize the sum of the squared Euclidean distances $\|\mathcal{F}(\mathbf{a}_{t_1}) - \mathbf{b}_{t_2}\|^2 + \|\mathcal{F}(\mathbf{a}_{t_2}) - \mathbf{b}_{t_1}\|^2$.

For the spontaneous condition, to capture the temporal structure in the latent dynamics, we used multiple timepoints of the source neurons to estimate the latent variable at a single timepoint. To predict $\boldsymbol{\kappa}_t$, we took a window of size L timepoints and concatenated the source activity within this time window, increasing the input dimensionality of the encoder: $\boldsymbol{\kappa}_t = \mathcal{E}([\mathbf{a}_{t-\frac{L}{2}}, \ldots, \mathbf{a}_t, \ldots \mathbf{a}_{t+\frac{L}{2}-1}])$. The number and size of the hidden layers were unchanged. For the experiments presented, the time window was $1.2\,\mathrm{s}$ ($L = 6$ timepoints).

In all three conditions, the paired source-target datasets were split into train-validation-test sets with a 50%-20%-30% split. For stimulus-driven datasets, the split was done using stimulus identity to ensure that all permuted pairs of repeats of a single stimulus were in the same set after splitting. For spontaneous datasets, instead of assigning each timepoint to one of three sets independently, we separated the time series into $10\,\mathrm{s}$ chunks with $2\,\mathrm{s}$ buffers between each chunk, and assigned each chunk to one of three sets. This procedure prevents the training set from contaminating the test set through the slow temporal autocorrelation of calcium signals.

G.2 Model Fitting

Linear models were fit on the training set using the closed-form solution of Reduced Rank Regression with a ridge penalty. The ridge penalty (which varied between 10^{-1}-10^{6}) giving the best performance in the validation set was selected.

For NCE, the parameters of the encoder $\mathcal{E}$ and the parameters of the decoder, i.e., the readout weights and biases $\mathbf{U} \in \mathbb{R}^{B \times d}$ and $\mathbf{c} \in \mathbb{R}^{B}$, the activation parameter p, and the baseline firing rates $\mathbf{r}_0 \in \mathbb{R}^{B}$, were learned by minimizing the mean-squared error (MSE) loss $\frac{1}{T}\sum_{t=1}^{T} \|\mathcal{F}(\mathbf{a}_t) - \mathbf{b}_t\|^2$. NCE models were trained in two phases: pretraining and fine-tuning. In the pretraining phase, we used a form of knowledge distillation [94]. Instead of predicting the activity of target neurons directly, the NCE was trained on the predicted activity of target neurons from the best-fitting linear model, one that potentially had a higher pre-activation dimension, d, than the NCE. In the fine-tuning phase, the NCE was trained to predict the true neuronal activity.

In both phases of training, we augmented the training set by adding noise to the activity of source neurons. First, we added Poisson-like shot noise present in two-photon recordings by generating zero-mean Gaussian noise with a variance equal to the activity of each neuron at a given timepoint, scaled by a factor of 0.05. Next, we added independent Gaussian noise with a mean of 0 and variance of 0.05. Finally, we incorporated multiplicative noise, multiplying the activity of each neuron at each timepoint with an independent Gaussian random variable with a mean of 1 and variance of 0.05. The original source dataset was concatenated with three noise-augmented datasets to produce the final training set.

In each phase, parameters were optimized through stochastic gradient descent using Adam [95] and a batch size of 2,048, with a learning rate of 10^{-3}, momentum parameters $\beta_1 = 0.85, \beta_2 = 0.95$, and an L_2 penalty of 10^{-5}. Model parameters were randomly initialized with Kaiming initialization [96]. Within each phase, we iterated between optimizing the readout parameters, encoder parameters, and all parameters. At each iteration, the model with the best validation performance, evaluated using the coefficient of determination, was selected. Once models were fitted, we report the fraction of explained variance on the test set. To normalize this value to only account for the "explainable" variance, we computed the coefficient of determination of each model, and normalized by the maximum of the best model across all pre-activation dimensionalities for the given set of neurons.

G.3 Latent dynamics of spontaneous activity

To determine whether the extracted latent variables related to the behavioral state of the animal, we compared them to the running speed of the animal. For each session of spontaneous activity, we trained an NCE model with a pre-activation dimension of 4, as described above. After training, we took the activity of the source neurons across the entire session (removing the train/validation/test splits), and computed the inferred latents.

The running speed of the mouse is highly correlated with inferred latent variables. We computed the correlation coefficient between each latent and the running speed, and found high positive and negative correlations for most variables. This supports the hypothesis that the latents inferred by the NCE are related to the behavioral state of the animal.

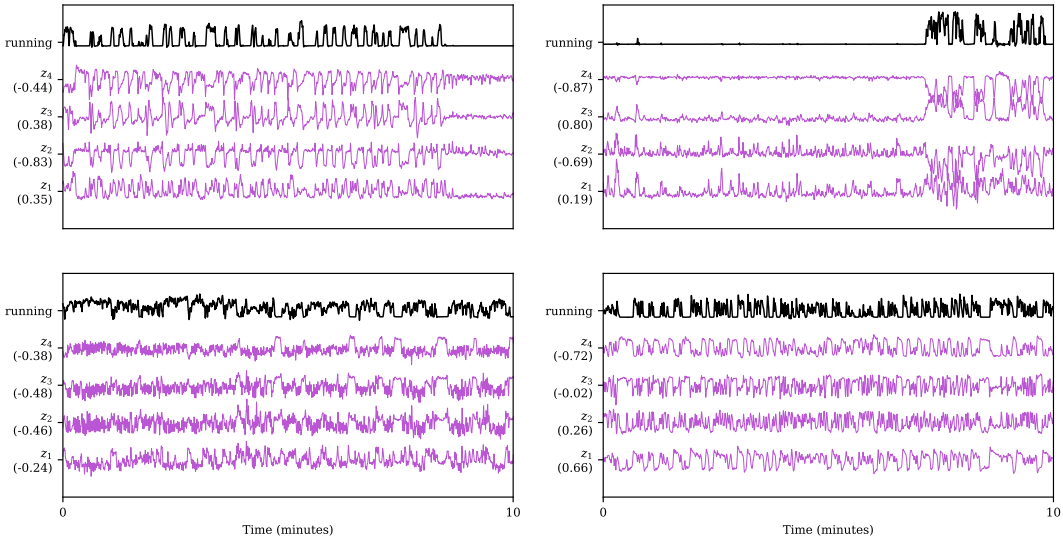

Figure 5: **Extracted latents correlate with running speed.** For four example sessions of spontaneous activity, the dynamics of the extracted latents z_i (purple) are correlated with the running speed (black). Correlation coefficient between each latent and running speed is reported in parantheses on the y-axis. All curves are smoothed with a 1s Gaussian kernel for visualization only.

G.4 Compute Resources

Volumetric calcium movies were processed using a custom workstation running Ubuntu 20.04.6 LTS with an Intel(R) Xeon(R) w9-3475X CPU (36 cores, 2.20 GHz), NVIDIA RTX A4500 GPU (20GB VRAM), and 512 GB DDR5 RAM. Preprocessing with Suite3D took <20 hours per session.

All NCE experiments were performed on a workstation running Windows 10 Pro, with an Intel (R) Core(TM) i7-11700k CPU (8 cores, 3.6 GHz), NVIDIA RTX 3060 GPU (12GB VRAM), and 128 GB DDR4 RAM. Total runtime of all NCE experiments was <96 hours. All preliminary experiments that are not reported in the paper were conducted on this workstation, with similar runtimes.

G.5 Software and Licenses

This work makes use of several publicly available software resources including: Python (PSF License), Matplotlib (PSF License), NumPy (BSD License), SciPy (BSD License), CuPy (MIT License), PyTorch (BSD-3 License), Suite3D (AGPL-3 License).

G.6 Code and Data Availability

All data and code used to train the NCE and generate the results, as well as code used for simulations in Fig. 2, are anonymized and shared via Figshare: https://figshare.com/s/aac82eac1829b7ec406e.