Graph Neural Networks for Connectivity Inference in Spatially Patterned Neural Responses

Taehoon Park GKSDIDG@HANYANG.AC.KR
Juhyeon Kim WNGUS1310@HANYANG.AC.KR
Donghee Kang EHDGML1312@HANYANG.AC.KR
Kijung Yoon KIYOON@HANYANG.AC.KR
Hanyang University, Seoul, 04763, Korea

Editors: Sophia Sanborn, Christian Shewmake, Simone Azeglio, Arianna Di Bernardo, Nina Miolane

Abstract

A continuous attractor network is one of the most common theoretical framework for studying a wide range of neural computations in the brain. Many previous approaches have attempted to identify continuous attractor systems by investigating the state-space structure of population neural activity. However, establishing the patterns of connectivity for relating the structure of attractor networks to their function is still an open problem. In this work, we propose the use of graph neural networks combined with the structure learning for inferring the recurrent connectivity of a ring attractor network and demonstrate that the developed model greatly improves the quality of circuit inference as well as the prediction of neural responses compared to baseline inference algorithms.

Keywords: continuous attractor networks, connectivity inference, graph neural networks

1. Introduction

Continuous attractor networks (CAN) have become some of the most influential circuit models in systems neuroscience. A CAN defined by a symmetric recurrent connectivity leads to spontaneous and stable neural activity patterns and the stable states can be used to represent numerous external variables such as the orientation of visual stimuli (Ben-Yishai et al., 1995), head direction (Zhang, 1996), eye position (Seung, 1996), and spatial location (Fuhs and Touretzky, 2006; Burak and Fiete, 2009). The CAN's wide range of utility has spurred several computational studies to find evidence of attractor dynamics in the brain.

The classic approach for identification of continuous attractors is to probe for the invariance of correlation structures in neural population activity (Taube et al., 1990; Yoganarasimha et al., 2006; Fyhn et al., 2007; Yoon et al., 2013; Gardner et al., 2019; Trettel et al., 2019). The invariance is an essential condition because the states of the system are internally generated by strong recurrent connections, which constrain the activity of neurons to form a certain coactivation pattern; thus, pairwise cell-cell relationships should remain invariant across time, environmental conditions, and behavioral states, independent of external sensory inputs (Burak and Fiete, 2009). Another recent attempt to address this challenge has been made to directly characterize the full state-space of the population responses and visualize them by topological data analysis (Low et al., 2018; Chaudhuri et al., 2019; Rybakken et al., 2019; Gardner et al., 2022). The resulting topological structure of

© 2022 T. Park, J. Kim, D. Kang & K. Yoon.

the attractor manifold provides the hints of an underlying circuit mechanism, and its invariance across the aforementioned external factors validates the main predictions of CAN models.

Nevertheless, few approaches are more reliable for identifying CAN than directly measuring the wiring diagrams of neural circuits. Hence, estimating the network connectivity from large population recordings has also been a long-standing problem (Paninski et al., 2003; Schneidman et al., 2006; Pillow et al., 2008). A fundamental challenge in circuit inference is that the mapping from circuits to activity is not injective: many neural circuits can generate the same activity states (Das and Fiete, 2020; Powell et al., 2021; Curto and Morrison, 2019). Moreover, several previous statistical models with strong parametric assumptions limit their ability to recover direct synaptic connections; thus, the inferred connectivity could be substantially different from the true connections, especially when neurons are highly correlated (Das and Fiete, 2020).

Here, we take a new approach to inferring structural connectivity with no stringent parametric assumptions based on neural spike data generated from a strongly recurrent ring attractor network. We adopt the framework of graph neural networks (GNNs) (Hamilton et al., 2017; Bronstein et al., 2017; Battaglia et al., 2018) combined with the structure learning module, an architectural design strategy that relies on learning the underlying circuitry jointly with predicting multi-neuronal spike responses. We show that this approach can be significantly more expressive than statistically sophisticated inference algorithms. Furthermore, we propose the use of an iterative neural message-passing scheme (Gilmer et al., 2017) that is closely matched to the generative nature of CAN. Taken together, these ideas suggest new possibilities for less biased inference of recurrent neural circuits.

2. Methods

2.1. Generative Model

We consider a ring network of 100 neurons where the process of generating neural spikes is by either a simple threshold-crossing or simulating an inhomogeneous Poisson process (Appendix A). The generative network model is designed to have rotation-invariant recurrent connections with a local Mexican hat weight profile and exhibit spatially periodic activity patterns drifting over time (Figure 1a)[1]. We record spikes from the network for 8 minutes with a time step $\Delta t = 0.1$ ms and represent the spike trains via $\mathbf{x} \in \{0,1\}^{N \times L}$, where N is the number of neurons and L is the length of time steps (Figure 1a). The data is split into 80%, 10%, and 10% over time for training, validation and testing.

2.2. Inference Model

Structure learning module The goal of this module is to model the pairwise connection strength $\mathbf{w}_{ij}$ given every pair of $(\mathbf{x}_i, \mathbf{x}_j)$ where $\mathbf{x}_i = (\mathbf{x}_i^1, \ldots, \mathbf{x}_i^L)$ denotes the spike train of ith neuron. We apply a 1D convolution f_{Conv1D} with 10 kernels of size 200 [2] and a stride of 20 over each input spike train, vectorize the feature maps along time dimension, and use

1. This generative model is equivalent to the highly structured ring network in the strongest weight regime by Das and Fiete (2020) when the scalar weight strength parameter $r = 0.025$.
2. The kernel size is set twice as long as the synaptic time constant ($\tau = 10$ ms) of the generative model.

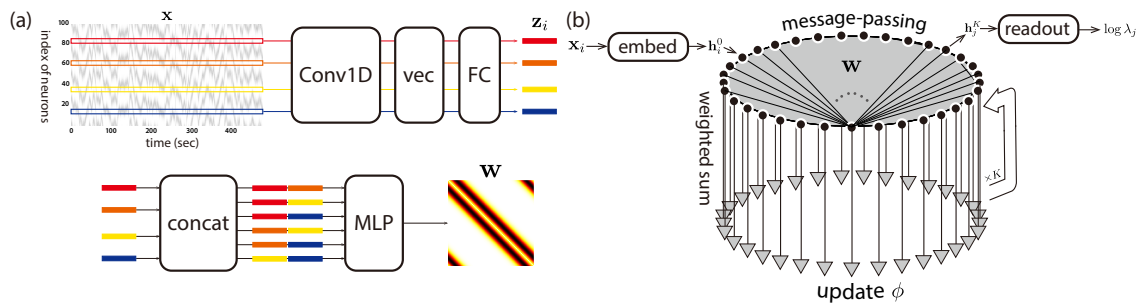

Figure 1: **Schematics of the proposed inference model. (a)** Structure learning module, **(b)** Spike prediction module.

a fully connected layer f_{out} to get a reduced output embedding vector $\mathbf{z}_i$:

$$\mathbf{z}_i = f_{\text{out}}\left(\text{vec}\left(f_{\text{Conv1D}}(\mathbf{x}_i)\right)\right) \tag{1}$$

Aside from these layers, we use batch normalization (Ioffe and Szegedy, 2015) placed after `ReLU` activation function. We then concatenate every pair of $(\mathbf{z}_i, \mathbf{z}_j)$ and apply a multilayer perceptron (MLP) that has two hidden layers with 32 units to infer the strength of network connectivity:

$$\mathbf{w}_{ij} = \text{MLP}(\mathbf{z}_i \parallel \mathbf{z}_j) \tag{2}$$

The resulting inferred weight matrix $\mathbf{w} \in \mathbb{R}^{N \times N}$ is assumed to be symmetric and have no self-loop. This constraint is weaker than the rotation-invariance of target recurrent weight strengths. In simple terms, every neuron in the proposed inference model does not have to share the same outgoing synaptic weights to all of the other neurons around the ring network.

Spike prediction module The task of this module is to predict the spike activity $\mathbf{x}$ for every neuron given the latent circuitry $\mathbf{w}$:

$$p_\theta(\mathbf{x}|\mathbf{w}) = \prod_{t=1}^{T} p_\theta\left(\mathbf{x}^{t+1}|\mathbf{x}^t, \ldots, \mathbf{x}^1, \mathbf{w}\right) \tag{3}$$

where $\mathbf{x}^t = (\mathbf{x}_1^t, \ldots, \mathbf{x}_N^t)$ represents the spike activity of all N neurons at time t. We assume that the attractor dynamics can be implicitly learned by modeling instantaneous neural spike rates $\boldsymbol{\lambda}^{t+1}$ where $\text{Poisson}(\boldsymbol{\lambda}^{t+1}) = p_\theta\left(\mathbf{x}^{t+1}|\mathbf{x}^t, \ldots, \mathbf{x}^1, \mathbf{w}\right)$ through the message-passing operations:

$$\mathbf{h}_{i,t}^0 = g_{\text{emb}}\left(\mathbf{x}_i^t, \ldots, \mathbf{x}_i^{t-2\tau/\Delta t+1}\right) \tag{4}$$

$$\mathbf{h}_{i,t}^{k+1} = \phi\left(\mathbf{h}_{i,t}^k,\ \mathbf{h}_{i,t}^0 \,\middle\|\, \sum_{j \neq i} \mathbf{w}_{ij} \cdot \psi\left(\mathbf{h}_{i,t}^k \parallel \mathbf{h}_{j,t}^k\right)\right) \tag{5}$$

$$\log\left(\lambda_i^{t+1}\right) = g_{\text{out}}\left(\mathbf{h}_{i,t}^K\right) \tag{6}$$

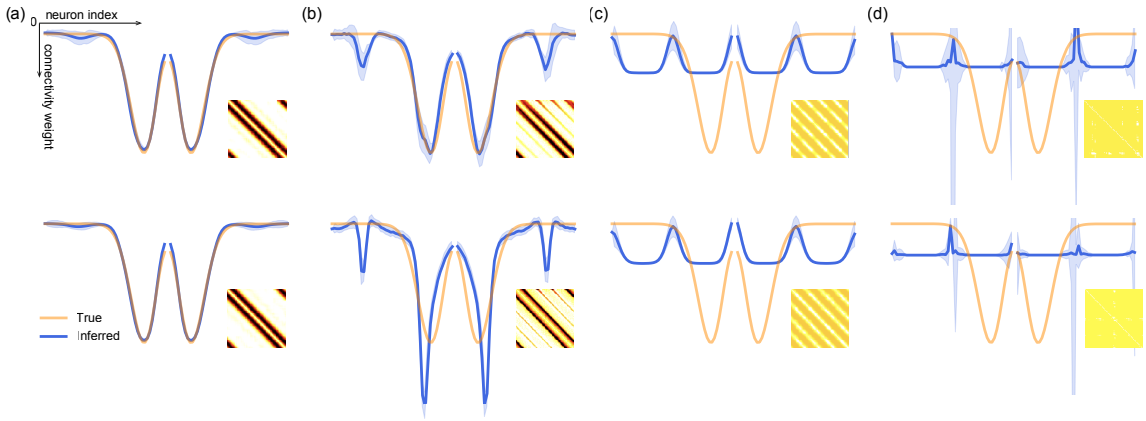

Figure 2: **Quality of circuit inference using 4.8×10^6 spikes from a strongly recurrent ring attractor network. (a)** True Mexican hat-shaped connectivity weight (orange) superimposed by the average inferred weight profile by GNN (blue). The inset at the bottom right presents the inferred weight matrix $\hat{\mathbf{W}}$. The model performs inference on spike data generated by threshold-crossing (top) and linear-nonlinear-poisson (bottom) models. The followings are the quality of inference by **(b)** GLM, **(c)** seqNMF, and **(d)** TCA.

where $\mathbf{h}_{i,t}^0$ is the initial embedding of the ith neuron's spike history for the past $\frac{\tau}{\Delta t}(= 200)$ steps at time t. At kth step of message passing, we compute the message sent from neuron j to i through ψ and a function ϕ combines the sum of incoming messages weighted by connection strength $\mathbf{w}_{ij}$ with the previous latent state $\mathbf{h}_{i,t}^k$ to generate the updated neuronal state $\mathbf{h}_{i,t}^{k+1}$. After K rounds of message passing, we use g_{out} to read out the log-firing rate from the output of the final propagation step (for our spike trains, K=1 was enough[3]). This firing rate determines the rate of an inhomogeneous Poisson process that generates spikes in neuron i at time $t+1$. In this experiment, we use separate two-layer MLPs for ψ and g_{out}, a fully connected layer for g_{emb}, and the gated recurrent unit is applied to ϕ.

2.3. Training

The training loss is the negative log-likelihood of the recorded population activity and the objective is to predict the spike times accurately by minimizing the likelihood using the Adam optimizer with a learning rate of 5×10^{-4}, decayed by a factor of 0.5 every 50 epochs:

$$\mathbf{\Theta}^*, \mathbf{w} = \arg\min_{\mathbf{\Theta}} \sum_{i=1}^{N} \sum_{t=1}^{L} \left(\lambda_i^t - \mathbf{x}_i^t \log \lambda_i^t \right) \tag{7}$$

where the parameters $\mathbf{\Theta}$ are the weights of the structure learning and spike prediction network modules, and $\mathbf{w}$ is the best inferred connectivity weight matrix.

3. Our task is not multi-step but single-step spike predictions because the spatially patterned neural responses are not driven by an external input but drift around the ring over time. One possible issue with the single step prediction is that the inferred weights cannot have a considerable effect on short-term dynamics. Meanwhile, a large number of message-passing iterations may reduce the effect of structure learning module and for this reason we set K to 1.

Table 1: Quality of spike prediction $\mathcal{L}$ and inference error Δ

generative model(→)	threshold-crossing		linear-nonlinear-poisson	
inference model(↓)	$\mathcal{L}$	Δ	$\mathcal{L}$	Δ
TCA	-	0.7619	-	0.7612
seqNMF	-	0.7885	-	0.7963
GLM	0.0409	0.2441	0.0430	0.4841
Ours	**0.0322**	**0.0614**	**0.0334**	**0.0487**

3. Experiments

To evaluate the quality of inferred connectivity, the outgoing weight profiles $\mathbf{w}$ are realigned by minimizing the gap (ℓ_1-norm) between the estimated weights of all neurons. We then compute the average weight profile $\bar{\mathbf{w}}$ and rescale $\mathbf{w}$ by minimizing ℓ_1 deviation between $\bar{\mathbf{w}}$ and $\mathbf{W}$ to match the scale of the true weights $\mathbf{W}$. It is necessary because the message function ψ in Equation (5) may compensate for an arbitrary scale of $\mathbf{w}$. Notably, the estimated bias during rescaling is an order of magnitude smaller than the rescaling factor, which suggests that the type of inferred connectivity (i.e. excitatory or inhibitory interaction) is in good agreement with the ground truth.

We next use the following two metrics to compare the proposed approach to three baselines[4]: i) the normalized inference error, computed as $\Delta = \|\mathbf{W} - \hat{\mathbf{W}}\|_F / \|\mathbf{W}\|_F$ where $\hat{\mathbf{W}}$ is the scaled version of $\mathbf{w}$ and $\|\cdot\|_F$ denotes the Frobenius-norm, and ii) the log-likelihood score $\mathcal{L}$ in Equation (7). Every alternative inference method exhibits an error biased toward inferring unconnected neurons, evidenced by side bumps in the weight profile or by multiple off-diagonal stripes in the weight matrix (Figures 2b-2d). This type of systematic inference error is interpreted as overestimated connections resulting from highly correlated activity by strongly recurrent networks (Das and Fiete, 2020). Despite the failure to explain away strong correlations, the proposed inference model alleviates the side bumps (Figure 2a) and thus significantly reduces biased errors (Table. 1). More accurate estimation of spike activity in our model (Table. 1) suggests that the GNN-based spike prediction module is expressive enough to learn and match actual neural dynamics of a ring attractor network.

4. Conclusion

We presented a GNN-based neural inference model to understand the network mechanism of neural circuits by reconstruction of structural connectivity from population neural spike data. We demonstrated that the proposed model is highly effective at inferring direct connections of a strongly recurrent network. It will be interesting to test this model framework by examining real population recording data, and to see whether the inferred circuitry supports the continuous attractor dynamics in the brain.

4. The baselines are generalized linear model (GLM) (Pillow et al., 2008), sequence non-negative matrix factorization (seqNMF) (Mackevicius et al., 2019), and tensor component analysis (TCA) (Williams et al., 2018) (see more details in Appendix B). TCA and seqNMF are not the models for circuit inference but aim to extract low-dimensional representations and dynamics of neural data. Nonetheless, we have run those models to identify low-dimensional neuron factors (i.e. features) and obtain their correlation structure since the correlation structure offer a crude estimate of network connectivity.

References

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

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

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

Yoram Burak and Ila R Fiete. Accurate path integration in continuous attractor network models of grid cells. *PLoS Computational Biology*, 5(2):e1000291, 2009.

Rishidev Chaudhuri, Berk Gerçek, Biraj Pandey, Adrien Peyrache, and Ila Fiete. The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature Neuroscience*, 22(9):1512–1520, 2019.

Carina Curto and Katherine Morrison. Relating network connectivity to dynamics: opportunities and challenges for theoretical neuroscience. *Current Opinion in Neurobiology*, 58:11–20, 2019.

Abhranil Das and Ila R Fiete. Systematic errors in connectivity inferred from activity in strongly recurrent networks. *Nature Neuroscience*, 23(10):1286–1296, 2020.

Mark C Fuhs and David S Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *Journal of Neuroscience*, 26(16):4266–4276, 2006.

Marianne Fyhn, Torkel Hafting, Alessandro Treves, May-Britt Moser, and Edvard I Moser. Hippocampal remapping and grid realignment in entorhinal cortex. *Nature*, 446(7132): 190–194, 2007.

Richard J Gardner, Li Lu, Tanja Wernle, May-Britt Moser, and Edvard I Moser. Correlation structure of grid cells is preserved during sleep. *Nature Neuroscience*, 22(4):598–608, 2019.

Richard J Gardner, Erik Hermansen, Marius Pachitariu, Yoram Burak, Nils A Baas, Benjamin A Dunn, May-Britt Moser, and Edvard I Moser. Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128, 2022.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017.

William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *IEEE Data Engineering Bulletin*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. PMLR, 2015.

Daniel Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13, 2000.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Ryan J Low, Sam Lewallen, Dmitriy Aronov, Rhino Nevers, and David W Tank. Probing variability in a cognitive map using manifold inference from neural dynamics. *BioRxiv*, page 418939, 2018.

Emily L Mackevicius, Andrew H Bahle, Alex H Williams, Shijie Gu, Natalia I Denisenko, Mark S Goldman, and Michale S Fee. Unsupervised discovery of temporal sequences in high-dimensional datasets, with applications to neuroscience. *Elife*, 8:e38471, 2019.

Liam Paninski, Eero Simoncelli, and Jonathan Pillow. Maximum likelihood estimation of a stochastic integrate-and-fire neural model. *Advances in Neural Information Processing Systems*, 16, 2003.

Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

Daniel J Powell, Eve Marder, and Michael P Nusbaum. Perturbation-specific responses by two neural circuits generating similar activity patterns. *Current Biology*, 31(21):4831–4838, 2021.

Erik Rybakken, Nils Baas, and Benjamin Dunn. Decoding of neural data using cohomological feature extraction. *Neural computation*, 31(1):68–93, 2019.

Elad Schneidman, Michael J Berry, Ronen Segev, and William Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440 (7087):1007–1012, 2006.

H Sebastian Seung. How the brain keeps the eyes still. *Proceedings of the National Academy of Sciences*, 93(23):13339–13344, 1996.

Paris Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *International Conference on Independent Component Analysis and Signal Separation*, pages 494–499. Springer, 2004.

Jeffrey S Taube, Robert U Muller, and James B Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. ii. effects of environmental manipulations. *Journal of Neuroscience*, 10(2):436–447, 1990.

Sean G Trettel, John B Trimper, Ernie Hwaun, Ila R Fiete, and Laura Lee Colgin. Grid cell co-activity patterns during sleep reflect spatial overlap of grid fields during active behaviors. *Nature Neuroscience*, 22(4):609–617, 2019.

Alex H Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I Ryu, Krishna V Shenoy, Mark Schnitzer, Tamara G Kolda, and Surya Ganguli. Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 98(6):1099–1115, 2018.

Doreswamy Yoganarasimha, Xintian Yu, and James J Knierim. Head direction cell representations maintain internal coherence during conflicting proximal and distal cue rotations: comparison with hippocampal place cells. *Journal of Neuroscience*, 26(2):622–631, 2006.

KiJung Yoon, Michael A Buice, Caswell Barry, Robin Hayman, Neil Burgess, and Ila R Fiete. Specific evidence of low-dimensional continuous attractor dynamics in grid cells. *Nature Neuroscience*, 16(8):1077–1084, 2013.

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

Appendix A. Generative network models

We construct a ring network of $N = 100$ neurons where the outgoing synaptic weights $\mathbf{W}_{ij}$ around the ring are defined by a local Mexican hat profile:

$$\mathbf{W}_{ij} = e^{-d_{ij}^2/2\sigma_1^2} - ae^{-d_{ij}^2/2\sigma_2^2} \tag{8}$$

where d_{ij} is the distance between neurons i and j, $\sigma_1 = 6.98$, and $\sigma_2 = 7$ (in units of neuron index). All inhibitory recurrent weights by setting $a = 1.0005 > 1$ allow the pattern formation and dynamical stability.

Threshold-crossing model The input to each neuron at time step t is given by

$$\mathbf{g}(t) = r\mathbf{W}\mathbf{s}(t) + \mathbf{b}(t) \tag{9}$$

where $\mathbf{s}(t)$ is the vector of synaptic activations, $\mathbf{W}$ is the $N \times N$ matrix of recurrent connectivity defined by Equation (8), and $\mathbf{b}(t) = 10^{-3}(1 + \xi(t))$ is the feed-forward inputs. $\xi(t)$ is a white Gaussian noise per neuron, with zero mean and s.d. $\sigma_\xi = 0.3$, resulting in a Poisson-like variance proportional to the mean activation. We set the weight strength $r = 0.025$. The neurons in this generative model emit a spike when the input $g_i(t)$ to neuron i at time step t exceeds a threshold θ. The synaptic activation from neurons that just spiked is incremented by 1 and otherwise decays exponentially with time constant $\tau = 10$ ms according to the following equation:

$$\mathbf{s}(t + \Delta t) = \mathbf{s}(t)\left(1 - \frac{\Delta t}{\tau}\right) + \sigma(t) \tag{10}$$

where $\sigma(t)$ is the binary vector of spikes from the network.

Linear-nonlinear Poisson model The inputs to this model are determined by Equation (9) except that the external input $\mathbf{b}(t) = 10^{-3}$ that is constant with no noise. The neural firing rate $\lambda_i(t)$ is determined by passing the summed input $g_i(t)$ to the rectifying nonlinear activation function:

$$\lambda_i(t) = \lambda_0 \, \texttt{ReLU}\,[\, g_i(t) - \theta \,] \tag{11}$$

These firing rates determine the rate of an inhomogeneous Poisson process that generates $n_i(t)$ spikes in neuron i at time t.

$$n_i(t) \sim \mathbf{Pois}(\lambda_i(t)) \tag{12}$$

The synaptic activations are determined by Equation (10) except that $\sigma_i(t)$ is replaced by $n_i(t)$.

Appendix B. Baseline inference algorithms

Algorithm 1 Generalized Linear Model

```
Initialize the model parameters by setting b = 0, w = 0, s = 1
for i ∈ [1, N] do
    for k ∈ [1, 2000] do
        Coupling filter a = A_basis · s
        if i > 1 then
            circular shift w by i steps
        end if
        x_i = X_{i,L+1:end}
        X_rest = X_{[1:i−1,i+1:N],1:L−1}
        x_w = X_rest · w
        log λ_i = x_w ∗ a + b
        L = Σ_t Σ_i (λ_i^t − x_i^t log λ_i^t)
        b, w, s = arg min_{b,w,s}(L) by Quasi-Newton Method
    end for
end for
```

$\mathbf{A}_{\text{basis}}$ is a 200×10 matrix whose columns represent ten different raised cosine filters. Coupling filter $\mathbf{a}$ is constructed by the weighted sum of the basis. Filtered spike train $\mathbf{x_w}$ is obtained by multiplying $\mathbf{X}_{\text{rest}}$ and $\mathbf{w}$. We model log of neural spike rate as the convolution of $\mathbf{a}$ and $\mathbf{x_w}$ with bias b, where $*$ is discrete colvolution. Model parameters are updated by Quasi-Newton method.

Algorithm 2 Tensor Component Analysis (TCA)

Initialize matrices of factors $\mathbf{N}, \mathbf{K}, \mathbf{T}$ (Neuron, Trial and Time factors respectively)
$\mathbf{N} \in \mathbb{R}^{N \times R}, \mathbf{K} \in \mathbb{R}^{K \times R}, \mathbf{T} \in \mathbb{R}^{T \times R}$
The number of neurons N, components R, total time steps T, length of sequence L
$\mathbf{M} = \mathrm{EW}\,(\mathbf{X})$
EW: exponentially weighted moving average
Reshape $\mathbf{M} \in \mathbb{R}^{N \times L}$ to $\mathbb{R}^{N \times K \times T}$, $T = L/K$
for $k \in [1, 500]$ **do**
$\mathbf{N}_{:,r}, \mathbf{K}_{:,r}, \mathbf{T}_{:,r} = \underset{\mathbf{N}_{:,r}, \mathbf{K}_{:,r}, \mathbf{T}_{:,r}}{\arg\min} \; (\|\mathbf{M} - \sum_r \mathbf{N}_{:,r} \otimes \mathbf{K}_{:,r} \otimes \mathbf{T}_{:,r}\|_F)$,
by Alternating Least-Square Method
end for
for $r \in [1, R]$ **do**
$\mathbf{Z}^{(r)} = \mathbf{N}_{:,r} \otimes \mathbf{N}_{:,r} \otimes \mathbf{K}_{:,r} \otimes \mathbf{T}_{:,r}$
$\mathbf{E}^{(r)} = \sum_{t=1}^{T} \sum_{k=1}^{K} \left(\mathbf{Z}^{(r)}_{:,:,k,t} \right)$
end for
$\mathbf{E} = \frac{1}{R} \sum_r \left(\mathbf{E}^{(r)} \right)$

TCA (algorithm 2) is a three-dimensional extended version of principle component analysis, which factorizes the original tensor into $R = 100$ components of neuron $\mathbf{N} \in \mathbb{R}^{N \times R}$, trial $\mathbf{K} \in \mathbb{R}^{K \times R}$, and time $\mathbf{T} \in \mathbb{R}^{T \times R}$ factor matrices. First, we exponentially smooth spike train $\mathbf{X}$ along the time axis with smoothing factor $\alpha = 0.01$. Then smoothed spike train $\mathbf{M}$ is reshaped to have $K = 10$ trial components. Afterward, we apply TCA and take tensor outer product of each neuron factor $\mathbf{N}_{:,r}$ with itself followed by trial and time factors trying to find coupling strength with other neurons. Each factor is optimized with alternating least-square method. Finally, we take summation of the the result $\mathbf{Z}$ along trial and time axis. Overall predicted connectivity matrix $\mathbf{E}$ is obtained by averaging component-wise edge matrix $\mathbf{E}^{(r)}$.

Algorithm 3 Sequence Non-negative Matrix Factorization (seqNMF)

Initialize neuron factor tensor $\mathbf{W}$ and time factor matrix $\mathbf{H}$
  $\mathbf{W} \in \mathbb{R}^{N \times R \times L}$, $\mathbf{H} \in \mathbb{R}^{R \times T}$
  The number of neurons N, components R, total time steps T, length of sequence L
Define Smoothing Matrix $\mathbf{S} \in \mathbb{R}^{T \times T}$
  $\mathbf{S}_{i,j} = 1$ when $|i - j| < L$ otherwise 0
Define regularization parameter β
for $k \in [1, 100]$ **do**
  $\tilde{\mathbf{X}}_{i,t} = \sum_r \sum_l \mathbf{W}_{i,r,l} \mathbf{H}_{r,(t-l)}$
  $\mathbf{C}_{i,j} = \sum_l \sum_n \mathbf{W}_{n,i,l} \mathbf{X}_{n,j+l}$
  $\mathbf{W}, \mathbf{H} = \underset{\mathbf{W},\mathbf{H}}{\arg\min} \left(\left\| \tilde{\mathbf{X}} - \mathbf{X} \right\|_F + \beta \left\| \mathbf{C} \cdot \mathbf{S} \cdot \mathbf{H}^\top \right\|_{1, i \neq j} \right)$,
  by non-negative gradient descent (Lee and Seung, 1999, 2000; Smaragdis, 2004)
end for
for $r \in [1, R]$ **do**
  $\mathbf{Z}^{(r)} = \mathbf{W}_{:,r,0} \otimes \mathbf{W}_{:,r,1} \otimes \mathbf{H}_{r,:}$
  $\mathbf{E}^{(r)} = \sum_{t=1}^{T} \left(\mathbf{Z}^{(r)}_{:,:,t} \right)$
end for
$\mathbf{E} = \frac{1}{R} \sum_r \left(\mathbf{E}^{(r)} \right)$

SeqNMF (algorithm 3) is a dimension reduction method with regularized convolutional non-negative matrix factorization to extract sequential patterns. First, we define the smoothing matrix $\mathbf{S} \in \mathbb{R}^{T \times T}$, with the length of sequence $L = 2$ and regularization parameter $\beta = 0.001$. Then, we get neuron factors tensor $\mathbf{W} \in \mathbb{R}^{N \times R \times L}$ and time factors matrix $\mathbf{H} \in \mathbb{R}^{R \times T}$ by the method denoted in Algorithm 3, where $\| \cdot \|_{1, i \neq j}$ is the sum of absolute value of matrix exclude diagonal terms, and take tensor product of columns of learned neuron factors $\mathbf{W}_{:,r,0}$, $\mathbf{W}_{:,r,1}$ and time factor $\mathbf{H}_{r,:}$ to get edge features. Similar to TCA, we take summation along time axis of the result $\mathbf{Z}$ to get component-wise edge matrix. Overall predicted connectivity matrix $\mathbf{E}$ is obtained by averaging component-wise edge matrix $\mathbf{E}^{(r)}$. We use $R = 100$ components which is the same as TCA.