ARTICLE **Communicated by Eric Denovellis**

Extended Poisson Gaussian-Process Latent Variable Model for Unsupervised Neural Decoding

Della Daiyi Luo
dl67@rice.edu
Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005, U.S.A.

Bapun Giri
bapung@umich.edu
Kamran Diba
kdiba@umich.edu
Department of Anesthesiology, Neuroscience Graduate Program, University of Michigan Medical School, Ann Arbor, MI 48109, U.S.A.

Caleb Kemere
caleb.kemere@rice.edu
Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005, U.S.A.

Dimension reduction on neural activity paves a way for unsupervised neural decoding by dissociating the measurement of internal neural pattern reactivation from the measurement of external variable tuning. With assumptions only on the smoothness of latent dynamics and of internal tuning curves, the Poisson gaussian-process latent variable model (P-GPLVM; Wu et al., 2017) is a powerful tool to discover the low-dimensional latent structure for high-dimensional spike trains. However, when given novel neural data, the original model lacks a method to infer their latent trajectories in the learned latent space, limiting its ability for estimating the neural reactivation. Here, we extend the P-GPLVM to enable the latent variable inference of new data constrained by previously learned smoothness and mapping information. We also describe a principled approach for the constrained latent variable inference for temporally compressed patterns of activity, such as those found in population burst events during hippocampal sharp-wave ripples, as well as metrics for assessing the validity of neural pattern reactivation and inferring the encoded experience. Applying these approaches to hippocampal ensemble recordings during active maze exploration, we replicate the result that P-GPLVM learns a latent space encoding the animal's position. We further demonstrate that this latent space can differentiate one maze context from another. By inferring the latent variables of new neural data during running, certain neural patterns are observed to reactivate, in accordance

Neural Computation **36, 1449–1475** (2024)
https://doi.org/10.1162/neco_a_01685

with the similarity of experiences encoded by its nearby neural trajectories in the training data manifold. Finally, reactivation of neural patterns can be estimated for neural activity during population burst events as well, allowing the identification for replay events of versatile behaviors and more general experiences. Thus, our extension of the P-GPLVM framework for unsupervised analysis of neural activity can be used to answer critical questions related to scientific discovery.

1 Introduction

Memory critically requires the firing of neurons in the hippocampus during ongoing experiences and afterward as the resultant memories are consolidated. In rodents, individual neurons in the hippocampal subfields are activated when an animal explores a specific location (the neuron's place field). Sparsely active at an individual level, the ensemble of place fields is thought to provide a location signal that can support navigation. While rodent studies have focused on spatial memories, hippocampal neurons can be generally understood to represent the conjunction of the sensory features associated with a particular context (Moser et al., 2015) and the temporal sequences that connect local contexts across time during an experience (Eichenbaum & Cohen, 2014; Eichenbaum, 2017). In this cognitive map framework, the ensemble activity thus represents unobserved conceptual places that evolve over time on a latent manifold. Importantly, sequential firing patterns of neural ensembles reactivate in a temporally compressed manner during some of the population burst events (PBEs) that occur during sharp-wave ripple oscillations in sleep or quiet wakefulness (Wilson & McNaughton, 1994; Skaggs & McNaughton, 1996; Kudrimoti et al., 1999; Nádasdy et al., 1999). By decoding those events, it has been shown that the replay trajectories show a continuum of conformity to the original experience, including variability in momentum and both forward and reverse reexpression (Lee & Wilson, 2002; Foster & Wilson, 2006; Diba & Buzsáki, 2007; Csicsvari et al., 2007; Davidson et al., 2009; Krause & Drugowitsch, 2022). Traditionally, individual replay events have been identified based on a strong assumption of ordered consistency with patterns expressed during exploration. Consequently, unordered replay of contexts or ordered replay of more complicated routes are often excluded from subsequent analysis. Thus, while much has been learned about memory consolidation and recall from the study of replay, existing approaches have colored our understanding. Therefore, a technique for decoding neural activity without strongly stereotyping the patterns represented or requiring a specifically spatial encoding model would be a powerful tool for quantitatively exploring the cognitive map and thus understanding memory.

To extract the information from spike trains with minimal prior assumptions, one practicable approach is to find a low-dimensional embedding

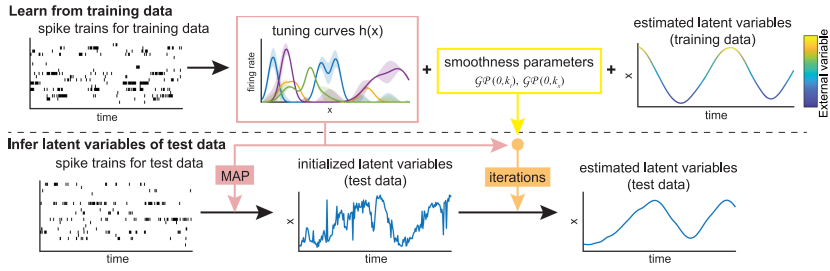

Figure 1: Schematic diagram of the extended P-GPLVM model. Internal tuning curves and smoothness information learned from training data are then used to constrain the inference of test data latent variables in the same latent space.

that can reveal the underlying dynamics. The Poisson gaussian-process latent variable model (P-GPLVM) proposed by Wu et al. (2017) is a probabilistic, nonlinear, and dynamic dimension-reduction approach. It infers temporally smooth low-dimensional latent neural trajectories and smooth, nonparametric internal tuning curves from spike trains without referring to external variables. This model consists of Poisson spiking observations and two gaussian processes, one governing the temporal evolution of latent variables and another governing the nonlinear mapping from high-dimensional neural data to low-dimensional latent variables.

In the learned low-dimensional latent space, mapping any possible external variable into the embedding, how external variables are represented in this latent space can be revealed, and by measuring the goodness of fit between sequences of ensemble activity and the learned model, we can detect the reactivation of neural patterns associated with experiences encoded in the training data (Yu et al., 2009; Rubin et al., 2019; Nieh et al., 2021). Unsupervised neural decoding can be achieved by dissociating the measurement of neural pattern reactivation from the measurement of external variable tuning. However, an approach for inferring latent variables of new data points in the learned latent space is lacking from the original P-GPLVM model, limiting its utility for decoding.

In this article, we extend the P-GPLVM framework to enable the latent variable inference of new inputs constrained by the learned smoothness parameters and tuning curves (see Figure 1) and develop subsequent analyses to evaluate the new input data in the latent space. We also present a preprocessing pipeline for PBE decoding when using models trained from behavioral data. The original P-GPLVM can be used to reveal encoded information in neural activity and discover neural trajectory evolution but only within the given training data. This extended model and PBE preprocessing pipeline, especially, make use of information learned in the model and enable the effective and unsupervised decoding of new neural activity during both behavior and PBEs.

2 Poisson Gaussian-Process Latent Variable Model

2.1 Model Structure. Binning spike counts into M time bins from N neurons creates the matrix of spike trains $\mathbf{Y} = \{\mathbf{y}_m\}_{m=1}^{M}$, $\mathbf{y}_m \in \mathbb{R}^{N\times 1}$. The mth time bin was recorded at time t_m, $m \in (1, \ldots, M)$. In this model, two latent variable matrices will be learned: the log firing rate for Poisson spiking, $\mathbf{F} = \{\mathbf{f}_m\}_{m=1}^{M}$, $\mathbf{f}_m \in \mathbb{R}^{N\times 1}$, and the P-dimensional latent variables, $\mathbf{X} = \{\mathbf{x}_m\}_{m=1}^{M}$, $\mathbf{x}_m \in \mathbb{R}^{P\times 1}$.

2.1.1 Latent Dynamics. Each feature of the latent variable matrix $\mathbf{X}$ evolves according to a gaussian process depending on time t, $x_p(t) \sim \mathcal{GP}(0, k_t)$, $p \in (1, \ldots, P)$, where $k_t(t, t') = r\exp(-|t - t'|/l)$, governing the temporal smoothness of x_p. Writing as a multivariate normal distribution,

$$\mathbf{x}_{p,1:M}|t_{1:M} \sim \mathcal{N}(\mathbf{0}, K_t). \tag{2.1}$$

The covariance K_t is an $M \times M$ matrix with entries k_t at all pairs of time bins.

2.1.2 Nonlinear Mapping. Let $h : \mathbb{R}^P \to \mathbb{R}$ be a nonlinear mapping function, describing the firing rate of a neuron in the mth time bin as $\lambda_m = h(\mathbf{x}_m)$. The log tuning curve of the nth cell in response to the latent variable $\mathbf{x}$ is modeled as another gaussian process as $f_n(\mathbf{x}) = \log h_n(\mathbf{x}) \sim \mathcal{GP}(0, k_x)$, $n \in (1, \ldots, N)$. This process has $k_x(\mathbf{x}, \mathbf{x}') = \rho\exp(-\|\mathbf{x} - \mathbf{x}'\|_2^2/2\delta^2)$ as a spatial covariance function. Therefore, the log firing rates of the nth neuron in all time bins have the multivariate normal distribution,

$$\mathbf{f}_{n,1:M}|\mathbf{x}_{1:M} \sim \mathcal{N}(\mathbf{0}, K_x), \tag{2.2}$$

with the $M \times M$ covariance matrix K_x, whose entries are k_x evaluated for all pairs of $\mathbf{x}$. By combining $\mathbf{f}$ over all the neurons, $\mathbf{F} \in \mathbb{R}^{N\times M}$ as firing rates in units of spike counts per time bin are obtained. Note that this is per bin rather than per second. This has consequences for PBE decoding, as described below.

2.1.3 Poisson Spiking. Finally, for the nth neuron in the mth time bin, observed spike counts $y_{n,m}$ are assumed to be drawn from a Poisson process given the latent firing rate $\lambda_{n,m} = \exp(f_n(\mathbf{x}_m))$,

$$y_{n,m}|f_n, \mathbf{x}_m \sim \text{Poisson}(\exp(f_n(\mathbf{x}_m))). \tag{2.3}$$

2.2 Model Training. During training, the model iteratively infers latent variables of training data without mapping constraints and optimizes smoothness parameters. In each iteration, $\mathbf{X}$ is first fixed and $\mathbf{F}$ is optimized. For each neuron, the posterior over $\mathbf{f}_n$ is given by

$$p(\mathbf{f}_n|\mathbf{y}_n, \mathbf{X}) \propto p(\mathbf{y}_n|\mathbf{f}_n)p(\mathbf{f}_n|\mathbf{X}). \tag{2.4}$$

The optimal $\hat{\mathbf{f}}_n$ is found by maximizing the log conditional distribution,

$$\log p(\mathbf{f}_n|\mathbf{y}_n, \mathbf{X}) = \log p(\mathbf{y}_n|\mathbf{f}_n) - \frac{1}{2}\mathbf{f}_n^\top K_x^{-1}\mathbf{f}_n - \frac{1}{2}\log|K_x| + constant. \tag{2.5}$$

Then, fixing $\mathbf{F}$, the optimal $\mathbf{X}$ is discovered by maximizing the conditional likelihood,

$$\mathbf{X}_{\text{MAP}} = \text{argmax}_{\mathbf{X}} \sum_{n=1}^{N} p(\mathbf{y}_n|\mathbf{X})p(\mathbf{X}), \tag{2.6}$$

where $p(\mathbf{y}_n|\mathbf{X})$ is given by

$$p(\mathbf{y}_n|\mathbf{X}) \propto \int p(\mathbf{y}_n|\mathbf{f}_n)p(\mathbf{f}_n|\mathbf{X})d\mathbf{f}_n. \tag{2.7}$$

Using Laplace's method, the approximated log likelihood conditioned on $\mathbf{X}$ is

$$\log q(\mathbf{y}_n|\mathbf{X}) = \log p(\mathbf{y}_n|\hat{\mathbf{f}}_n) - \frac{1}{2}\hat{\mathbf{f}}_n^\top K_x^{-1}\hat{\mathbf{f}}_n - \frac{1}{2}\log|I_M + K_x W_n|, \tag{2.8}$$

where $W_n = -\nabla\nabla \log p(\mathbf{y}_n|\mathbf{f}_n)$. Hyperparameters $\boldsymbol{\theta} = \{\rho, \delta, r, l\}$ are found by maximizing the same likelihood function.

3 Extensions of P-GPLVM

3.1 Constrained Latent Variable Inference of New Data.

3.1.1 Latent Variable Initialization. During model training, P-GPLVM learns smoothness parameters, $\boldsymbol{\theta}$, and internal mapping function, $\log h_n : \mathbb{R}^P \to \mathbb{R}$, parameterized by latent variables $\hat{\mathbf{f}}_{\text{train}} = \{\hat{f}_m\}_{m=1}^M$ for each cell and $\hat{\mathbf{X}}_{\text{train}} = \{\hat{\mathbf{x}}_m\}_{m=1}^M$ from training data. In the original P-GPLVM paper, the tuning curve vectors $\mathbf{f}_{\text{grid}}$ are evaluated at the grid of latent variables $\mathbf{X}_{\text{grid}} = \{\mathbf{x}_g\}_{g=1}^G$ using a joint gaussian distribution with $\hat{\mathbf{X}}_{\text{train}}$ and $\hat{\mathbf{f}}_{\text{train}}$,

$$\begin{bmatrix} \hat{\mathbf{f}}_{\text{train}} \\ \mathbf{f}_{\text{grid}} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K_{\text{train}} & \mathbf{k} \\ \mathbf{k}^\top & K_{\text{grid}} \end{bmatrix}\right), \tag{3.1}$$

where $K_{\text{train}} = k_x(\hat{\mathbf{X}}_{\text{train}}, \hat{\mathbf{X}}_{\text{train}})$, $K_{\text{grid}} = k_x(\mathbf{X}_{\text{grid}}, \mathbf{X}_{\text{grid}})$, and $\mathbf{k} = k_x(\hat{\mathbf{X}}_{\text{train}}, \mathbf{X}_{\text{grid}})$ are the covariance matrices. The entry $k_x(\mathbf{X}_1, \mathbf{X}_2)_{m,g} = k_x(\mathbf{x}_{1m}, \mathbf{x}_{2g})$. The posterior distribution of $\mathbf{f}_{\text{grid}}$ can then be written as

$$\mathbf{f}_{\text{grid}}|\mathbf{X}_{\text{grid}}, \hat{\mathbf{f}}_{\text{train}}, \hat{\mathbf{X}}_{\text{train}} \sim \mathcal{N}(\mathbf{k}^\top K_{\text{train}}^{-1} \hat{\mathbf{f}}_{\text{train}}, \text{diag}(K_{\text{grid}}) - \mathbf{k}^\top K_{\text{train}}^{-1} \mathbf{k}). \quad (3.2)$$

Given new input data from the same group of neurons, $\mathbf{Y}_{\text{new}} = \{\mathbf{y}_t\}_{t=1}^T$, $\mathbf{y}_t \in \mathbb{R}^{N\times 1}$, the latent variables, $\mathbf{X}_{\text{new}} = \{\mathbf{x}_t\}_{t=1}^T$, $\mathbf{x}_t \in \mathbb{R}^{P\times 1}$, do not have an analytical solution as the hidden layer $\mathbf{f}_n$ is unknown. Hence, for each $\mathbf{y}_t$, the posterior probability of $\mathbf{x}_t$ location is estimated in a coarse grid $\mathbf{X}_{\text{grid}}$ spanning the latent space using a Bayesian approach. $\mathbf{X}_{\text{new}}$ are initialized with elements in $\mathbf{X}_{\text{grid}}$ that maximize the posterior probability. At the tth time bin, the posterior over each element in $\mathbf{X}_{\text{grid}}$ is

$$p(\mathbf{x}_g|\mathbf{y}_t, \hat{\mathbf{F}}_{\text{train}}, \hat{\mathbf{X}}_{\text{train}}) \propto p(\mathbf{y}_t|\mathbf{x}_g, \hat{\mathbf{F}}_{\text{train}}, \hat{\mathbf{X}}_{\text{train}}) p(\mathbf{x}_g), \quad (3.3)$$

where we've assumed a uniform prior over $\mathbf{x}_g$ during initialization. The prior over $\mathbf{y}_t$ is

$$p(\mathbf{y}_t|\mathbf{x}_g, \hat{\mathbf{F}}_{\text{train}}, \hat{\mathbf{X}}_{\text{train}}) = \prod_{n=1}^{N} p(y_{n,t}|f_{n,g}) p(f_{n,g}|\mathbf{x}_g, \hat{\mathbf{f}}_{\text{train}n}, \hat{\mathbf{X}}_{\text{train}}). \quad (3.4)$$

The latent variable $\mathbf{x}_t$ corresponding to $\mathbf{y}_t$ is initialized with the $\mathbf{x}_g$ with maximum posterior.

3.1.2 Impose Smoothness Constraints. Merely using MAP tuning curves to constrain the latent variable inference of new input data omits the learned temporal smoothness in the latent space. To impose both temporal smoothness and tuning constraints, subsequent iterations are needed. We noted that employing $\mathbf{f}_{\text{grid}}$ and $\mathbf{X}_{\text{grid}}$ as tuning curve vectors is computationally expensive; considering a samples per dimension, a P-dimensional $\mathbf{X}_{\text{grid}}$ contains a^P elements. Moreover, a majority of the grid elements contain minimal information as no $\hat{\mathbf{x}}_{\text{train}}$ is located nearby. To save computational costs during iterations and to preserve the tuning details of $\hat{\mathbf{f}}_{\text{train}}$ in response to $\hat{\mathbf{X}}_{\text{train}}$ as much as possible, $\hat{\mathbf{F}}_{\text{train}}$ and $\hat{\mathbf{X}}_{\text{train}}$ were directly used as the tuning curve vectors (TC), $\hat{\mathbf{F}}_{\text{TC}}$, and their encoded latent variables, $\hat{\mathbf{X}}_{\text{TC}}$.

Iterations are similar to those during unconstrained inference as described in section 2.2, but the smoothness parameters are fixed as learned, and the mapping function $\log h_n$ is constrained by $\hat{\mathbf{f}}_{\text{TC}}$ and $\hat{\mathbf{X}}_{\text{TC}}$. This is achieved by substituting $p(\mathbf{f}_n|\mathbf{X}, \hat{\mathbf{f}}_{\text{TC}}, \hat{\mathbf{X}}_{\text{TC}})$ for $p(\mathbf{f}_n|\mathbf{X})$ in equations 2.4 and 2.7. As in equation 3.1, the joint distribution with $\hat{\mathbf{f}}_{\text{TC}}$ and $\hat{\mathbf{X}}_{\text{TC}}$ gives the posterior of $\mathbf{f}_n$ as

$$\mathbf{f}_n|\mathbf{X}, \hat{\mathbf{f}}_{\text{TC}_n}, \hat{\mathbf{X}}_{\text{TC}} \sim \mathcal{N}(\boldsymbol{\mu}_n, \sigma^2 I_T), \quad (3.5)$$

where $\boldsymbol{\mu}_n = \mathbf{k}^\top K_{\mathrm{TC}}^{-1} \hat{\mathbf{f}}_{\mathrm{TC}n}$, $\mathbf{k} = k_x(\hat{\mathbf{X}}_{\mathrm{TC}}, \mathbf{X})$, and $K_{\mathrm{TC}} = k_x(\hat{\mathbf{X}}_{\mathrm{TC}}, \hat{\mathbf{X}}_{\mathrm{TC}})$ and σ^2 is the observation noise. Correspondingly, equation 2.5 is modified as

$$\begin{aligned} &\log p(\mathbf{f}_n | \mathbf{y}_n, \mathbf{X}, \hat{\mathbf{f}}_{\mathrm{TC}n}, \hat{\mathbf{X}}_{\mathrm{TC}}) \\ &= \log p(\mathbf{y}_n | \mathbf{f}_n) - \frac{1}{2\sigma^2} (\mathbf{f}_n - \boldsymbol{\mu}_n)^\top (\mathbf{f}_n - \boldsymbol{\mu}_n) - \frac{T}{2} \log \sigma^2. \end{aligned} \tag{3.6}$$

Finally, obtaining the optimal $\hat{\mathbf{f}}_n$, equation 2.8 is modified as

$$\begin{aligned} \log q(\mathbf{y}_n | \mathbf{X}, \hat{\mathbf{f}}_{\mathrm{TC}n}, \hat{\mathbf{X}}_{\mathrm{TC}}) = \log p(\mathbf{y}_n | \hat{\mathbf{f}}_n) - \frac{1}{2\sigma^2} (\hat{\mathbf{f}}_n - \boldsymbol{\mu}_n)^\top (\hat{\mathbf{f}}_n - \boldsymbol{\mu}_n) \\ - \frac{1}{2} \log |I_T + \sigma^2 W_n|. \end{aligned} \tag{3.7}$$

3.2 Preprocessing for PBE Decoding. Critically for our model, neural activity during PBEs is temporally compressed relative to that expressed during exploration. Consequently, the firing rate model learned on exploration-related neural activity will not properly model PBEs. This is addressed in two steps, choosing a shorter time bin (implicitly compressing time) and explicitly scaling the model parameters. Temporal compression implies that the time intervals between pairs of place cells activating during exploration running are expected to be proportional to those during replay. Thus, the cross-correlation histogram (Harrison et al., 2013; Karlsson & Frank, 2009) of spike trains from place cell pairs is used to find the PBE time bin size that would best match the binned behavioral data. Place cells are identified in either running direction and then are pooled. For each place cell pair with spike trains $s_1, \ldots, s_M$ from one cell and $t_1, \ldots, t_N$ from another, time lags between any pairs of spikes (s_m, t_n) within 4 seconds are recorded. The histogram of all these spike time lags is the cross-correlation histogram (CCH) for this cell pair. All histograms are centered at 0 s time lag and share the same number of bins. For ease, the histogram bin size of CCH during active exploration is set the same as that used for P-GPLVM analysis of these data. Among all pairwise combinations of place cells, cell pairs with place field (where peak firing rate occurs) distances larger than the 80th percentile of the distribution of travel distance in 4 seconds are excluded from subsequent analysis. For each included cell pair, the Pearson correlation between CCH during exploration and CCH during PBEs is evaluated. The optimal bin size for PBEs is found by maximizing the sum of all significant positive Pearson correlation coefficients using a range of possible PBE bin sizes.

As the unit of our tuning curves is spike counts per time bin, for PBE decoding, the firing rates of P-GPLVM tuning curves learned using neural activity during active exploration must be further scaled. Scaling by the ratio of the time bin sizes is found to yield good PBE-decoding performance.

We varied this scaler by −40% to +100%, and the resulting metrics are consistent (i.e., 0.02–0.07 in section 4.4, where the time bin ratio is 0.034).

3.3 Analysis in Latent Space.

3.3.1 Number of Well-Separated Manifolds. The hippocampus remaps between different environments, meaning that whether a given neuron is active, and if so, how its spatial tuning will relate to that of other neurons is essentially random in different environments (Moser et al., 2015; Alme et al., 2014), though recent studies have challenged this concept (Cai et al., 2016). This implies that different environments should correspond to well-separated manifolds in latent space. Thus, we asked whether the trajectories of the low-dimensional latent variable $\hat{\mathbf{X}}_{\text{train}}$ are organized on a single manifold or on multiple well-separated manifolds. This procedure consists of constructing a K-nearest neighbor graph in $\hat{\mathbf{X}}_{\text{train}}$ and then counting the number of connected components in the graph. Components containing less than 3% of the total number of points are considered residual components. Starting with a small number, K is gradually increased until there are no residual components left. The number of well-separated manifolds will stabilize as K continues to increase multiple times. This stabilized number is defined as the number of connected components in $\hat{\mathbf{X}}_{\text{train}}$. Each separated manifold is denoted as $\hat{\mathbf{X}}(i)$ hereafter.

3.3.2 Congruence with the Learned Manifold(s). We present three metrics to assess the congruence of new test data with the learned model. The most important one, log likelihood, reflects the adjacency of new input with the tuning curve vectors not only in the low-dimensional latent space but also in the high-dimensional neuronal space, which by comparing with shuffled data can answer the basic question of whether it is a neural pattern reactivation. After being validated as reactivation, spatial consistency and step distance in the low-dimensional latent space can provide a high-level summary of the temporal pattern and spatial distribution of the latent trajectory. Training and test data can contain multiple continuous recording segments. Each continuous segment is a single neural trajectory in the latent space. When the neural cofiring patterns and dynamics of new inputs $\mathbf{Y}_{\text{new}}$ match the learned model perfectly, their estimated latent trajectories $\mathbf{X}_{\text{new}}$ will (1) have much higher likelihood values compared with shuffled data, (2) span the manifold of $\hat{\mathbf{X}}_{\text{TC}}$, and (3) progress smoothly along the manifold.

Log likelihood. Log likelihood measures how well the learned P-GPLVM model fits the test data. The joint probability of $\mathbf{Y}$, $\mathbf{F}$, $\mathbf{X}$ for test data is computed as

$$p(\mathbf{Y}, \mathbf{F}, \mathbf{X}|\hat{\mathbf{F}}_{\text{TC}}, \hat{\mathbf{X}}_{\text{TC}}, \boldsymbol{\theta}) = p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{X}, \hat{\mathbf{F}}_{\text{TC}}, \hat{\mathbf{X}}_{\text{TC}}, \rho, \delta)p(\mathbf{X}|r, l). \tag{3.8}$$

The log likelihood (LLH) of a continuous segment in test data is written as

$$\begin{aligned} LLH &= \log p(\mathbf{Y}, \mathbf{F}, \mathbf{X} | \hat{\mathbf{F}}_{\mathrm{TC}}, \hat{\mathbf{X}}_{\mathrm{TC}}, \boldsymbol{\theta}) \\ &= \sum_{n=1}^{N} \sum_{t=1}^{T} \log p(y_{n,t} | f_{n,t}) + \sum_{n=1}^{N} \log p(\mathbf{f}_n | \mathbf{X}, \hat{\mathbf{f}}_{\mathrm{TC}_n}, \hat{\mathbf{X}}_{\mathrm{TC}}, \rho, \delta) \\ &\quad + \sum_{p=1}^{P} \log p(\mathbf{x}_p | r, l). \end{aligned} \tag{3.9}$$

Recall that

$$y_{n,t} | f_{n,t} \sim \mathrm{Poisson}(\exp(f_{n,t})), \tag{3.10}$$

$$\mathbf{f}_n | \mathbf{X}, \hat{\mathbf{f}}_{\mathrm{TC}_n}, \hat{\mathbf{X}}_{\mathrm{TC}}, \rho, \delta \sim \mathcal{N}(k_x(\hat{\mathbf{X}}_{\mathrm{TC}}, \mathbf{X})^\top k_x(\hat{\mathbf{X}}_{\mathrm{TC}}, \hat{\mathbf{X}}_{\mathrm{TC}})^{-1} \hat{\mathbf{f}}_{\mathrm{TC}n}, \sigma^2 I_T), \tag{3.11}$$

$$\mathbf{x}_p | r, l \sim \mathcal{N}(\mathbf{0}, K_t). \tag{3.12}$$

Spatial consistency. This metric measures how well a latent trajectory, $\mathbf{X}_{\mathrm{new}} = \{\mathbf{x}_t\}_{t=1}^{T}$, spans the latent variable manifold(s) of the tuning curve vector, $\hat{\mathbf{X}}_{\mathrm{TC}} = \{\hat{\mathbf{x}}_m\}_{m=1}^{M}$. First, for each point $\mathbf{x}_t$ in the trajectory, its K-nearest neighbors are searched in the manifold $\hat{\mathbf{X}}_{\mathrm{TC}}$, obtaining the set of its nearest neighbors $S_t = \{\hat{\mathbf{x}}_{m_k}\}_{k=1}^{K}$ and its distances to those neighbors $\{||\mathbf{x}_t - \hat{\mathbf{x}}_{m_k}||_2\}_{k=1}^{K}$. When the trajectory spans the manifold nicely, the union of neighbor sets, $\bigcup_{t=1}^{T} S_t$, should have many more elements than by chance. The number of all identified neighbors is then weighted by the neighbor distances to integrate the neighborhood quality into the measure.

For each $\hat{\mathbf{x}}_m \in \bigcup_{t=1}^{T} S_t$, its distance to its closest point in $\mathbf{X}_{\mathrm{new}}$, $d_m = \min_{t,k} ||\mathbf{x}_t - \hat{\mathbf{x}}_{m_k}||_2$ is estimated. Then the spatial consistency of this trajectory in the latent space is computed as

$$consistency = \sum_{m=1}^{M} w_m, \tag{3.13}$$

where

$$w_m = \begin{cases} \frac{c}{c+d_m}, & \text{if } \hat{\mathbf{x}}_m \in \bigcup_{t=1}^{T} S_t \\ 0, & \text{otherwise} \end{cases} \tag{3.14}$$

which means that when one $\hat{\mathbf{x}}_m$ is identified as a neighbor, it contributes to the spatial consistency. This contribution is weighted by the distance to its nearest neighbor on the trajectory, which is 1 when the distance is 0, and then decays as the distance increases, reaching 0.5 when the distance

is c. The parameter c is set as 20% of the standard deviation of $\hat{\mathbf{X}}_{\mathrm{TC}}$. When $\hat{\mathbf{X}}_{\mathrm{TC}}$ contains more than one separated manifold, the standard deviation is computed as the average within-manifold standard deviation,

$$c = 20\% \sqrt{\frac{1}{M} \sum_{i} \sum_{\hat{\mathbf{x}}_m \in \hat{\mathbf{X}}(i)} ||\hat{\mathbf{x}}_m - \boldsymbol{\mu}_i||_2^2}, \tag{3.15}$$

where $\boldsymbol{\mu}_i$ is the center of manifold $\hat{\mathbf{X}}(i)$.

A higher value of spatial consistency indicates that this trajectory travels a long way along the tuning curve manifold(s) rather than only jiggling at a small local area or locating far away. The spatial consistency to $\hat{\mathbf{X}}_{\mathrm{TC}}$ is first used to evaluate the neural trajectory behavior in the latent space. Further, when $\hat{\mathbf{X}}_{\mathrm{TC}}$ contains more than one well-separated manifold $\hat{\mathbf{X}}(i)$, we also estimate the manifold contribution ratio to the consistency value, which is computed as in equation 3.13 but only for $\hat{\mathbf{x}}_m \in \hat{\mathbf{X}}(i)$. The ratio between these contribution values from manifolds indicates to which manifold the trajectory belongs.

Average step distance. This metric reveals the consistency between temporal adjacency and pattern-matched location adjacency. The average Euclidean distance of each step in each latent trajectory of $\mathbf{X}_{\mathrm{new}}$ is estimated. In general, the tuning curve constraint tends to drive $\mathbf{X}_{\mathrm{new}}$ toward their pattern-matched locations on the $\hat{\mathbf{X}}_{\mathrm{TC}}$ manifold. When the new data have different temporal sequences from the training data, pattern-matched locations of temporally adjacent points are mostly distant, leading to longer step distances.

3.3.3 External Variables Decoding. Similar trajectories in the latent space indicate repetitions of neural activity encoding similar experiences. Therefore, the decoding of external variables can be simply achieved by a K-nearest neighbor (KNN) decoder. Given a test data point, its corresponding external variables can be revealed by referring to those of its K nearest neighbors among $\hat{\mathbf{X}}_{\mathrm{TC}}$, the training data points, in the latent space. The external variables of the training data neighbors are not necessarily consistent since the neural activity might not be tuned by only one single external variable.

4 Experiments and Results

The data set used in this article is a neural recording collected in the dorsal hippocampal CA1 and CA3 areas from a rat. The recording started with a 3 hour rest session (pre), and then the animal was exposed to a novel Z-shaped maze (maze1) for 50 minutes, running back and forth to get rewards at the two ends. Following another two hours of rest (post1), the

animal explored a novel U-shaped maze (maze2) for 50 minutes. Finally, the animal went through a 5 hour rest session (post2). Spike trains during the two maze exploration sessions were binned into 500 ms time bins. Animal positions were linearized, and then active running periods (speed >3 cm/s and peak speed >5 cm/s) were extracted along with their corresponding time bin indices.

Known for encoding a wide range of variables like spatial locations and sensory cues and even abstract knowledge, neural activity in the hippocampus is believed to form a cognitive map of their real-world experiences supported by latent representations (Whittington et al., 2022). In this data set, as the neural activity at each time bin is represented as one data point in the latent space in this model, we expect to see the locations of latent data points disambiguating different neural patterns and to see the spatial structure of all latent data points revealing the topology of the encoded animal experiences.

4.1 Exploration in One Maze. We first examined this P-GPLVM with the neural activity when the animal was exposed to only one maze (maze1). The linearized animal positions during running in maze1 exploration session are shown in Figure 2A and corresponding original positions in Figure 2B. The data from spike trains during active running in maze1 for 200 s in total were selected as training data for the model (orange lines in Figure 2A, colored scatters in Figure 2B).

Theoretically, a higher number of latent space dimensions can capture more details about information encoded in the neural activity. The more complicated the behavioral paradigm is, the higher the dimension of latent space is needed to reveal the topology of the underlying neural structure fully. However, a higher latent space dimension dictates longer model training time and significantly larger memory demand, and it might bring about the curse of dimensionality as gaussian process estimation depends heavily on distances in the latent space. For this data set, we picked the dimension of latent variables to be 3 for reasonable training time, easy visualization, and good representation of the animal's behavioral paradigm. Higher latent dimensions (i.e., 4, 5) have results of similar metric values.

After fitting the model, smoothness parameters $\boldsymbol{\theta}$ and latent variables of training data, $\hat{\mathbf{X}}_{\text{train}}$ and $\hat{\mathbf{F}}_{\text{train}}$, were obtained. Only one well-separated manifold is found in $\hat{\mathbf{X}}_{\text{train}}$, suggesting that there are no multiple distinctly different neural processes found in these training data, aligned with the fact that all the training data are from similar experiences in maze1. The trajectories in this manifold share a similar path, revealing the topology of the animal behavior where it ran along a maze track repeatedly. Given the known spatial tuning in the hippocampus, to examine the information encoded in the latent space, $\hat{\mathbf{X}}_{\text{train}}$ are color-coded with the corresponding animal linearized positions. Position information appears to be encoded along the manifold smoothly (see Figure 2C). The latent neural trajectories seem to be a bit

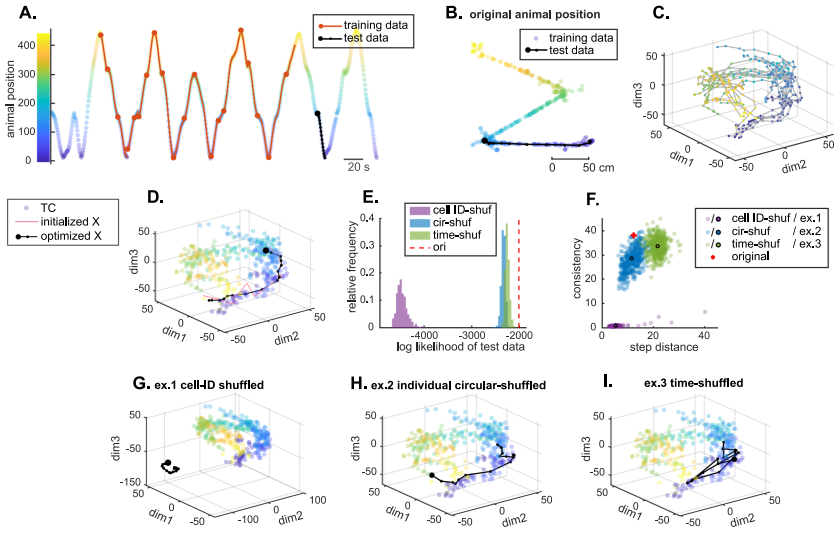

Figure 2: (A) Linearized animal position during running periods in maze1 exploration session. The orange/black lines indicate the continuous segments used as training and test data. The start of each segment is marked by a large filled circle. Each step in the test data is marked by a small dot. Scatters are color-coded by linearized animal positions, the same in subsequent panels (B to D and G to I). (B) Colored scatters of original animal positions in the training data. The black line indicates the same test data segment in panel A. (C) Latent variables of training data in the learned P-GPLVM. Continuous segments are indicated by dotted lines as latent neural trajectories. (D) Inferred neural trajectory of test data in the learned latent space. The semitransparent markers are identical to the dot scatters in panel C, which, in this panel, indicates the latent variable manifold of tuning curve vectors, $\hat{\mathbf{X}}_{\mathrm{TC}}$. The black line depicts the inferred neural trajectory of test data, progressing along the manifold $\hat{\mathbf{X}}_{\mathrm{TC}}$ in accordance with the real animal positions (from blue to purple). (E) Log likelihood of the original test data (dashed red line) and its shuffled versions (histograms). (F) Estimated step distances versus spatial consistency to $\hat{\mathbf{X}}_{\mathrm{TC}}$ of the inferred neural trajectory in panel D compared with its cell identity-shuffled (cell ID-shuf), individual circular-shuffled (cir-shuf), and time-shuffled (time-shuf) versions. The three black-edged circles indicate one example of cell ID-shuffled test data (ex.1), of circular-shuffled test data (ex. 2), and of time-shuffled test data (ex. 3), respectively. (G, H, I) Same presentation as in panel D, but the inferred latent trajectories of ex. 1 to 3 in panel F.

unkempt, which is still reasonable because this was the animal's first exposure to maze1.

Next, the spike trains of a continuous running segment in maze1 are used as test data to evaluate the decoding performance (black line in

Figures 2A and 2B). We estimated the neural pattern reactivation of the test data in the learned latent space. In Figure 2D, the manifold of tuning curve vector latent variables $\hat{\mathbf{X}}_{\mathrm{TC}}$, which is $\hat{\mathbf{X}}_{\mathrm{train}}$ in this setting, is shown as the colored semitransparent markers. Using the Bayesian approach, the initialized latent variables of test data appear as a bumpy trajectory (light red line). The constrained inference is used, and the latent variables converged to a smooth latent trajectory after 15 iterations (black line). By comparing the color in Figures 2B and 2D, it is obvious that the inferred neural trajectory of test data proceeds closely along those of training data with similar experiences, where the animal traveled from positions in blue to positions in purple, in accordance with the reactivation of similar neural patterns.

To validate this internal neural pattern reactivation and contrast the inferred latent variables of test data, three kinds of surrogate data were generated:

- Cell identity(ID)-shuffled data, by randomly permuting the cell identity of the test data, which scrambles the original neural coactivation patterns and permutes neuronal firing rates but preserves the temporal sequences.
- Local individual circular-shuffled data, by randomly and circularly permuting the test data of each cell individually within each continuous segment, which scrambles coactivation patterns but preserves the temporal smoothness and local neuronal firing rates.
- Local time-shuffled data, by randomly permuting the temporal order of the test data within each continuous segment, which preserves the coactivation patterns but scrambles the time sequences.

For each shuffle type, we generated 500 surrogate test data. Metric values and example latent trajectories of the original and surrogate data are shown in Figure 2.

Figure 2E depicts the distribution of LLH values of the inferred latent variables of the original test data and of the surrogate test data. Overall, the original test data have a higher LLH value than all the surrogate data. Figure 2F shows their step distance and spatial consistency values in scatters. Inferred latent trajectories of example surrogate data are shown in Figures 2G to 2I, whose metric values are indicated in the black-edged scatter in Figure 2F. In the latent space, the trajectories of cell ID-shuffled and time-shuffled test data look distinct from that of the original test data trajectory, while the circular-shuffled test data trajectory has visually similar behavior.

Note that the iterative inference process searches for an optimal balance between temporal smoothness and tuning curve constraints. Because of the changed relative firing rates among cells in cell ID-shuffled data, the tuning curve constraints push the trajectories far away from $\hat{\mathbf{X}}_{\mathrm{TC}}$, where temporal smoothness constraint dominates, leading to small step distances, diminutive spatial consistency values, and the lowest LLHs among all types of shuffled data. With preserved local neuronal firing rates, circular-shuffled

data trajectories stay in the similar area as the original data, but due to the scrambled coactivation patterns and preserved temporal smoothness, the tuning curve constraints are compromised by the temporal smoothness constraint, resulting in less likely but smooth trajectories with step distances comparable to the original test data, slightly lower spatial consistency values, and significantly lower LLH values. As for time-shuffled data, with intact coactivation patterns at each time point, the tuning curve constraints are dominating during the inference, driving the latent variables to the corresponding locations on the manifold of $\hat{\mathbf{X}}_{\text{TC}}$ as the original test data do, while the temporal smoothness is sacrificed, resulting in spiky trajectories with large step distances, comparable spatial consistency, and slightly lower LLH values.

We contrast the LLH values of original test data with its shuffled versions by z-scoring. Given that the shuffled test data have occasional occurrences of extremely low LLH values, the traditional z-scoring using mean and standard deviation is not ideal because of its sensitivity to outliers. Therefore, we use a robust z-scoring here, calculated by dividing the difference from the median by the median absolute deviation. The z-scored LLH of the original test data among its circular-shuffled versions is 13.56, and among its time-shuffled versions, 8.98, indicating that the test data are a valid sequential reactivation of neural patterns captured in the training data. This measure separately verifies the matching of coactivation patterns between test data and tuning curves (when compared with circular-shuffled test data), and the matching of temporal sequences (when compared with time-shuffled test data).

4.2 Exploration in Two Mazes. Next, we looked into neural activity during exploration in more than one environment context. Since hippocampal place field remapping is observed among different contexts (Kubie et al., 2020), we wondered whether this remapping is present and, if so, whether this model can detect the different neural processes of two contexts.

Spike trains during active running in maze1 for 200 s in total (yellow lines in Figure 3A, colored scatters on the left in Figure 3B) and those in maze2 for 200 s in total (scarlet lines in Figure 3A, colored scatters on the right in Figure 3B) are selected and concatenated as the training data to train a new P-GPLVM. The test data are from maze1, the same as in the previous section (black line in Figures 3A and 3B). All the semitransparent markers in the scatter plot are colored by the corresponding linearized animal positions.

In $\hat{\mathbf{X}}_{\text{train}}$ of the newly learned model, two well-separated manifolds, m1 ($\hat{\mathbf{X}}(1)$) and m2 ($\hat{\mathbf{X}}(2)$), are found, each of which turns out to contain all data points from one of the maze contexts. Using the same color code as in Figures 3A and 3B on $\hat{\mathbf{X}}_{\text{train}}$, position information appears to be encoded along the corresponding context manifold smoothly (see Figure 3C). Thus, this approach is capable of detecting the presence of two environment contexts as

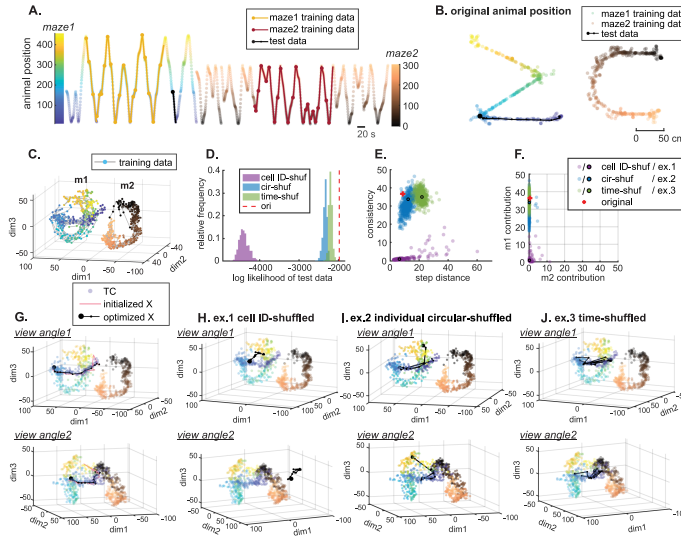

Figure 3: (A) Linearized animal position during running periods in maze1 and maze2 exploration sessions. The yellow/scarlet lines indicate the segments in maze1/maze2 used as training data. The black line indicate the test data. Scatters are color-coded by contexts and linearized animal positions, the same in panels B to C and G to J. (B) Scatters of original animal positions in training data from maze1 (left) and maze2 (right). The black line indicates the same test data segment as in panel A. (C) Following fitting the training data, latent variables are separated into two distinct manifolds, m1 and m2. Each data point is colored by its corresponding linearized animal position as in panel A. (D) Log likelihoods (LLH) of test data (red dashed line), cell ID-shuffled (purple histogram), individual circular-shuffled (blue histogram), and time-shuffled surrogate data (green histogram). (E) Inferred latent neural trajectory step distance versus spatial consistency to $\hat{\mathbf{X}}_{\mathrm{TC}}$ of original test data (red +), of cell ID-shuffled (purple scatters), of individual circular-shuffled (blue scatters), and of time-shuffled (green scatters) surrogate data. Example surrogate data indicated in one black-edged circle from either shuffle type is then shown for visualization. (F) Manifold contribution to neural trajectory spatial consistency from m1 versus from m2, of original and surrogate test data. (G) Latent neural trajectories as initialized (light red line) and optimized after iterations (black line) of test data in the latent space. Latent variables of tuning curve vectors, $\hat{\mathbf{X}}_{\mathrm{TC}}$, are shown in semitransparent markers color-coded by the animal positions, same as in panel C. Both initialized and optimized trajectories of test data are associated with the corresponding context manifold. Unlike the bumpy initialization, the optimized neural trajectory is smooth and progresses according to the animal positions. (H–J) Same representations as in panel G inferred trajectory of panel H, ex.1 for cell ID-shuffled test data, panel I, ex. 2 for individual circular-shuffled test data, and panel J, ex. 3, for time-shuffled test data, as indicated in black-edged circles in panels E and F.

well as capturing the encoded position information merely from the neural activity in the training data.

Again, cell ID-shuffled, individual circular-shuffled, and time-shuffled surrogate test data were generated. Along with the original test data, their latent variables in the new latent space are inferred, which behave similar to those in section 4.1. Figure 3D depicts the distribution of LLH values of the original and surrogate test data. Cell ID-shuffled test data have much lower LLHs than all the other types of shuffled data. Circular-shuffled data have slightly smaller LLHs than time-shuffled data. The original test data have a higher LLH value than all surrogate data, where z-scored LLH among its circular-shuffled versions is 11.75, and among its time-shuffled versions, 8.40, confirming that the test data provide a valid sequential neural pattern reactivation.

Next, we investigated the content of this neural pattern reactivation by looking into the latent variable locations relative to $\hat{\mathbf{X}}_{\text{train}}$. The inferred latent trajectory of the test data and initialization in this new latent space are shown in Figure 3G. The trajectory is similar to those neural trajectories of training data with similar experiences, traversing the same arm of maze1, from positions blue to purple, just as in Figure 3B.

As a high-level summary of the spatial distribution and dynamics of these neural trajectories, Figure 3E shows the step distance and spatial consistency values to $\hat{\mathbf{X}}_{\text{train}}$, while Figure 3F shows the manifold contribution to spatial consistency values from m1 versus from m2. Examples are shown in Figures 3H to 3J (metric values indicated in black-edged scatters in Figures 3E and 3F). The inferred cell ID-shuffled test data trajectories tend to have small steps and are located far from both manifolds, having diminutive spatial consistency values. Compared with the original test data trajectory, circular-shuffled data have smooth trajectories around the same area, with slightly smaller spatial consistency values and comparable step distances. Latent variables of time-shuffled test data have similar locations to the original test data but mostly take big jumps across time. Like the original test data, both circular-shuffled and time-shuffled test data have similarly high m1 contribution to spatial consistency and diminutive m2 contribution, suggesting the test data are reactivating neural patterns encoding experience in maze1 rather than maze2.

4.3 Validating the Neural Pattern Reactivation. In the results shown in the previous section, the test data from maze1 were found to be a neural pattern reactivation of training data in maze1 rather than maze2. However, it is not always the case that the training data have included experiences similar to the test data. To examine the validity of neural pattern reactivation, the spike trains during multiple other continuous running segments in maze1 and in maze2 were used as test data (see the black lines in Figure 4A), and we compared their inferred latent variables in the latent space learned from only one maze.

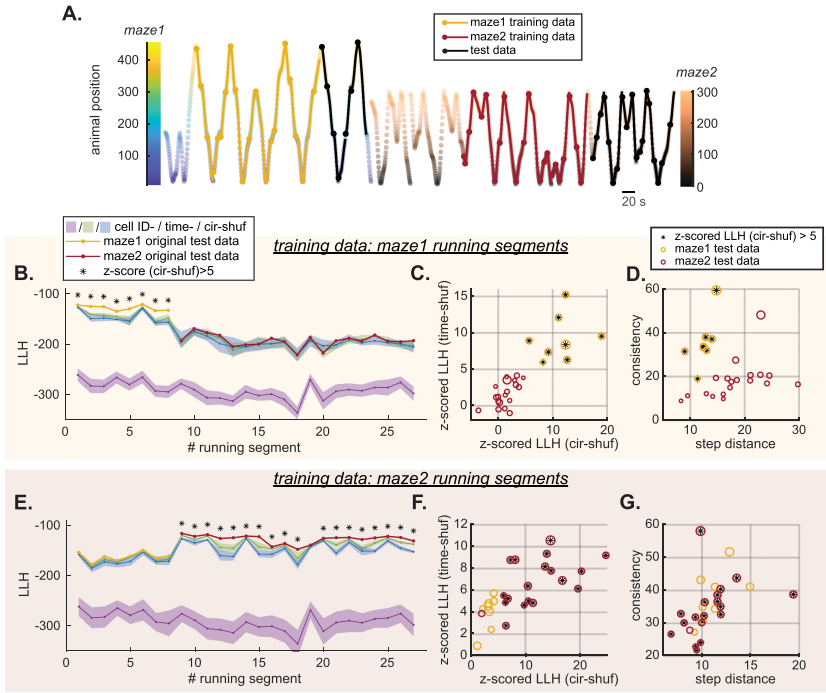

Figure 4: (A) Linearized animal position during running periods in maze1 and maze2 exploration sessions. The yellow/scarlet lines indicate the segments in maze1/maze2 used as training data. The black lines indicate the test data in maze1 and maze2. Scatters are color-coded by contexts and linearized animal positions. (B–D) Results in the latent space learned from running segments in maze1. (E–F) Results in the latent space learned from running segments in maze2. (B,E) Log likelihood (LLH) values of original test data and their shuffled versions in the corresponding latent space. LLHs are normalized by the number of time bins to show all segments together. The yellow/scarlet dotted line shows the LLH value of each running segment in the original test data in maze1/maze2. The purple/green/blue dotted line indicates the median LLH value of cell ID-/time-/circular-shuffled versions of each running segment in test data. The shades indicate ± 3 * median absolute deviation. Segments whose robust z-scored LLHs among their circular-shuffled versions greater than 5 are marked by an asterisk. (C,F) Each circle shows the z-scored LLH value of each running segment among its circular-shuffled versions versus its time-shuffled versions. The size of the marker indicates the spatial consistency value of the segment. Yellow/scarlet circles indicate running segments in maze1/maze2. (D,G) Each circle shows the average step distance versus spatial consistency value of each running segment with the same representation as in panels C and F.

The latent space learned from maze1 is the same as in section 4.1, trained by the maze1 part of the training data in section 4.2 (see the yellow lines in Figure 3A). Figure 4B shows the contrast of LLH values among the original test data and shuffled versions of each running segment. Similar to the inferred latent variables of circular shuffled test data in Figure 2H, latent variables of test data in maze2 still span the training data manifold. However, the LLH values of original test data in maze2 (scarlet dotted line) are lower than those in maze1 (yellow dotted line) because they belong to a neural process different from the training data. Nevertheless, if the test data contain only experiences from maze2, we cannot make this comparison to verify that they are not neural pattern reactivations. The original test data from both mazes have distinctly higher LLH values than their cell ID-shuffled versions (purple dotted line and shading). Contrasting with their time- (green dotted line and shading) and circular-shuffled versions (blue dotted line and shading), LLH values remain noticeably higher for the original test data in maze1 but become comparable for the original test data in maze2. Therefore, test data with an LLH value significantly higher than the cell ID-shuffled versions do not qualify as a valid neural pattern reactivation. Local individual circular-shuffled data are a better fit than simple cell ID-shuffled test data to serve as a proper surrogate data of "natural random cofiring patterns" due to the relatively stable neuronal firing rates even in different contexts. For each running segment, the robust z-scored LLH of the original test data among its time- and circular-shuffled versions is depicted as a circle in Figure 4C, whose size indicates the spatial consistency value of that segment. The z-scored LLHs among circular-shuffled versions separate test data in maze1 from those in maze2 with a threshold of 5, differentiating the test data that belong to the neural process the same as the training data from those that do not.

Similar results are observed in the latent space learned from maze2, trained by the maze2 part of the training data in section 4.2 (see the scarlet lines in Figure 3A). Setting a threshold of 5 for the z-scored LLH values among circular-shuffled data, all test data passing the threshold are in maze2, belonging to the same neural process as the training data. Only one running segment of short duration (5.5 s) in maze2 is missed.

Figures 4D and 4G show the spatial consistency and the average step distance values of latent variables of the test data in the two latent spaces. Test data that are from the same neural process as training data seem to have higher spatial consistency values and smaller steps than those that are not. But there is no consistent threshold separating the test data from different mazes.

Overall, z-scored LLH value among circular shuffled test data is sensitive to the difference of population cofiring patterns and thereby is able to differentiate neural processes and serve as an indicator of the validity of neural pattern reactivation. To our knowledge, this is the first method by which neural repetition can be robustly evaluated in a neural-process-specific

manner with no need for comparison with the neural activity known to belong to certain neural processes.

4.4 PBE Decoding. Reactivation of hippocampal place cell sequences of previous experiences has been observed in PBEs, which occurred during slow wave sleep or when the animal had paused running in the exploration session, typified by their decoded positions traversing a spatial path at fast and constant speeds (Lee & Wilson, 2002; Karlsson & Frank, 2009; Davidson et al., 2009). We show the neuronal spiking in part of the maze1 running test data in section 4.3 and in an example replay event during a PBE in Figures 5A and 5B. For visualization, cells are sorted according to the peaks of their internal tuning curves along the common path of $\hat{\mathbf{X}}(1)$, the maze1 manifold in section 4.2.

We have demonstrated that the model learned from training data during running can capture the neural pattern reactivation in test data during running well. Next, we evaluated whether this running-state model could be used to estimate the pattern reactivation and infer the replayed experience in neural data during PBEs, which occurred when the animal had paused running in the exploration session.

4.4.1 Identifying Replay Events. First, the best time bin size for PBE data was searched. In both the maze1 and maze2 exploration sessions, the 80th percentile of 4 seconds' travel distance is 86 cm to 87 cm. Therefore, only place cell pairs with their place field distances less than 87 cm are included in the subsequent analysis. It turns out that for both maze sessions, the sum of significant ($p < 0.01$) positive Pearson correlation coefficient r between CCH during running and CCH during PBEs across cell pairs reached its peak when the PBE time bin size is 17 ms. Therefore, PBEs were binned into 17 ms time bins.

Traditionally, assuming that spatial location is the major factor driving the neural activity change, PBEs are decoded using a memoryless Bayesian decoder based on the spatial tuning curves of identified place cells, which are estimated by averaging the neural firings at each spatial bin across time during active running. Then, replay scores are evaluated by the linearity of the decoded position distribution as a function of time. Replay events are identified by their high replay scores compared with shuffled data. In this data set, we estimated spatial tuning curves in the two running directions separately and used a Bayesian model to decode the PBE positions following the traditional method. The Bayesian decoded position posterior distribution of the example replay event in Figure 5B is shown as the grayscale matrix in Figure 5D.

In contrast, our model identifies replay events of experiences by verifying internal neural pattern reactivation among PBEs, no longer restricted to a certain replay route.

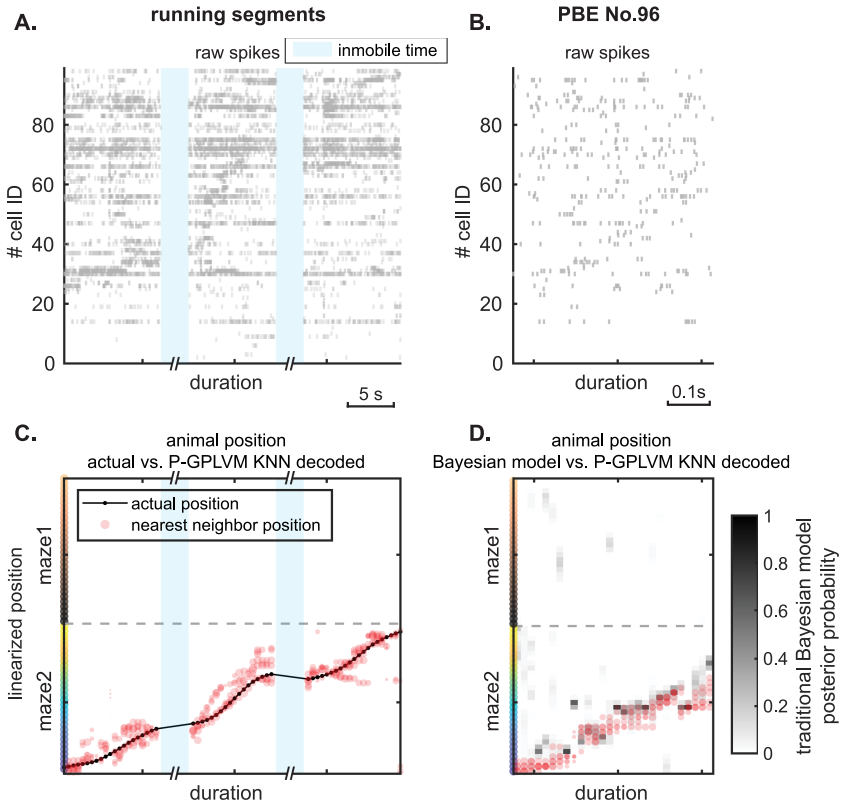

Figure 5: Illustration of a replay event. (A) Spikes during active running segments. Cells are sorted according to the peaks of their internal tuning curves. Cells with firing rates larger than 10 Hz are excluded. (B) Spikes during an example replay event during a PBE. Cell order is the same as in panel A. (C) The black dotted line depicts the actual animal position during the running segments in panel A. Red dots depict animal positions of training data points identified as the nearest neighbors of test data points in the P-GPLVM latent space. Each red dot represents one nearest neighbor whose size indicates the neighbor adjacency. (D) The grayscale matrix depicts the animal position posterior distribution by traditional Bayesian position decoder. Each pixel represents the posterior probability of a position bin at a time point, normalized across position bins to sum to 1. Corresponding context and position colors of position bins are indicated on the left border, the same color code as Figure 3A. Red dots depict animal positions of training data points identified as the nearest neighbors of test data points in the P-GPLVM latent space.

In the latent space learned using running data from the two maze exploration sessions described in section 4.2, we examined whether the binned PBE data could be identified as neural pattern reactivations. Firing rates of the tuning curve vectors were scaled by the time bin size ratio 17/500. Among 562 PBEs detected within immobile periods of the maze1 exploration session, we randomly selected 100 PBEs as test data. To evaluate the resulting latent trajectories, for each PBE, 200 surrogate data for each of the three shuffling types were generated and metric values were assessed.

We consider a PBE to be a significant replay event when its LLH value is larger than all of those of its circular-shuffled versions and its robust z-scored LLH among circular-shuffled versions is greater than 3. This threshold is lowered from 5, which is for test data in the behavioral timescale in section 4.3, to 3 for test data in PBEs whose neural firing patterns are not exactly the same. Moreover, a significant replay event should have a latent trajectory located close to the tuning curve latent variable $\hat{\mathbf{X}}_{\mathrm{TC}}$ manifolds to guarantee the strength of the tuning curve constraint. Therefore, the average distance of the latent trajectory to its nearest neighbors identified during measuring the spatial consistency value is estimated. PBEs with an average nearest neighbor distance larger than the standard deviation of $\hat{\mathbf{X}}_{\mathrm{TC}}$ are excluded. Given that neuron firings of a replay are not exactly the temporally compressed version of those during corresponding behavior, this criterion is a strict indicator of the ensemble cofiring pattern reactivation and therefore a strict indicator of the validity of being a replay event. Fifty out of the selected 100 PBEs meet the criterion and are considered significant replay events.

4.4.2 Decoding Replayed Experiences. Following the identification as a replay event, we can now investigate what is replayed and how it is replayed.

The external variables of replayed experiences can be inferred by those of their nearest neighbors in $\hat{\mathbf{X}}_{\mathrm{TC}}$ in the latent space. For test data during running, the decoding result of neural activity in Figure 5A using a KNN position decoder in the P-GPLVM latent space is shown in Figure 5C. The actual animal position of each test data point in these running segments (black dotted line) is reliably mirrored by those of its nearest neighbors in $\hat{\mathbf{X}}_{\mathrm{TC}}$ (red dots, $k = 6$). Similarly, based on the inferred latent neural trajectory of neural activity in Figure 5B, we overlap the KNN position decoding results as red dots with the traditional Bayesian decoding results in Figure 5D. Results from both methods suggest the experience of running from position blue to position green is replayed. However, note that the traditional Bayesian model is tracking repeated animal position change and our method is tracking repeated experience. The traditional Bayesian decoder could work well here because position information is indeed encoded in the neural activity in this data set, as shown in section 4.1. Figure 6C shows the manifold contribution to spatial consistency value from m1 manifold versus from m2 manifold of each PBE. As expected, since the animal hadn't

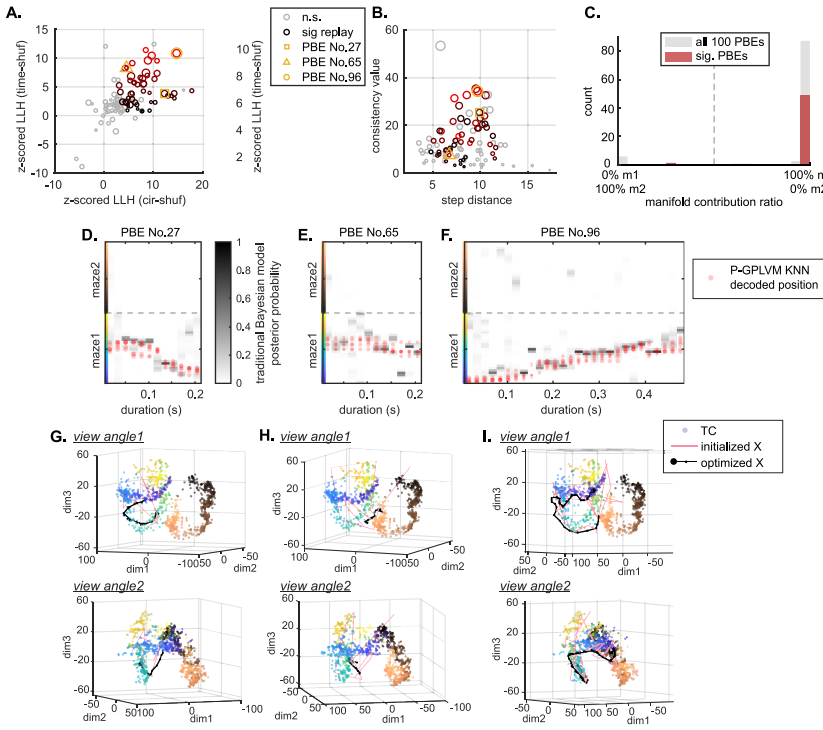

Figure 6: Inferring latent trajectories of PBE data in the P-GPLVM latent space learned from running data. (A) Robust z-scored LLH values among circular-shuffled versions versus among time-shuffled versions of the 100 selected PBEs. The size of each circle indicates the spatial consistency value of the PBE. PBEs not meeting the replay criterion are in gray, and PBEs identified as significant replay events are colored according to their z-scored LLH among time-shuffled versions. (B) Average step distance of the latent trajectory of each test datum versus their spatial consistency to the tuning curve latent variable manifolds, $\hat{\mathbf{X}}_{\mathrm{TC}}$. PBEs are presented in the same way as in panel A. (C) Histogram of ratio of manifold contribution to spatial consistency. All of the significant PBEs are associated with the maze1 manifold. (D–F) Same representation as in Figure 5D for the example PBEs (No. 27, No. 65, and No. 96) which are highlighted by markers in panels A and B. (G–I) Inferred latent trajectories of the example PBEs in panels D to F in the P-GPLVM latent space. While the initialized trajectories are jumpy, after iterations, the trajectories of the PBEs all converge to the maze1 manifold (same representation as in Figure 3G).

experienced maze2 yet, almost all of the probable replay events belong to the m1 manifold rather than to the m2 manifold, indicating that they are replay events of the maze1 experience.

The continuity of the replay can be revealed by the z-scored LLH among its time-shuffled versions. Figure 6A shows the LLH values z-scored among circular-shuffled versions versus z-scored among time-shuffled versions, of the 100 selected PBEs. Each PBE is depicted as a circle, whose size stands for its spatial consistency value. All significant replay events are colored according to their z-scored LLH among time-shuffled versions, and insignificant PBEs are colored in gray.

The temporal pattern of the replayed experience and the portion of experiences being replayed in each identified replay were also investigated with the aid of its average step distance and spatial consistency to $\hat{\mathbf{X}}_{\mathrm{TC}}$ (see Figure 6B). Three example PBEs among those significant replay events are highlighted to illustrate the implications of those metrics (yellow markers in Figures 6A and 6B). For comparison, their posterior probabilities of animal positions obtained by the traditional Bayesian model (grayscale matrix) are shown in Figures 6D to 6F, along with their KNN position decoding results obtained in our P-GPLVM latent space (red dots). Their inferred latent trajectories are visualized subsequently in Figures 6G to 6I.

PBE No. 27 in Figures 6D and 6G exemplifies replay events that have "jumps" in their trajectories, which are not the typical replays that can be recognized in traditional methods. The traditional Bayesian position decoder reveals its replayed route from location green to location blue with a jump in the middle. According to the KNN decoder in the latent space, the latent trajectory of this event suggests a similar route as the Bayesian model, only that the jump is smoothed out to some extent. Note that even after searching for the best-matched time bin size and scaling the firing rate, the neural firing patterns of a replay event are still not exactly the same as those during the behavioral timescale. Therefore, when the tuning curve constraints for PBEs are not strong enough, sudden jumps in the replay trajectories can be smoothed by the temporal smoothness constraint in P-GPLVM. However, this discontinuity can still be revealed by its low z-scored LLH among time-shuffled data. The latent trajectory of this example PBE has a larger step distance relative to those with comparable spatial consistency because of the jump.

PBE No. 65 in Figures 6E and 6H exemplifies replay events that have more complicated routes. As suggested by the Bayesian decoded positions, its route goes to and fro around locations green and yellow. The KNN decoder shows a smoothed route at similar locations. Indicated by its relatively high z-scored LLH among time-shuffled data, PBE No. 65 is more likely to be a continuous replay compared with the previous example, PBE No. 27. Because of the smoothed to-and-fro route, the latent trajectory of PBE No. 65 has a small step distance and a small spatial consistency value, which indicates stationary or slightly oscillating trajectories.

PBE No. 96 in Figures 6F and 6I, also in Figure 5D, exemplifies the typical replays recognized in traditional methods, which are continuous and one-directional. Both the traditional Bayesian decoder and our P-GPLVM

KNN position decoder disclose this PBE as a continuous replay of experience traveling from location purple all the way to location green. Its long step distance and high consistency value close to the diagonal of the chart in Figure 6B also suggest that it replays extensive experience at a steady speed.

To sum up, for a significant replay event, by combining the observation of metric values, we are able to evaluate the external variables, the temporal continuity, and the temporal pattern of the replayed experience and also the portion of experiences being replayed. Note that we have not used position data in any of our analyses except inferring external variables. This approach is the first presentation of a method for continuously decoding and identifying replay events without relying on external variables.

4.4.3 Replayed Experience Overview. Finally, with no need to assume external variables that drive the neural activity or predetermine replay behavior patterns, the summary of replay behavior throughout all sessions in this experiment can be revealed by applying the extended P-GPLVM model to all PBEs detected. Latent neural trajectories of all PBEs are inferred. For each PBE, 100 circular-shuffled versions were generated to identify significant replay events. PBE occurrence time versus the ratio of manifold contribution to PBE spatial consistency value is shown in Figure 7 from pre rest session to post2 rest session in time order, where each PBE is depicted as a circle, whose size indicates its spatial consistency value to $\hat{\mathbf{X}}_{\mathrm{TC}}$. Circles on the top/bottom line mean 100% of identified nearest neighbors for the latent neural trajectories are from the m1/m2 manifold, indicating affiliation with maze1/maze2 context experience. Circles located in between suggest mixtures. PBEs identified as significant replay events are colored according to its z-scored LLH value among their circular-shuffled versions. In the pre rest session, identified significant replay events are comparably sparse and have small spatial consistency values. In maze1 and maze2 exploration sessions, as expected, almost all the significant replay events are associated with their current contexts and have comparably larger consistency and z-scored LLH values, which indicates long and sequential experience replay. In the post1 rest session, the majority of significant replay events are associated with the maze1 context and most of them have small consistency values. Those replay events became less associated with maze1 as the rest time increased. In the post2 rest session, replay events seem to be more frequent, and most of them are associated with maze2 context.

5 Conclusion

As the results have shown, without referring to any external variables, P-GPLVM is a powerful tool to capture nonlinear neural population dynamics in the hippocampus by discovering the low-dimensional structure

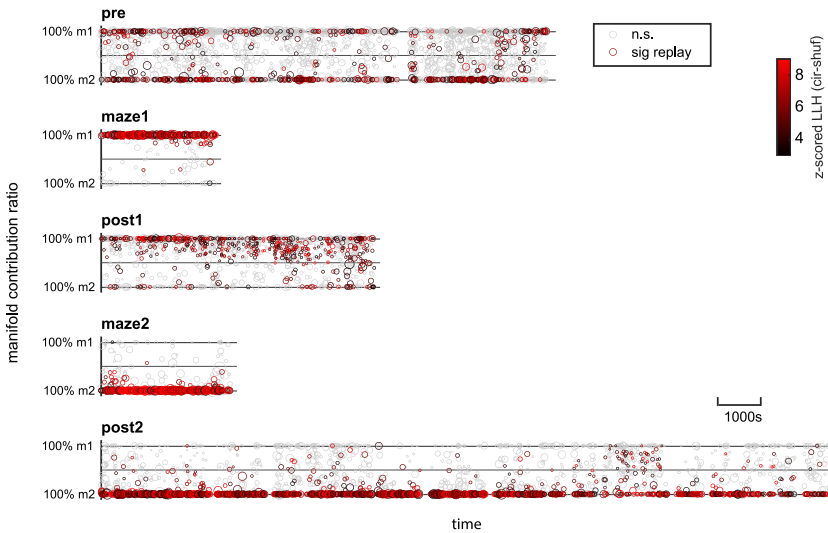

Figure 7: PBE occurrence time versus ratio of manifold contribution for PBE latent trajectories in all sessions. The top solid line indicates 100% of the trajectory nearest neighbors are from the maze1 manifold (m1). The bottom solid line indicates 100% of the nearest neighbors are from the maze2 manifold (m2). Each circle represents one detected PBE, whose size indicates its spatial consistency value. Significant replay events are colored in black to light red according to their z-scored LLH values among their circular-shuffled versions.

underlying the neural activity, which reveals important encoded external variables such as the number of contexts and the topology of animal spatial behavior (i.e., position).

In this article, this model is extended to leverage those advantages in the neural decoding scenario by enabling the constrained latent variable inference of new neural data and proposing a family of analyses for result evaluations. This constrained inference approach requires much less computational cost than relearning the model with training data combining old and new data. This extended model is flexible in that for new neural data during either running or PBEs, neural trajectories can be inferred in the latent space learned from training data during running. For the first time, internal neural pattern reactivation can be verified in a neural-process-specific manner without the need for explicit comparison with the neural activity known to belong to certain neural processes. External variables can then be decoded based on the external experiences corresponding to the repeated neural patterns. Metrics are defined that for the first time enable the identification of versatile continuously decoded replay using a model trained only with neural activity (and refine this definition to

allow for both sequential and nonsequential events). Given the accessible probabilistic internal tuning curves, this model can also provide insights for research looking into cell ensemble composition.

Acknowledgments

This research was supported by R01NS115233.

References

Alme, C. B., Miao, C., Jezek, K., Treves, A., Moser, E. I., & Moser, M.-B. (2014). Place cells in the hippocampus: Eleven maps for eleven rooms. *Proceedings of the National Academy of Sciences*, *111*(52), 18428–18435. 10.1073/pnas.1421056111

Cai, D. J., Aharoni, D., Shuman, T., Shobe, J., Biane, J., Song, W., et al. (2016). A shared neural ensemble links distinct contextual memories encoded close in time. *Nature*, *534*(7605), 115–118. 10.1038/nature17955

Csicsvari, J., O'Neill, J., Allen, K., & Senior, T. (2007). Place-selective firing contributes to the reverse-order reactivation of ca1 pyramidal cells during sharp waves in open-field exploration. *European Journal of Neuroscience*, *26*(3), 704–716. 10.1111/j.1460-9568.2007.05684.x

Davidson, T. J., Kloosterman, F., & Wilson, M. A. (2009). Hippocampal replay of extended experience. *Neuron*, *63*(4), 497–507. 10.1016/j.neuron.2009.07.027

Diba, K., & Buzsáki, G. (2007). Forward and reverse hippocampal place-cell sequences during ripples. *Nature Neuroscience*, *10*(10), 1241–1242. 10.1038/nn1961

Eichenbaum, H. (2017). The role of the hippocampus in navigation is memory. *Journal of Neurophysiology*, *117*(4), 1785–1796. 10.1152/jn.00005.2017

Eichenbaum, H., & Cohen, N. J. (2014). Can we reconcile the declarative memory and spatial navigation views on hippocampal function? *Neuron*, *83*(4), 764–770. 10.1016/j.neuron.2014.07.032

Foster, D. J., & Wilson, M. A. (2006). Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature*, *440*(7084), 680–683. 10.1038/nature04587

Harrison, M. T., Amarasingham, A., & Kass, R. E. (2013). of synchronous spiking. *Spike Timing: Mechanisms and Function*, *53*, 77–120.

Karlsson, M. P., & Frank, L. M. (2009). Awake replay of remote experiences in the hippocampus. *Nature Neuroscience*, *12*(7), 913–918. 10.1038/nn.2344

Krause, E. L., & Drugowitsch, J. (2022). A large majority of awake hippocampal sharp-wave ripples feature spatial trajectories with momentum. *Neuron*, *110*(4), 722–733.e8. 10.1016/j.neuron.2021.11.014

Kubie, J. L., Levy, E. R., & Fenton, A. A. (2020). Is hippocampal remapping the physiological basis for context? *Hippocampus*, *30*(8), 851–864. 10.1002/hipo.23160

Kudrimoti, H. S., Barnes, C. A., & McNaughton, B. L. (1999). Reactivation of hippocampal cell assemblies: Effects of behavioral state, experience, and EEG dynamics. *Journal of Neuroscience*, *19*(10), 4090–4101. 10.1523/JNEUROSCI.19-10-04090.1999

Lee, A. K., & Wilson, M. A. (2002). Memory of sequential experience in the hippocampus during slow wave sleep. *Neuron*, *36*(6), 1183–1194. 10.1016/S0896-6273(02)01096-6

Moser, M.-B., Rowland, D. C., & Moser, E. I. (2015). Place cells, grid cells, and memory. *Cold Spring Harbor Perspectives in Biology*, *7*(2), a021808. 10.1101/cshperspect.a021808

Nádasdy, Z., Hirase, H., Czurkó, A., Csicsvari, J., & Buzsáki, G. (1999). Replay and time compression of recurring spike sequences in the hippocampus. *Journal of Neuroscience*, *19*(21), 9497–9507.

Nieh, E. H., Schottdorf, M., Freeman, N. W., Low, R. J., Lewallen, S., Koay, S. A., . . . Tank, D. W. (2021). Geometry of abstract learned knowledge in the hippocampus. *Nature*, *595*(7865), 80–84. 10.1038/s41586-021-03652-7

Rubin, A., Sheintuch, L., Brande-Eilat, N., Pinchasof, O., Rechavi, Y., Geva, N., & Ziv, Y. (2019). Revealing neural correlates of behavior without behavioral measurements. *Nature Communications*, *10*(1), 1–14. 10.1038/s41467-018-07882-8

Skaggs, W. E., & McNaughton, B. L. (1996). Replay of neuronal firing sequences in rat hippocampus during sleep following spatial experience. *Science*, *271*(5257), 1870–1873. 10.1126/science.271.5257.1870

Whittington, J. C., McCaffary, D., Bakermans, J. J., & Behrens, T. E. (2022). How to build a cognitive map. *Nature Neuroscience*, *25*(10), 1257–1272. 10.1038/s41593-022-01153-y

Wilson, M. A., & McNaughton, B. L. (1994). Reactivation of hippocampal ensemble memories during sleep. *Science*, *265*(5172), 676–679. 10.1126/science.8036517

Wu, A., Roy, N. A., Keeley, S., & Pillow, J. W. (2017). Gaussian process based nonlinear latent structure discovery in multivariate spike train data. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, *30*. Curran.

Yu, B. M., Cunningham, J. P., Santhanam, G., Ryu, S. I., Shenoy, K. V., & Sahani, M. (2009). Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of Neurophysiology*, *102*(1), 614–635. 10.1152/jn.90941.2008

Received October 30, 2023; accepted March 28, 2024.