

892 Appendix

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

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

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

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

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

897 we have

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

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

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

899 The integral above can be solved, which yields

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

900 Using the sum and difference trigonometric identities,

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

901 we can write

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

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

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

904 B Derivation of pairwise correlations Eq. (6)

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

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

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

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

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

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

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

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

911 The correlation between any pair of neurons i and j is

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

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

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

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

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

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

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

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

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

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

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

919 Hence,

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

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

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

922 C Random low-rank RNNs with tractable latent dynamics

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

$$\mathbf{W} = \boldsymbol{\xi} \mathbf{M} \boldsymbol{\xi}^T, \quad (19)$$

926 where $\boldsymbol{\xi}$ 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
927 “overlap” matrix [45]. As explained in [45], thanks to the fact that the “patterns” $\boldsymbol{\xi}$ are Gaussian, the

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

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

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

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

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

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

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

936 (iii) For any mean $\mu \in \mathbb{R}$ and variance $\sigma^2 \geq 0$, if ϕ is the Gaussian c.d.f. $\phi(x) =$
 937 $\frac{1}{\sqrt{2\pi}\sigma} \int_{t=-\infty}^x \exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right) dt$, Eq. (20) reduces to

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

938 The proposition shows the nontrivial effect of the activation function has on the latent dynamics.
 939 For example, let us take again $d = 2$. When ϕ is the step function with bias $b = 0$, and if
 940 $\mathbf{M} = \sqrt{2\pi} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$, we recover the latent dynamics Eq. (5), which generates a stable limit cycle.
 941 In contrast, when ϕ is the ReLU function with bias $b = 0$, Eq. (21) becomes the linear system
 942 $\dot{\boldsymbol{\kappa}} = -\boldsymbol{\kappa} + \mathbf{M} \boldsymbol{\kappa}$, which can generate rotational dynamics but cannot generate a stable limit cycle.

943 *Proof.* To prove these statements, it suffices to solve the integral,

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

944 for the three different functions ϕ .

945 For (i),

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

946 For (ii),

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

947 and we integrate by parts,

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

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

$$\begin{aligned} \int_{y=-\infty}^{\infty} y \phi(y\|\boldsymbol{\kappa}\|) \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy &= \int_{y=-\infty}^{\infty} \phi'(y\|\boldsymbol{\kappa}\|) \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \\ &= \int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y\|\boldsymbol{\kappa}\|-\mu)^2}{2\sigma^2}\right) \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy. \end{aligned}$$

949 Then, we complete the square,

$$\begin{aligned} &\int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y\|\boldsymbol{\kappa}\|-\mu)^2}{2\sigma^2}\right) \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \\ &= \frac{1}{\sqrt{2\pi}} \int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\|\boldsymbol{\kappa}\|^2 y^2 - 2\mu\|\boldsymbol{\kappa}\|y + \mu^2 + \sigma^2 y^2}{2\sigma^2}\right) dy \\ &= \frac{1}{\sqrt{2\pi}} \int_{y=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\sigma^2 + \|\boldsymbol{\kappa}\|^2}{2\sigma^2} \left(y - \frac{\mu\|\boldsymbol{\kappa}\|}{\sigma^2 + \|\boldsymbol{\kappa}\|^2}\right)^2 + \frac{\mu^2\|\boldsymbol{\kappa}\|^2}{2\sigma^2(\sigma^2 + \|\boldsymbol{\kappa}\|^2)} - \frac{\mu^2}{2\sigma^2}\right) dy \\ &= \frac{1}{\sqrt{2\pi}(\sigma^2 + \|\boldsymbol{\kappa}\|^2)} \underbrace{\int_{y=-\infty}^{\infty} \frac{\sqrt{\sigma^2 + \|\boldsymbol{\kappa}\|^2}}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\sigma^2 + \|\boldsymbol{\kappa}\|^2}{2\sigma^2} \left(y - \frac{\mu\|\boldsymbol{\kappa}\|}{\sigma^2 + \|\boldsymbol{\kappa}\|^2}\right)^2\right) dy}_{=1} \\ &\quad \times \exp\left(-\frac{\mu^2}{2(\sigma^2 + \|\boldsymbol{\kappa}\|^2)}\right). \end{aligned}$$

950 \square

951 **D From the eigenvalues of the activity matrix $\mathbf{A}^{(N,T)}$ to the eigenvalues of the**

952 **random feature kernel operator Eq. (10)**

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

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

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

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

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

$$\frac{\lambda_n^{(N,T)}}{N} = \frac{\eta_n^{(N,T)}}{T}, \quad \forall n \in \llbracket 1, \min(N, T) \rrbracket.$$

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

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

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

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

960 Furthermore, we have that

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

961 Finally, by a theorem from Koltchinskii and Giné [53, Theorem 3.1], the scaled eigenvalues $\eta_n^{(T)}/T$
962 of the of the kernel matrix $\mathbf{K}^{(T)}$ converge (in a ℓ^2 sense) to the eigenvalues λ_n of the integral operator
963 Eq. (10). In conclusion, we have that, for any given n ,

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

E Gaussian processes from RNNs

As in Appendix B we start from an infinite-size network where the latent variables κ 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 κ . 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 $\tilde{w}_1, \tilde{w}_2, \dots, \tilde{w}_M$ 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 $M \rightarrow \infty$. For the mean, we have

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

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

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

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,

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

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 $\vartheta_1, \vartheta_2, \dots, \vartheta_m$,

$$(g^{(M)}(\vartheta_1), g^{(M)}(\vartheta_2), \dots, g^{(M)}(\vartheta_m))^T \xrightarrow[M \rightarrow \infty]{\mathcal{L}} \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma), \quad (22)$$

where

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

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 $(g(\vartheta_1), g(\vartheta_2), \dots, g(\vartheta_m))^T$ in Eq. (22), for any finite sequence of angles $\vartheta_1, \vartheta_2, \dots, \vartheta_m$. 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 $M \rightarrow \infty$.

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 [83, 84] 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 [90] in all excitatory cortical neurons (CamKII x Ai162) were implanted with a 4 mm imaging window over visual cortex. Mice were headfixed under a custom Light Beads Microscope [64], modified with an optical stabilization module. A volumetric field of view of $2 \times 2 \times 0.3 \text{ mm}^3$ over primary and higher visual areas was imaged with 2-3 μm lateral and 20 μm axial voxel spacing, with volume rates between 4.1-4.5 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 Hz and spatial frequency of 0.04 cycles/degree were presented for 2 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 s with a random ISI between 0.9-1.3 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 [65], an analysis pipeline for motion-correction, cell detection and signal extraction from volumetric two-photon 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 subtraction. Deconvolved traces were re-sampled at 5 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 s for images, 2.0 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. 3 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 κ_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: $\kappa_t = \mathcal{E}([\mathbf{a}_{t-\frac{L}{2}}, \dots, \mathbf{a}_t, \dots, \mathbf{a}_{t+\frac{L}{2}-1}])$. The number and size of the hidden layers were unchanged. For the experiments presented, the time window was 1.2 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 s chunks with 2 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 [91]. 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 dimensionality, 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 [92] 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 [93]. 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 dimensionality 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.

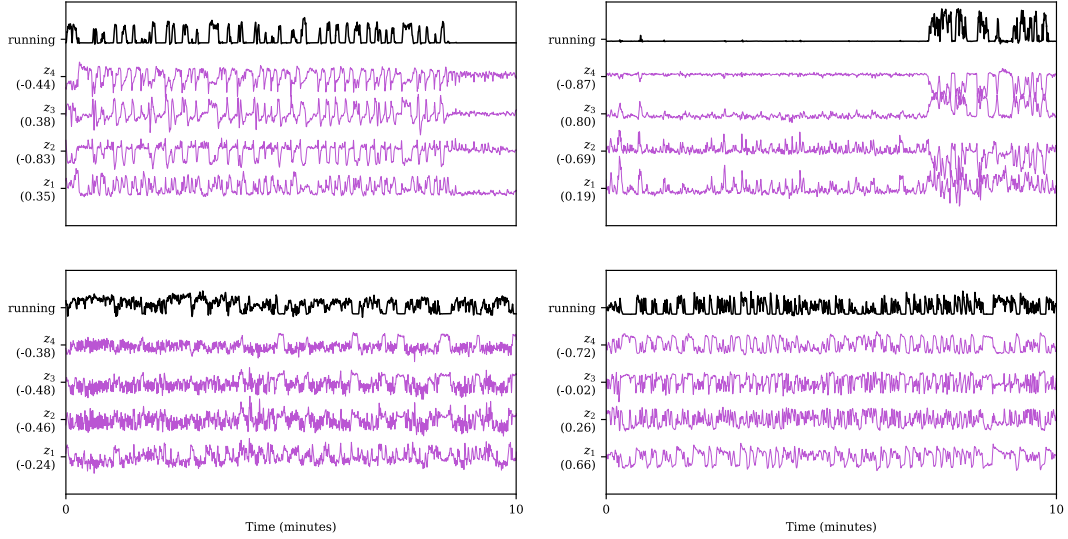


Figure 4: **Extracted latents correlate with running speed.** For four example sessions of spontaneous activity, the dynamics of the extracted latents z_i (purple) are correlated with the running speed (black). Correlation coefficient between each latent and running speed is reported in parantheses on the y-axis. All curves are smoothed with a 1s Gaussian kernel for visualization only.

1097 G.4 Compute Resources

1098 Volumetric calcium movies were processed using a custom workstation running Ubuntu 20.04.6 LTS
 1099 with an Intel(R) Xeon(R) w9-3475X CPU (36 cores, 2.20 GHz), NVIDIA RTX A4500 GPU (20GB
 1100 VRAM), and 512 GB DDR5 RAM. Preprocessing with Suite3D took <20 hours per session.

1101 All NCE experiments were performed on a workstation running Windows 10 Pro, with an Intel (R)
 1102 Core(TM) i7-11700k CPU (8 cores, 3.6 GHz), NVIDIA RTX 3060 GPU (12GB VRAM), and 128
 1103 GB DDR4 RAM. Total runtime of all NCE experiments was <96 hours. All preliminary experiments
 1104 that are not reported in the paper were conducted on this workstation, with similar runtimes.

1105 G.5 Software and Licenses

1106 This work makes use of several publicly available software resources including: Python (PSF
 1107 License), Matplotlib (PSF License), NumPy (BSD License), SciPy (BSD License), CuPy (MIT
 1108 License), PyTorch (BSD-3 License), Suite3D (AGPL-3 License).

1109 G.6 Code and Data Availability

1110 All data and code used to train the NCE and generate the results, as well as code used for simulations
 1111 in Fig. 2, are anonymized and shared via Figshare: <https://figshare.com/s/aac82eac1829b7ec406e>.