A Differentiable Partially Observable Generalized Linear Model with Forward-Backward Message Passing

Chengrui Li [1] **Weihan Li** [1] **Yule Wang** [1] **Anqi Wu** [1]

Abstract

The partially observable generalized linear model (POGLM) is a powerful tool for understanding neural connectivity under the assumption of existing hidden neurons. With spike trains only recorded from visible neurons, existing works use variational inference to learn POGLM meanwhile presenting the difficulty of learning this latent variable model. There are two main issues: (1) the sampled Poisson hidden spike count hinders the use of the pathwise gradient estimator in VI; and (2) the existing design of the variational model is neither expressive nor time-efficient, which further affects the performance. For (1), we propose a new differentiable POGLM, which enables the pathwise gradient estimator, better than the score function gradient estimator used in existing works. For (2), we propose the forward-backward message passing sampling scheme for the variational model. Comprehensive experiments show that our differentiable POGLMs with our forward-backward message passing produce a better performance on one synthetic and two real-world datasets. Furthermore, our new method yields more interpretable parameters, underscoring its significance in neuroscience. Code: `https://github.com/JerrySoybean/poglm`.

1. Introduction

Understanding neural connectivity is a critical research question in neuroscience. The generalized linear model (GLM) (Pillow et al., 2008) with its variants (Linderman et al., 2016; Roudi et al., 2015; Li et al., 2024a) form a mainstream set of tools for inferring connectivity from neural populations. However, a nonnegligible problem in neural recording is that the recorded neurons are only a small part of the entire population in a certain target region of our interests. A GLM for such an incomplete problem is referred to as a partially observable GLM (POGLM) (Pillow & Latham, 2007; Jimenez Rezende & Gerstner, 2014; Linderman et al., 2017), which considers both visible and hidden neurons.

The general goal of POGLM is to learn the model parameter set θ given only the spike trains $\boldsymbol{X}$ from visible neurons, especially the connectivity between neurons (including both visible and hidden neurons). The spike trains from hidden neurons $\boldsymbol{Z}$ is the latent variable in POGLM. Variational inference (VI) (Blei et al., 2017) is the most commonly used method for solving such a latent variable model. In VI, the target is to maximize the observable data's evidence lower bound

$$\begin{aligned} &\mathrm{ELBO}(\boldsymbol{X};\theta,\phi) \\ =&\mathbb{E}_{q(\boldsymbol{Z}|\boldsymbol{X};\phi)}\left[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta)-\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi)\right] \\ =&\ln p(\boldsymbol{X};\theta)-\mathrm{KL}(q(\boldsymbol{Z}|\boldsymbol{X};\phi)\|p(\boldsymbol{Z}|\boldsymbol{X};\theta)) \\ \leqslant&\ln p(\boldsymbol{X};\theta) \end{aligned} \tag{1}$$

w.r.t. θ and ϕ, where $p(\boldsymbol{X},\boldsymbol{Z};\theta)$ is the generative model and $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ is the variational model parameterized by ϕ approximating the posterior $p(\boldsymbol{Z}|\boldsymbol{X};\theta)$. Maximizing Eq. 1 requires sampling hidden spike train $\boldsymbol{Z}$ from $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ and computing the gradient estimator of $\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)$ w.r.t. θ and ϕ.

However, results from existing works demonstrated the difficulties of solving such a complicated model. Particularly, two issues have caught our attention:
(1) A good way of deriving $\frac{\partial\,\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\phi}$ is via the pathwise gradient estimator (Kingma & Welling, 2013), which will be hindered with a discrete $\boldsymbol{Z}$. The only alternative is the score function gradient estimator which usually has higher variance than the pathwise gradient estimator (Paisley et al., 2012; Bengio et al., 2013; Schulman et al., 2015).
(2) The sampling scheme of the variational model $\boldsymbol{Z}\sim q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ in most of the existing works is a GLM on hidden neurons (Jimenez Rezende & Gerstner, 2014; Kajino, 2021; Li et al., 2024b), which is only conditioned on history visible and sampled history hidden spikes. This design makes the sampling and inference procedure slow and omits

[1]School of Computational Science & Engineering, Georgia Institute of Technology, Atlanta, USA. Correspondence to: Chengrui Li <cnlichengrui@gatech.edu>, Anqi Wu <anqiwu@gatech.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

the influence of hidden spikes on future visible spikes.

Given these issues, our paper aims to solve these two limitations in the existing works and study the POGLM more comprehensively. In Sec. 2, we will propose a new differentiable POGLM that enables the pathwise gradient estimator for VI. We will also introduce different variational distribution families for sampling hidden spikes, especially our newly proposed forward-backward message passing. In Sec. 3, we will conduct extensive comparisons between combinations of different inference methods (including the original POGLM, our newly proposed differentiable POGLM, and other intermediate models) $\times$ different variational sampling schemes on a synthetic dataset and two real-world neural datasets. The results will demonstrate the superiorities of our differentiable POGLM and our forward-backward message passing sampling scheme.

2. Models

2.1. Background: POGLM

Generative model. We start from the partially observable generalized linear model (POGLM) (Pillow & Latham, 2007; Jimenez Rezende & Gerstner, 2014; Linderman et al., 2017) that studies the mutual interactions between neurons underlying the corresponding neural spike trains. Assume V of N neurons are visible and the remaining $H = N - V$ neurons are hidden. We denote $\boldsymbol{X} \in \mathbb{N}^{T\times V}$ as the observed spike train recorded from V visible neurons across T time bins, and $x_{t,v}$ as the spike count generated by the v-th visible neuron in the t-th time bin. $\boldsymbol{Z} \in \mathbb{N}^{T\times H}$ as the latent spike train recorded from H hidden neurons across T time bins, and $z_{t,h}$ as the spike count generated by the h-th hidden neuron in the t-th time bin. The complete generative model $p(\boldsymbol{X}, \boldsymbol{Z}; \theta)$ is depicted in Fig. 1(a) (Pillow et al., 2008). For a visible neuron v, its firing rate at time t is

$$f_{t,v} = \sigma\left(b_v + \sum_{v'=1}^{V} w_{v\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{h'=1}^{H} w_{v\leftarrow h'} \cdot \left(\sum_{l=1}^{L} z_{t-l,h'}\ \psi_l\right)\right), \tag{2}$$

and its spike count is generated by

$$x_{t,v} \sim \text{Poisson}(f_{t,v}). \tag{3}$$

$\sigma(\cdot)$ is a non-linear function (e.g., Softplus); $\boldsymbol{b}_V = [b_1, b_2, \ldots, b_V]^{\mathrm{T}} \in \mathbb{R}^V$ is the background intensity vector of the V visible neurons; $\boldsymbol{W}_{V\leftarrow V} = [w_{v\leftarrow v'}]_{V\times V} \in \mathbb{R}^{V\times V}$ is the weight matrix representing the weights from visible neurons to visible neurons; $\boldsymbol{W}_{V\leftarrow H} = [w_{v\leftarrow h'}]_{V\times H} \in \mathbb{R}^{V\times H}$ is the weight matrix representing the weights from hidden neurons to visible neurons;

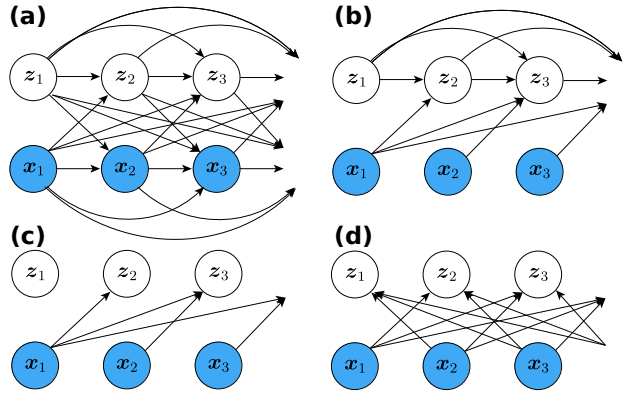

Figure 1. **(a)**: The generative model of the complete POGLM $p(\boldsymbol{X}, \boldsymbol{Z}; \theta)$. **(b)**, **(c)**, **(d)**: The forward-self, forward, and forward-backward sampling scheme of the variational model $q(\boldsymbol{Z}|\boldsymbol{X}; \phi)$.

$\boldsymbol{\psi} = [\psi_1, \psi_2, \ldots, \psi_L]^{\mathrm{T}} \in \mathbb{R}_+^L$ is the pre-defined basis function summarizing history spikes from $t - L$ to $t - 1$. Similarly, for a hidden neuron h, its firing rate at time t is

$$f_{t,h} = \sigma\left(b_h + \sum_{v'=1}^{V} w_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{h'=1}^{H} w_{h\leftarrow h'} \cdot \left(\sum_{l=1}^{L} z_{t-l,h'}\ \psi_l\right)\right), \tag{4}$$

and its spike count is generated by

$$z_{t,h} \sim \text{Poisson}(f_{t,h}) \tag{5}$$

with parameters $\boldsymbol{b}_H = [b_1, b_2, \ldots, b_H]^{\mathrm{T}} \in \mathbb{R}^H$; $\boldsymbol{W}_{H\leftarrow V} = [w_{h\leftarrow v'}]_{H\times V} \in \mathbb{R}^{H\times V}$; $\boldsymbol{W}_{H\leftarrow H} = [w_{h\leftarrow h'}]_{H\times H} \in \mathbb{R}^{H\times H}$.

Therefore, POGLM is a latent variable model whose learnable parameter set is $\theta = \{\boldsymbol{b}, \boldsymbol{W}\}$, where $\boldsymbol{b}$ and $\boldsymbol{W}$ can be presented in the form of block (partitioned) matrix/vector:

$$\boldsymbol{b} = \begin{bmatrix} \boldsymbol{b}_V \\ \boldsymbol{b}_H \end{bmatrix} \in \mathbb{R}^N, \quad \boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{V\leftarrow V} & \boldsymbol{W}_{V\leftarrow H} \\ \boldsymbol{W}_{H\leftarrow V} & \boldsymbol{W}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}. \tag{6}$$

Variational inference. Since POGLM is a latent variable model, the goal is to learn the model parameter θ while also inferring the latent variable $\boldsymbol{Z}$. Given the complicated nature of POGLM (Fig. 1(a)), there is no closed form posterior distribution $p(\boldsymbol{Z}|\boldsymbol{X}; \theta)$. Therefore, we need to choose a good variational model $q(\boldsymbol{Z}|\boldsymbol{X}; \phi)$ parameterized by ϕ to do variational inference (VI) (Blei et al., 2017). We will discuss different choices of the variational models in Sec. 2.3. Now, we use a simple homogeneous Poisson variational model for illustration. The firing rate of a hidden neuron at time t is given by

$$f_{t,h} = \sigma(c_h), \tag{7}$$

and the spike count is $z_{t,h} \sim \text{Poisson}(f_{t,h})$. The variational parameter set is $\phi = \{\boldsymbol{c}_H\}$, where $\boldsymbol{c}_H = [c_1, \ldots, c_H]^{\mathrm{T}}$.

With a selected variational model $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$, VI can be adopted. We maximize the model's evidence lower bound $\text{ELBO}(\boldsymbol{X};\theta,\phi)$ (Eq. 1) w.r.t. θ and ϕ, so that the learned θ could estimate the model's true parameter θ^{true} and the variational distribution $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ could approximate the unknown posterior distribution $p(\boldsymbol{Z}|\boldsymbol{X};\theta)$ of the latent variable $\boldsymbol{Z}$. Given the complicated form of the POGLM model (Fig. 1(a)), there is still no closed form of ELBO (Eq. 1). Hence, we need its numerical estimator

$$\begin{aligned}&\widehat{\text{ELBO}}(\boldsymbol{X};\theta,\phi)\\ =&\hat{\mathbb{E}}_{q(\boldsymbol{Z}|\boldsymbol{X};\phi)}[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta) - \ln q(\boldsymbol{Z}|\boldsymbol{X};\phi)]\\ =&\frac{1}{K}\sum_{k=1}^{K}\left[\ln p\left(\boldsymbol{X},\boldsymbol{Z}^{(k)};\theta\right) - \ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi\right)\right],\end{aligned} \tag{8}$$

where $\left\{\boldsymbol{Z}^{(k)}\right\}_{k=1}^{K}$ are K Monte Carlo samples from $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$. The derivative w.r.t. θ is simple (Appendix. A.1):

$$\frac{\partial\,\text{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\theta} \approx \frac{\partial}{\partial\theta}\widehat{\text{ELBO}}(\boldsymbol{X};\theta,\phi). \tag{9}$$

Since $\boldsymbol{Z} \in \mathbb{N}^{T\times H}$ are discrete spike counts from the hidden neurons, the derivative w.r.t. ϕ at a particular value ϕ_0 requires the score function gradient estimator (Appendix. A.1):

$$\begin{aligned}\frac{\partial\,\text{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\phi} \approx \frac{1}{K}\sum_{k=1}^{K}\Bigg\{&\left[\ln p\left(\boldsymbol{X},\boldsymbol{Z}^{(k)};\theta\right)\right.\\ &\left.-\ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi_0\right)\right]\frac{\partial}{\partial\phi}\ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi\right)\Bigg\}.\end{aligned} \tag{10}$$

However, previous literature shows that the score function gradient estimator for maximizing ELBO w.r.t. ϕ exhibits high variance (Paisley et al., 2012; Bengio et al., 2013; Kingma & Welling, 2013; Schulman et al., 2015), and hence it could be better to seek a reparameterization trick for sampling $\boldsymbol{Z}^{(k)}$ from $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ so that pathwise gradient estimator can be applied. Given there is no reparameterization trick for Poisson distribution, we have to relax the discrete latent variable $\boldsymbol{Z}$ into a continuous variable and reformulate a differentiable POGLM as follows.

2.2. A Differentiable POGLM

Relaxation for differentiability. In this subsection, we formulate a differentiable POGLM via the Gumbel-Softmax distribution (Jang et al., 2016; Maddison et al., 2016). We first set a large enough upper-bound M so that $z_{t,h} \in \{0, 1, \ldots, M-1\}$. Practically, M doesn't need to be very large since the number of spikes in a short enough time bin is limited. Usually, we hope the number of spikes in each time bin to be very small (most of them should be 0 or 1) so that the precision of the spike train can be preserved. Without loss of generality, we use $M = 5$ in our following experiments. Then, a categorical distribution can be used to approximate the corresponding Poisson distribution:

$$z_{t,h} \sim \text{Cat}(\boldsymbol{\pi}(f_{t,h})), \tag{11}$$

where

$$\boldsymbol{\pi}(f) = \left(1 - \sum_{m=1}^{M-1}\frac{f^m \mathrm{e}^f}{m!}, \frac{f^1\mathrm{e}^f}{1!}, \ldots, \frac{f^{M-1}\mathrm{e}^f}{(M-1)!}\right) \tag{12}$$

expands the Poisson distribution truncated at M. Then, we can use Gumbel-Softmax (GS) to relax the discrete $z_{t,h}$ into a soft one-hot version

$$\tilde{\boldsymbol{z}}_{t,h} = (\tilde{z}_{t,h,0}, \ldots, \tilde{z}_{t,h,M-1}) \sim \text{GS}(\boldsymbol{\pi}(f_{t,h});\tau), \tag{13}$$

where $\tilde{\boldsymbol{z}}_{t,h}$ is a soft one-hot vector over a simplex $\Delta^{M-1} := \left\{\boldsymbol{z} \in [0,1]\middle|\sum_{m=0}^{M-1} z_m = 1\right\}$. Specifically,

$$\tilde{z}_{t,h,m} = \frac{\exp[(\ln\pi_m(f_{t,h}) + g_{t,h,m})/\tau]}{\sum_{m'=0}^{M-1}\exp[(\ln\pi_{m'}(f_{t,h}) + g_{t,h,m'})/\tau]}, \tag{14}$$

where $g_{t,h,m} \overset{\text{i.i.d.}}{\sim} \text{Gumbel}(0,1)$. In practice, we can sample g by sampling u from $\text{Uniform}(0,1)$ and computing $g = -\ln(-\ln(u))$. $\tau > 0$ is a temperature hyperparameter forcing $\tilde{\boldsymbol{z}}_{t,h}$ to be a soft one-hot representation closing to a corner of the simplex Δ^{M-1}. When $\tau \to 0$, $\tilde{\boldsymbol{z}}_{t,h}$ becomes the hard one-hot representation of the spike count $z_{t,h}$. It is common to choose the temperature τ in Gumbel-Softmax from $[0.1, 1]$. If τ is too large, the relaxation will be too soft; if τ is too small, numerical issues could arise. In our model, τ is used to force the soft one-hot close to one corner of the simplex, so we tried $\tau \in \{0.2, 0.5, 1\}$ and found $\tau = 0.5$ is an optimal choice that gives good and stable categorical approximation without numerical issue. We fix $\tau = 0.5$ in this differentiable model in our experiments, which is a common moderate choice. More details about the Gumbel-Softmax distribution including its likelihood function are in Jang et al. (2016); Maddison et al. (2016).

Generative and variational model. Given the soft one-hot $\tilde{z}_{t,h,m}$, we next define the equivalent soft hidden spike count as

$$z_{t,h} = \sum_{m=0}^{M-1} m \cdot \tilde{z}_{t,h,m}. \tag{15}$$

Now we are ready to define the complete differentiable generative model $p\left(\boldsymbol{X}, \tilde{\boldsymbol{Z}};\theta\right)$. Visible neurons' spikes $\boldsymbol{X}$ are generated with Eq. 2 ($f_{t,v}$) and Eq. 3 (Poisson) where $z_{t,h}$ in Eq. 2 is now defined by the above Eq. 15. Latent variable $\tilde{\boldsymbol{Z}}$

are generated from Eq. 4 ($f_{t,h}$) and Eq. 13 (GS), instead of Eq. 4 and Eq. 5 (Poisson). Similarly, the sampling process of $\tilde{\boldsymbol{Z}}$ in the variational model $q\left(\tilde{\boldsymbol{Z}}\middle|\boldsymbol{X};\phi\right)$ changes from Eq. 7 ($f_{t,h}$) and Eq. 5 (Poisson) to Eq. 7 and Eq. 13 (GS). Now, the complete spike train of this differentiable POGLM is $\left\{\boldsymbol{X},\tilde{\boldsymbol{Z}}\right\}$ where $\tilde{\boldsymbol{Z}} \in \left(\Delta^{M-1}\right)^{T\times H} \subsetneqq [0,1]^{T\times H\times M}$.

Pathwise gradient estimator. With both the generative model and the variational model differentiable, the pathwise gradient estimator can be used for optimizing ϕ. Particularly, by the reparameterization trick in Eq. 14 abbreviated as $\tilde{\boldsymbol{Z}}|\boldsymbol{X};\phi = r(\boldsymbol{G}|\boldsymbol{X};\phi)$ where $\boldsymbol{G} \sim \mathrm{Gumbel}(\boldsymbol{G};0,1)$, we have the transformation relationship $q\left(\tilde{\boldsymbol{Z}}\middle|\boldsymbol{X};\phi\right)\,\mathrm{d}\tilde{\boldsymbol{Z}} = \mathrm{Gumbel}(\boldsymbol{G};0,1)\,\mathrm{d}\boldsymbol{G}$ (Schulman et al., 2015). Then, the pathwise gradient estimator of the derivative of ELBO w.r.t. ϕ is:

$$\frac{\partial\,\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\phi} \approx \frac{\partial}{\partial\phi}\widehat{\mathrm{ELBO}}(\boldsymbol{X};\theta,\phi), \tag{16}$$

and

$$\begin{aligned}&\widehat{\mathrm{ELBO}}(\boldsymbol{X};\theta,\phi)\\ =&\frac{1}{K}\sum_{k=1}^{K}\left[\ln p\left(\boldsymbol{X},\tilde{\boldsymbol{Z}}^{(k)};\theta\right)\right) - \ln q\left(\tilde{\boldsymbol{Z}}^{(k)}\middle|\boldsymbol{X};\phi\right)\right],\end{aligned} \tag{17}$$

where $\tilde{\boldsymbol{Z}}^{(k)} = r\left(\boldsymbol{G}^{(k)}\middle|\boldsymbol{X};\phi\right)$ and $\left\{\boldsymbol{G}^{(k)}\right\}_{k=1}^{K}$ are K Monte Carlo samples from $\mathrm{Gumbel}(\boldsymbol{G};0,1)$. The detailed derivation of this is shown in Appendix. A.1.

Relax to general continuous distributions. In fact, the differentiable POGLM introduced above is already compatible with any continuous distributions that satisfy the following two requirements: (1) the distribution is parameterized by a single mean parameter, since the GLM structure provides a single firing rate $f_{t,h}$ representing the mean statistic of the (equivalent soft) spike count $z_{t,h}$; and (2) a reparameterization trick should exist for sampling such a distribution. For example, in the generative and variational models, we can assume a soft hidden spike count is from an exponential distribution

$$z_{t,h} \sim \mathrm{Exp}(1/f_{t,h}) \tag{18}$$

with mean $f_{t,h}$ computed from Eq. 4 and 7. The sampling can be parameterized as

$$z_{t,h} = -f_{t,h}\ln(1-u), \quad u \sim \mathrm{Unif}(0,1). \tag{19}$$

Compared with the GS distribution in which the equivalent soft hidden spike count is close to an integer in $\{0,1,\ldots,M-1\}$, $z_{t,h}$ from the exponential distribution can be any value in $\mathbb{R}_{\geqslant 0}$. More details about the possible choices of the distributions are in Sec. 3 and Fig. 3.

2.3. Sampling scheme of the variational model

So far, we have proposed the differentiable POGLM to resolve the first issue mentioned in Sec. 1. Now, we turn to the second issue—the choice of the variational model. Specifically, we need to design the formula for $f_{t,h}$ in the variational model. Clearly, the homogeneous one we introduced in Eq. 7 is oversimplified so that the variational distribution family of $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ is very far from the posterior distribution $p(\boldsymbol{Z}|\boldsymbol{X};\theta)$. A good choice of the variational distribution family that is much closer to the true posterior distribution is critical to the success of VI. Here, we discuss five candidates as follows:

• **Homogeneous Poisson**: $f_{t,h} = \sigma(c_h)$, $\forall t \in \{1,\ldots,T\}$, and the variational parameter set is $\phi = \{\boldsymbol{c}_H\}$, where $\boldsymbol{c}_H = [c_1,\ldots,c_H]^{\mathrm{T}}$. However, this is too simple to serve as a variational distribution family in practice.

• **Inhomogeneous Poisson (mean-field)**: $f_{t,h} = \sigma(c_{t,h})$, and $\phi = \left\{\boldsymbol{C}_{T\times H} \in \mathbb{R}^{T\times H}\right\}$. Although the mean-field theory is widely applicable to a lot of latent variable models, it lacks the dependency on the visible spike train $\boldsymbol{X}$. For POGLM, this learned ϕ is only bonded to the training spike train $p(\boldsymbol{Z}_{\mathrm{train}}|\boldsymbol{X}_{\mathrm{train}};\theta) \approx q(\boldsymbol{Z}_{\mathrm{train}}|\boldsymbol{X}_{\mathrm{train}};\phi)$, but unable to be generalized to the test spike train $p(\boldsymbol{Z}_{\mathrm{test}}|\boldsymbol{X}_{\mathrm{test}};\theta) \not\approx q(\boldsymbol{Z}_{\mathrm{test}}|\boldsymbol{X}_{\mathrm{test}};\phi)$. Besides, both homogeneous and inhomogeneous Poisson have no message passing between neurons, and hence are very unhelpful for learning the neural connection matrix $\boldsymbol{W}$.

• **Forward-self** (Jimenez Rezende & Gerstner, 2014; Kajino, 2021): A typical and intuitive way is to assume the true posterior distribution $p(\boldsymbol{Z}|\boldsymbol{X};\theta)$ can be approximated by a variational distribution $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ which is also a GLM w.r.t. $\boldsymbol{Z}$ where $\boldsymbol{X}$ is fixed (Fig. 1(b)), i.e.,

$$\begin{aligned}f_{t,h} =&\sigma\Bigg(c_h + \sum_{v'=1}^{V} a_{h\leftarrow v'}\cdot\left(\sum_{l=1}^{L} x_{t-l,v'}\,\psi_l\right)\\ &+\sum_{h'=1}^{H} a_{h\leftarrow h'}\cdot\left(\sum_{l=1}^{L} z_{t-l,h'}\,\psi_l\right)\Bigg),\end{aligned} \tag{20}$$

and $\phi = \{\boldsymbol{c}_H,\boldsymbol{A}\}$. Particularly,

$$\boldsymbol{A} = \begin{bmatrix}\boldsymbol{O}_{V\leftarrow V} & \boldsymbol{O}_{V\leftarrow H}\\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{A}_{H\leftarrow H}\end{bmatrix} \in \mathbb{R}^{N\times N}. \tag{21}$$

The top two blocks $\boldsymbol{O}_{V\leftarrow V},\boldsymbol{O}_{V\leftarrow H}$ are all zeros since we don't need to sample the visible spike train $\boldsymbol{X}$. $\boldsymbol{A}_{H\leftarrow V}$ and $\boldsymbol{A}_{H\leftarrow H}$ represent the visible-to-hidden and hidden-to-hidden influences, respectively. To use Eq. 20, we need to sample from $t=1$ to $t=T$ sequentially, since the current sample $z_{t,h}$ relies on the previous samples $\boldsymbol{Z}_{t-L:t-1,1:H}$.

• **Forward**: Due to the low efficiency of the forward-self sampling process, an easier alternative approach is to eliminate the hidden-to-hidden block (i.e., the third term in Eq. 20). Then we get the forward message passing scheme

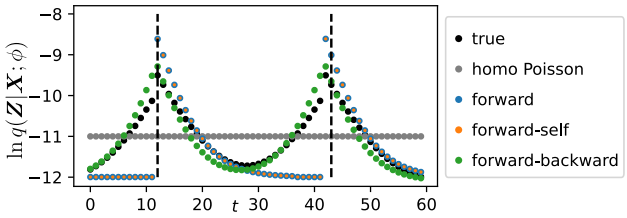

Figure 2. An example of comparison among different variational distributions $q(\boldsymbol{Z}|\boldsymbol{X};\phi)$ with the true posterior $p(\boldsymbol{Z}|\boldsymbol{X};\theta)$ (black dots). There is one visible neuron and one hidden neuron. Two visible spikes from the visible neuron happen at the dashed lines. Each dotted curve represents the approximated log-likelihood of one hidden spike happening at different time bins. Only the forward-backward recapitulates the true posterior distribution. The forward and forward-self miss the uprising trends before the two observed spikes, due to lack of a back-propagated message.

(Fig. 1(c)):

$$f_{t,h} = \sigma\left(c_h + \sum_{v'=1}^{V} a_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right)\right), \quad (22)$$

and $\phi = \{\boldsymbol{c}_H, \boldsymbol{A}\}$. Now,

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{O}_{V\leftarrow V} & \boldsymbol{O}_{V\leftarrow H} \\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{O}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}. \quad (23)$$

The forward variational distribution can be sampled in a parallel style, since $z_{t,h}$ are no longer conditioned on each other. Note that eliminating the hidden-to-hidden block could omit the factor of hidden-to-hidden influences theoretically. But in fact, it is very challenging to learn the actual hidden-to-hidden influences $\boldsymbol{W}_{H\leftarrow H}$ practically given the long sequential sampling procedure. Furthermore, although most of the neurons are actually unrecorded in practice, people usually assume only a few hidden neurons ($V > H$) in POGLM, hoping these hidden neurons can serve as representatives. In fact, it is unrealistic to learn a lot of hidden neurons due to the complicated nature of the POGLM problem itself. If there are a lot of hidden neurons, the problem will be super cumbersome and no method can succeed. Under the condition of $V > H$, therefore, omitting the $\boldsymbol{W}_{H\leftarrow H}$ block might not be very harmful.

• **Forward-backward**: Both of the previous two sampling schemes omit to mimic an important relationship—the hidden-to-visible influences $\boldsymbol{W}_{V\leftarrow H}$ in the generating process $p(\boldsymbol{X}, \boldsymbol{Z};\theta)$. Therefore, we introduce the forward-backward message passing scheme (Fig. 1(d)),

$$\begin{aligned} f_{t,h} =& \sigma\Bigg(c_n + \sum_{v'=1}^{V} a_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) \\ &+ \sum_{v'=1}^{V} a_{v'\leftarrow h} \cdot \left(\sum_{l=1}^{L} x_{t+l,v'}\ \psi_l\right)\Bigg), \end{aligned} \quad (24)$$

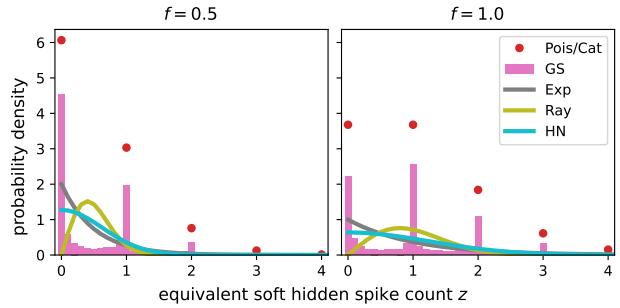

Figure 3. A visualization of different choices of the (soft) hidden spike count distribution, under firing rate $f = 0.5$ and $f = 1.0$. Most of z from GS approximating the original Poisson distribution are close to integer points, but the three continuous distributions (Exp, Ray, and HN) are not.

and $\phi = \{\boldsymbol{c}_H, \boldsymbol{A}\}$. Now,

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{O}_{V\leftarrow V} & \boldsymbol{A}_{V\leftarrow H} \\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{O}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad (25)$$

where the $\boldsymbol{A}_{V\leftarrow H}$ block mimics the hidden-to-visible influences $\boldsymbol{W}_{V\leftarrow H}$ in the generating process $p(\boldsymbol{X}, \boldsymbol{Z};\theta)$, via including the contribution from future visible spikes $\boldsymbol{X}_{t+1:t+L,1:V}$ into sampling the current $z_{t,h}$ (i.e., the third term in Eq. 24).

Fig. 2 visually compares different variational distributions, helping us understand the superiority of the forward-backward sampling, which excels in approximating the true posterior distribution.

3. Experiments

For a comprehensive analysis and comparison, we consider the **method combinations** of different **inference methods** $\times$ different **variational sampling schemes**.

Inference methods. We consider seven inference methods. Each of them is identified by the distribution of the hidden spike given firing rate $\mathbb{P}[z; f]$ (see Tab. 1) and the gradient estimator used in VI.

• **Poisson (Pois)**: This is the original POGLM (Eq. 2, 3, 4, 5, 7). This is named after the Poisson distribution of the hidden spike train $\boldsymbol{Z}$ of the original POGLM. Since this is a discrete distribution, only the score function gradient estimator can be adopted in VI.

• **Categorical (Cat)**: This is the first intermediate model between the original POGLM and the differentiable POGLM, where we don't use Gumbel-Softmax in Eq. 13 to approximate but keep the categorical distribution (Eq. 11). Same as Poisson, only the score function gradient estimator can be adopted in VI.

• **Gumbel-Softmax-score (GS-s)**: This is the differentiable POGLM with GS (Eq. 13) as the soft hidden spike count

Table 1. Different distributions of hidden spike given firing rate $\mathbb{P}[z; f]$, used in the generative model and the variational model. For simplicity, we omit the subscript of z, $\tilde{\boldsymbol{z}} = (\tilde{z}_0, \ldots, \tilde{z}_{M-1})$, and f indexing the hidden neuron and time bin.

distribution	sample	likelihood	can use pathwise
Poisson (Pois)	$z \sim \text{Poisson}(f)$	$\mathbb{P}[z; f] = \frac{f^z \mathrm{e}^{-z}}{z!}$	✗
categorical (Cat)	$z \sim \text{Cat}(\boldsymbol{\pi}(f))$	$\mathbb{P}[z; f] = \boldsymbol{\pi}(f)_z$	✗
Gumbel-Softmax (GS)	$\tilde{\boldsymbol{z}}_{t,h} \sim \text{GS}(\boldsymbol{\pi}(f_{t,h}); \tau)$	Eq. 36	✓
exponential (Exp)	$z \sim \text{Exp}\left(\frac{1}{f}\right)$	$\mathbb{P}[z; f] = \frac{1}{f} \exp(-fz)$	✓
Rayleigh (Ray)	$z \sim \text{Ray}\left(\sqrt{\frac{2}{\pi}} f\right)$	$\mathbb{P}[z; f] = \frac{\pi z}{2f^2} \exp\left(-\frac{\pi z^2}{4f^2}\right)$	✓
Half-normal (HN)	$z \sim \text{HN}\left(\sqrt{\frac{\pi}{2}} f\right)$	$\mathbb{P}[z; f] = \frac{2}{\pi f} \exp\left(-\frac{z^2}{\pi f^2}\right)$	✓

distributions, but we still use the score function gradient estimator (Eq. 10) when updating ϕ although this model is already differentiable.

• **Gumbel-Softmax-pathwise (GS-p)**: This is the differentiable POGLM with GS (Eq. 13) as the soft hidden spike count distributions, where we use the pathwise gradient estimator (Eq. 16) when updating ϕ. This is the inference method we expect to perform better than the previous three.

To experiment with the generalization from GS to other single-parameter continuous distributions, we try the following three and use the pathwise gradient estimator.

• **Exponential (Exp)**.

• **Rayleigh (Ray)**.

• **Half-normal (HN)**.

The aim of including the two intermediate models (Cat and GS-s) is to change the model step-by-step from the original POGLM with Poisson as the hidden spike count distribution and the score function gradient estimator to the differentiable POGLM with GS as the hidden spike count distribution and the pathwise gradient estimator. Through this controlled variable design, we can have a better understanding of the final differentiable POGLM with continuous soft hidden spike count distributions. Since we have no prior knowledge about which single-parameter distribution is better, we try the three common ones: Exp, Ray, and HN. A visualization of these distributions is shown in Fig. 3.

Variational sampling schemes. We consider three sampling schemes.

• **Forward (F)**: The sampling scheme illustrated in Eq. 20 and Fig. 1(c).

• **Forward-self (FS)**: The sampling scheme illustrated in Eq. 22 and Fig. 1(b).

• **Forward-backward (FB)**: The sampling scheme illustrated in Eq. 24 and Fig. 1(d).

The homogeneous and inhomogeneous Poisson will be ignored in the following experiments due to their over-simplicity or incompatibility with the test set, as illustrated in Sec. 2.3.

The original approach to solving POGLM can be viewed as the combination of Poisson $\times$ FS (Pillow & Latham, 2007; Jimenez Rezende & Gerstner, 2014; Linderman et al., 2017). Our newly proposed differentiable POGLM with GS and other continuous distributions combined with the FB message passing scheme should be the optimal combinations we expect. For clarity, Appendix. A.2 provides a comprehensive summary of these method combinations.

Evaluation. Although we have different inference methods, when evaluating log-likelihood (LL) on the test dataset, all inference methods are set back to the original POGLM form, where Poisson log-likelihood (Eq. 3 and Eq. 5) is adopted for a fair comparison. The LL metric can be used on both synthetic and real-world datasets. Furthermore, since we are also very concerned about the parameter recovery, as we illustrated in Sec. 1 that naively apply VI to solve the POGLM directly cannot obtain an ideal parameter recovery result due to the difficulty of the POGLM problem itself, we will compare the average absolute error of the estimated parameter on the synthetic dataset.

3.1. Synthetic Dataset

The synthetic dataset aims to compare different method combinations comprehensively. With the known true parameters, we can validate that better performance corresponds to smaller parameter errors. We can also check the benefit of applying the pathwise gradient estimator.

Dataset. We randomly create 10 sets of parameters for generating the synthetic datasets, wherein each set $w_{n \leftarrow n'} \overset{\text{i.i.d.}}{\sim} \text{Unif}(-2, 2)$ and $b_n \overset{\text{i.i.d}}{\sim} \text{Unif}(-0.5, 0.5)$. Each set corresponds to a trial, resulting in a total of 10 trials. The model consists of 5 neurons, with the first 3 being visible and the remaining 2 being hidden. For each trial, we generate 40 spike trains for training and 20 spike trains for testing. Each spike train has 100 time bins.

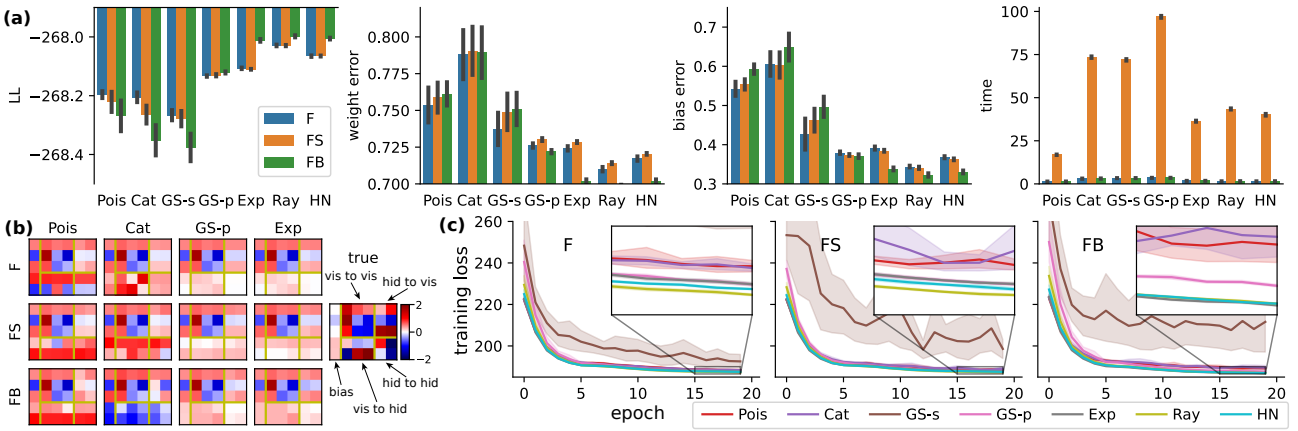

Figure 4. **(a)**: The test log-likelihood (LL), the weight error, the bias error, and the running time of different method combinations. **(b)**: An example of the learned weight matrix and bias vector compared with the true of selected method combinations. Visualization of all method combinations is in Fig. 8 in Appendix. A.3. **(c)**: The learning curves of different method combinations.

Experimental setup. The initial values for the linear weights and biases of the model used for learning are also randomly initialized as above. We utilize the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 0.05. The optimization process runs for 20 epochs, and within each epoch, optimization is performed using 4 batches, each of size 10. The entire process is repeated 10 times with different random seeds for each trial, and the performance and the error bar are reported based on these repetitions.

Results. In Fig. 4(a), we can see that for each sampling scheme, the test LL drops when we use categorical and then GS to approximate the discrete Poisson distribution. However, when we change the gradient estimator from the score function to the pathwise, the LL increases significantly for GS and exceeds the original Poisson. The LL keeps increasing when we change the distribution from GS to Exp, Ray, or HN. This implies that although the hidden spikes are generated from the Poisson distribution, the posterior distribution might not be Poisson, but a discrete distribution that is closer to a continuous distribution like the exponential in shape. From the view of different sampling schemes, FB becomes better than F and FS when a good inference method is used, e.g., GS-p or Exp. Without a good inference method, the forward sampling scheme (F) performs better due to its simplicity. In summary, a differentiable POGLM using pathwise gradient estimator $\times$ the FB sampling scheme promises a better performance.

The weight error and bias error in Fig. 4(a) quantitatively validate that a better LL corresponds to a smaller parameter error. Consistent with the LL bar plot, the weight error and bias error bar plots indicate that Exp, Ray, and HN are better than GS-p and better than the remaining. With a continuous distribution and the pathwise gradient estimator as the inference method, FB is the optimal sampling scheme.

Fig. 4(b) visualizes the recovered weight matrix and bias vector of some selected method combinations versus the true one used for generating the dataset. The differences between these recovered weights and biases are mainly from the hidden related weight blocks and hidden biases.

In addition to the performance, we also compare the running time of different method combinations in Fig. 4(a). Due to the sequential dependency of the FS, the running time is significantly longer than F and FB. This implies the benefit of excluding the complicated "self" message passing component in the variational model. Besides, converting a discrete Poisson distribution to its hard (Cat) and soft (GS) approximating distribution takes some extra time.

Fig. 4(c) plots the loss curves of different inference methods using different sampling schemes. The loss curves of those differentiable inference methods (GS-p, Exp, Ray, and HN) are smoother than the others. The small error bars of these differentiable inference methods imply the smaller variance of the pathwise gradient estimator when updating the variational model parameter ϕ than that of the score function gradient estimator.

3.2. Retinal Ganglion Cell (RGC) Dataset

This dataset aims to understand the performances of different method combinations on a real-world dataset, and the interpretability of the estimated model parameters.

Dataset. Next, we apply various method combinations to analyze a real neural spike train recorded from 27 retinal ganglion neurons while a mouse is engaged in a visual task for approximately 20 minutes (Pillow & Scott, 2012). Specifically, neurons 1–16 are OFF cells, while neurons 17–27 are ON cells.

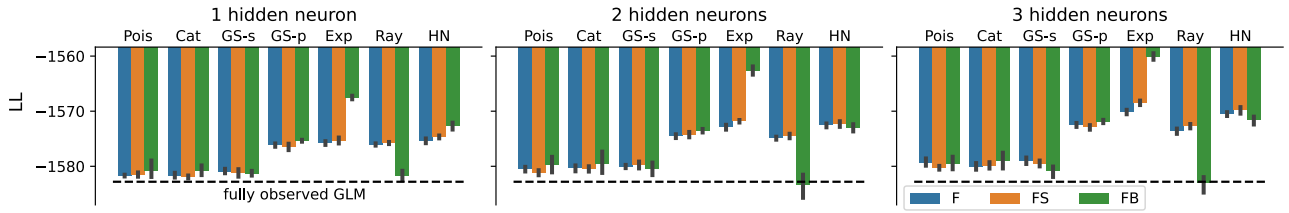

Figure 5. The test log-likelihood (LL) of different method combinations under $H \in \{1, 2, 3\}$ hidden neurons. The dashed black line represents the test LL of the fully observed GLM as the baseline.

Experimental setup. We partition the spike train into training and test sets, using the first 2/3 segment for training and the remaining 1/3 segment for testing. The raw spike train is binned into spike count via 50 ms time bins. To facilitate the application of the stochastic gradient descent algorithm, we divide the entire sequence into 14400 pieces in total, each consisting of 100 time bins. Since we do not have prior knowledge of how many hidden neurons should be assumed, we initially train a fully observed GLM as a baseline. Subsequently, we assume the presence of $H \in \{1, 2, 3\}$ hidden representative neurons and train the model using different method combinations. The optimization is performed using the Adam optimizer with a learning rate of 0.02. Each training procedure undergoes 20 epochs, employing a batch size of 32. We repeat the training and test process 10 times with different random seeds and report the performance and error bar.

Results. Similar to the synthetic dataset, we plot the test LL of different method combinations under 1, 2, and 3 hidden neurons in Fig. 5. Besides, a fully observed GLM is also included in Fig. 5 as the baseline. First of all, with the assumption of containing hidden neurons, the performances of all method combinations become better than the baseline, except for the Ray $\times$ FB. Second, differentiable inference methods are better than non-differentiable inference methods in general. Particularly, Exp $\times$ FB is significantly better than all other method combinations. Third, for most of the method combinations, increasing the number of hidden neurons improves the LL, especially for GS-p and Exp.

In addition to knowing that the differentiable inference methods $\times$ FB are better than others, we are also curious about the interpretation of the learned model parameters. Take 1 hidden neuron as an example, Fig. 6 shows that, the learned one hidden representative by Exp $\times$ FB serves as a negative feedback regulating unit. Specifically, the weights from this hidden representative to all OFF cells are negative, and to all ON cells are positive; the weights from nearly all OFF cells to this hidden representative are positive, and from all ON cells are negative. That is, the signs of the weights from this hidden representative to visible neurons provide clear indications of the type of visible neurons. Similar but weaker results can also be obtained by GS-p but not by Poisson.

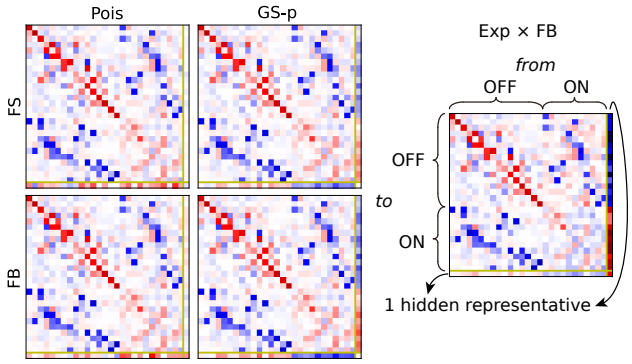

Figure 6. The learned weight matrix of selected method combinations. Visualization of all method combinations is in Fig. 9 in Appendix. A.3.

Fig. 9 in Appendix A.3 also supports the interpretability for more hidden neurons. For example, with 3 hidden representative neurons learned in the RGC dataset, the hidden to visible weights learned from Exp $\times$ FB form a particular non-random pattern. This pattern represents a more complex and diverse interaction between hidden representative neurons and visible neurons.

3.3. The PVC-5 Dataset

This dataset aims to investigate the performance variation of different method combinations w.r.t. the numbers of hidden neurons.

Dataset. Finally, we apply different method combinations to a dataset obtained from the primary visual cortex (PVC-5) (Chu et al., 2014)[1]. This dataset consists of recordings from the primary visual cortex (V1) of a macaque monkey over a 15-minute duration without the presentation of any stimuli. Three adjacent neurons were recorded through contact with an electrode.

Experimental setup. Similarly to the RGC dataset, we train the model on the initial 7.5 minutes of data and evaluate

[1] https://crcns.org/data-sets/pvc/pvc-5

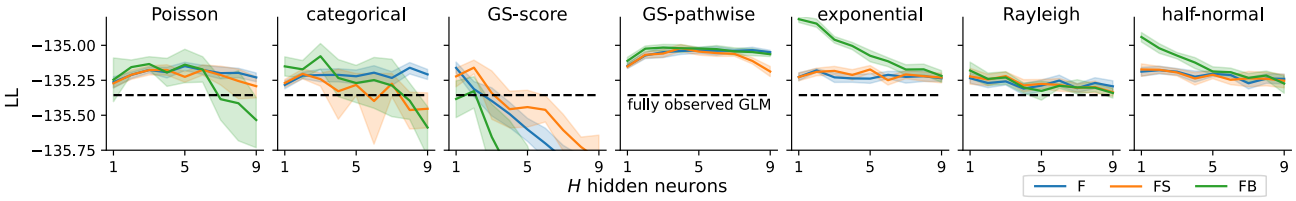

Figure 7. The curves of the test log-likelihood (LL) v.s. the number of hidden neurons H, for different method combinations.

the test log-likelihood on the subsequent 7.5 minutes. The raw spike train is binned into spike count via 20 ms time bins. The training set is divided into 225 pieces equally and the batch size for training is 25. Since there are only three visible neurons, we can try more numbers of hidden neurons $H \in \{1, \ldots, 9\}$ and understand the change of performance w.r.t. the number of hidden neurons, especially when $H \gg V$. The optimization is performed using the Adam optimizer for 20 epochs with a learning rate of 0.1. We repeat the training and test process 10 times with different random seeds and report the performance and error bar.

Results. Fig. 7 shows the performance of different method combinations w.r.t. number of hidden neurons. No matter what method combination we choose, the optimal number of hidden neurons is no more than 3. This means assuming more hidden neurons might not guarantee an improvement of the performance but is likely to introduce redundancy and result in a dropped performance. With more hidden neurons, the performance of those non-differentiable inference methods becomes even worse than a fully observed GLM.

Among all method combinations, Exp $\times$ FB with less than 3 hidden neurons seems to be the best. GS-p (with all three sampling schemes) is more robust to different numbers of hidden neurons than other inference methods. Furthermore, the variance of the test LL of those differentiable inference methods is much smaller than that of those non-differentiable inference methods, due to the benefit from using the pathwise gradient estimator.

4. Related Works

Some previous works consider the POGLM in its point process form, i.e., a generalized multivariate partially observable Hawkes process, in which spike trains are not binned into spike count but keep their raw form of spiking event timestamps. For example, Zhou & Sun (2021); Shelton et al. (2018); Mei et al. (2019) treated the problem as missing data (missing all the spiking data from hidden neurons); Kajino (2021) proposed a differentiable point process model to enable the use of the pathwise gradient estimator.

Through the point process, however, the data structure that stores the spike timestamps is usually not ideal (Xu, 2018). The detailed reasons are illustrated in Appendix. A.4. In this paper, we only focus on POGLM, i.e., pre-binned spike count trains. More detailed discussions regarding the relationship between the (discrete) GLM and the (continuous) generalized Hawkes process can also be found in Appendix. A.4.

5. Discussion

In this paper, we propose a differentiable version of the partially observable generalized linear model (POGLM), in which the pathwise gradient estimator becomes applicable when doing variational inference (VI). Due to the inexpressivity and low sampling efficiency of the existing forward-self sampling scheme, we propose the new forward-backward message passing sampling scheme, introduce the message passing from hidden neurons to visible neurons, and result in a better variational model for VI. Comprehensive comparisons between different method combinations on one synthetic and two real-world datasets show that a differentiable inference method with the forward-backward sampling scheme could produce a higher likelihood on the test set and better parameter recovery.

Note that the relaxation from the Gumbel-Softmax distribution to general continuous distributions loses the meaning of $\boldsymbol{Z}$ as representing spike counts, but can produce better performance. It is interesting but challenging to investigate whether a general continuous distribution is closer to the true posterior distribution than the discrete Poisson distribution. This limitation is a big topic that could be a future direction.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Bengio, Y., Léonard, N., and Courville, A. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*,

2013.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Chu, C., Chien, P. F., and Hung, C. Multi-electrode recordings of ongoing activity and responses to parametric stimuli in macaque v1. *CRCNS.org*, 10:K0J1012K, 2014.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Jimenez Rezende, D. and Gerstner, W. Stochastic variational learning in recurrent spiking networks. *Frontiers in computational neuroscience*, 8:38, 2014.

Kajino, H. A differentiable point process with its application to spiking neural networks. In *International Conference on Machine Learning*, pp. 5226–5235. PMLR, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Li, C., Kim, S. H., Rodgers, C., Choi, H., and Wu, A. One-hot generalized linear model for switching brain state discovery. In *International Conference on Learning Representations*, 2024a.

Li, C., Wang, Y., Li, W., and Wu, A. Forward χ^2 divergence based variational importance sampling. In *International Conference on Learning Representations*, 2024b.

Linderman, S., Adams, R. P., and Pillow, J. W. Bayesian latent structure discovery from multi-neuron recordings. *Advances in neural information processing systems*, 29, 2016.

Linderman, S. W., Wang, Y., and Blei, D. M. Bayesian inference for latent hawkes processes. *Advances in Neural Information Processing Systems*, 2017.

Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Mei, H., Qin, G., and Eisner, J. Imputing missing events in continuous-time event streams. In *International Conference on Machine Learning*, pp. 4475–4485. PMLR, 2019.

Paisley, J., Blei, D., and Jordan, M. Variational bayesian inference with stochastic search. *arXiv preprint arXiv:1206.6430*, 2012.

Pillow, J. and Latham, P. Neural characterization in partially observed populations of spiking neurons. *Advances in Neural Information Processing Systems*, 20, 2007.

Pillow, J. and Scott, J. Fully bayesian inference for neural models with negative-binomial spiking. *Advances in neural information processing systems*, 25, 2012.

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E., and Simoncelli, E. P. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

Roudi, Y., Dunn, B., and Hertz, J. Multi-neuronal activity and functional connectivity in cell assemblies. *Current opinion in neurobiology*, 32:38–44, 2015.

Schulman, J., Heess, N., Weber, T., and Abbeel, P. Gradient estimation using stochastic computation graphs. *Advances in neural information processing systems*, 28, 2015.

Shelton, C., Qin, Z., and Shetty, C. Hawkes process inference with missing data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Xu, H. Poppy: a point process toolbox based on pytorch. *arXiv preprint arXiv:1810.10122*, 2018.

Zhou, Z. and Sun, M. Multivariate hawkes processes for incomplete biased data. In *2021 IEEE International Conference on Big Data (Big Data)*, pp. 968–977. IEEE, 2021.

A. Appendix

A.1. Gradient estimators of ELBO

Here we provide detailed derivations for the gradient estimators w.r.t. θ and ϕ of $\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)$. The derivative w.r.t. θ is:

$$\begin{aligned}\frac{\partial\,\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\theta} =& \frac{1}{|\mathbb{N}^{T\times H}|}\sum_{\boldsymbol{Z}\in\mathbb{N}^{T\times H}} q(\boldsymbol{Z}|\boldsymbol{X};\phi)\frac{\partial}{\partial\theta}\left[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta)-\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi)\right]\\ \approx& \frac{1}{K}\sum_{k=1}^{K}\frac{\partial}{\partial\theta}\left[\ln p\left(\boldsymbol{X},\boldsymbol{Z}^{(k)};\theta\right)-\ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi\right)\right]\\ =& \frac{\partial}{\partial\theta}\widehat{\mathrm{ELBO}}(\boldsymbol{X};\theta,\phi).\end{aligned} \tag{26}$$

The score function gradient estimator (i.e., Eq. 10) of the derivative of ELBO w.r.t. ϕ at a particular value ϕ_0 is:

$$\begin{aligned}\frac{\partial\,\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\phi} =& \frac{1}{|\mathbb{N}^{T\times H}|}\sum_{\boldsymbol{Z}\in\mathbb{N}^{T\times H}}\frac{\partial}{\partial\phi}q(\boldsymbol{Z}|\boldsymbol{X};\phi)\left[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta)-\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi_0)\right]\\ &+q(\boldsymbol{Z}|\boldsymbol{X};\phi_0)\frac{\partial}{\partial\phi}\left[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta)-\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi)\right]\\ =& \frac{1}{|\mathbb{N}^{T\times H}|}\sum_{\boldsymbol{Z}\in\mathbb{N}^{T\times H}}\left[\ln p(\boldsymbol{X},\boldsymbol{Z};\theta)-\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi_0)\right]q(\boldsymbol{Z}|\boldsymbol{X};\phi)\frac{\partial}{\partial\phi}\ln q(\boldsymbol{Z}|\boldsymbol{X};\phi)\\ &-\frac{1}{|\mathbb{N}^{T\times H}|}\sum_{\boldsymbol{Z}\in\mathbb{N}^{T\times H}}\frac{\partial}{\partial\phi}q(\boldsymbol{Z}|\boldsymbol{X};\phi)\\ \approx& \frac{1}{K}\sum_{k=1}^{K}\left[\ln p\left(\boldsymbol{X},\boldsymbol{Z}^{(k)};\theta\right)-\ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi_0\right)\right]\frac{\partial}{\partial\phi}\ln q\left(\boldsymbol{Z}^{(k)}\middle|\boldsymbol{X};\phi\right)-0.\end{aligned} \tag{27}$$

The pathwise gradient estimator (i.e., Eq. 16) of the derivative of ELBO w.r.t. ϕ is:

$$\begin{aligned}\frac{\partial\,\mathrm{ELBO}(\boldsymbol{X};\theta,\phi)}{\partial\phi} =& \frac{\partial}{\partial\phi}\int_{\tilde{\boldsymbol{Z}}} q\left(\tilde{\boldsymbol{Z}}\middle|\boldsymbol{X};\phi\right)\left[\ln p\left(\boldsymbol{X},\tilde{\boldsymbol{Z}};\theta\right)-\ln q\left(\tilde{\boldsymbol{Z}}\middle|\boldsymbol{X};\phi_0\right)\right]\,\mathrm{d}\tilde{\boldsymbol{Z}}\\ =& \frac{\partial}{\partial\phi}\int_{\boldsymbol{G}}\mathrm{Gumbel}(\boldsymbol{G};0,1)\left[\ln p\left(\boldsymbol{X},r(\boldsymbol{G}|\boldsymbol{X};\phi);\theta\right)-\ln q(r(\boldsymbol{G}|\boldsymbol{X};\phi)|\boldsymbol{X};\phi)\right]\,\mathrm{d}\boldsymbol{G}\\ \approx& \frac{\partial}{\partial\phi}\sum_{k=1}^{K}\left[\ln p\left(\boldsymbol{X},r\left(\boldsymbol{G}^{(k)}\middle|\boldsymbol{X};\phi\right);\theta\right)-\ln q\left(r\left(\boldsymbol{G}^{(k)}\middle|\boldsymbol{X};\phi\right)\middle|\boldsymbol{X};\phi\right)\right]\\ =& \frac{\partial}{\partial\phi}\widehat{\mathrm{ELBO}}(\boldsymbol{X};\theta,\phi).\end{aligned} \tag{28}$$

A.2. Method combinations

We summarize different method combinations in this subsection.

A.2.1. Generative procedure

For the firing rate $f_{t,v}$ of a visible neuron v and the firing rate $f_{t,h}$ of a hidden neuron h at time t:

$$\begin{cases} f_{t,v} = \sigma\left(b_v + \sum_{v'=1}^{V} w_{v\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{h'=1}^{H} w_{v\leftarrow h'} \cdot \left(\sum_{l=1}^{L} z_{t-1,h'}\ \psi_l\right)\right), \\ f_{t,h} = \sigma\left(b_h + \sum_{v'=1}^{V} w_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{h'=1}^{H} w_{h\leftarrow h'} \cdot \left(\sum_{l=1}^{L} z_{t-1,h'}\ \psi_l\right)\right). \end{cases} \tag{29}$$

The parameter set is $\theta = \{\boldsymbol{b}, \boldsymbol{W}\}$, where $\boldsymbol{b} = \begin{bmatrix} \boldsymbol{b}_V \\ \boldsymbol{b}_H \end{bmatrix} \in \mathbb{R}^N$ and $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{V\leftarrow V} & \boldsymbol{W}_{V\leftarrow H} \\ \boldsymbol{W}_{H\leftarrow V} & \boldsymbol{W}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}$. For visible neurons, $x_{t,n} \sim \text{Poisson}(f_{t,n})$.

A.2.2. Variational sampling schemes

• **Homogeneous Poisson**

$$f_{t,h} = \sigma(c_h). \tag{30}$$

The variational parameter set is $\phi = \{\boldsymbol{c}_H\}$.

• **Inhomogeneous Poisson**

$$f_{t,h} = \sigma(c_{t,h}). \tag{31}$$

The variational parameter set is $\phi = \{\boldsymbol{C}_{T\times H}\}$.

• **Forward (F)**

$$f_{t,h} = \sigma\left(c_h + \sum_{v'=1}^{V} a_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right)\right). \tag{32}$$

The variational parameter set is $\phi = \{\boldsymbol{c}_H, \boldsymbol{A}\}$, where $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{O}_{V\leftarrow V} & \boldsymbol{O}_{V\leftarrow H} \\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{O}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}$.

• **Forward-self (FS)**

$$f_{t,h} = \sigma\left(c_h + \sum_{v'=1}^{V} a_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{h'=1}^{H} a_{h\leftarrow h'} \cdot \left(\sum_{l=1}^{L} z_{t-l,h'}\ \psi_l\right)\right). \tag{33}$$

The variational parameter set is $\phi = \{\boldsymbol{c}_H, \boldsymbol{A}\}$, where $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{O}_{V\leftarrow V} & \boldsymbol{O}_{V\leftarrow H} \\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{A}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}$.

• **Forward-backward (FB)**

$$f_{t,h} = \sigma\left(c_n + \sum_{v'=1}^{V} a_{h\leftarrow v'} \cdot \left(\sum_{l=1}^{L} x_{t-l,v'}\ \psi_l\right) + \sum_{v'=1}^{V} a_{v'\leftarrow h} \cdot \left(\sum_{l=1}^{L} x_{t+l,v'}\ \psi_l\right)\right) \tag{34}$$

The variational parameter set is $\phi = \{\boldsymbol{c}_H, \boldsymbol{A}\}$, where $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{O}_{V\leftarrow V} & \boldsymbol{A}_{V\leftarrow H} \\ \boldsymbol{A}_{H\leftarrow V} & \boldsymbol{O}_{H\leftarrow H} \end{bmatrix} \in \mathbb{R}^{N\times N}$.

A.2.3. HIDDEN SPIKE COUNT DISTRIBUTIONS

Table 2. Different distributions of hidden spike given firing rate $\mathbb{P}[z; f]$, used in the generative model and the variational model. For simplicity, we omit the subscript of z, $\tilde{\boldsymbol{z}} = (\tilde{z}_0, \ldots, \tilde{z}_{M-1})$, and f indexing the hidden neuron and time bin.

distribution	sample	likelihood	can use pathwise
Poisson (Pois)	$z \sim \mathrm{Poisson}(f)$	$\mathbb{P}[z; f] = \frac{f^z \mathrm{e}^{-z}}{z!}$	✗
categorical (Cat)	$z \sim \mathrm{Cat}(\boldsymbol{\pi}(f))$	$\mathbb{P}[z; f] = \boldsymbol{\pi}(f)_z$	✗
Gumbel-Softmax (GS)	$\tilde{\boldsymbol{z}}_{t,h} \sim \mathrm{GS}(\boldsymbol{\pi}(f_{t,h}); \tau)$	Eq. 36 bellow	✓
exponential (Exp)	$z \sim \mathrm{Exp}\left(\frac{1}{f}\right)$	$\mathbb{P}[z; f] = \frac{1}{f} \exp\left(-fz\right)$	✓
Rayleigh (Ray)	$z \sim \mathrm{Ray}\left(\sqrt{\frac{2}{\pi}} f\right)$	$\mathbb{P}[z; f] = \frac{\pi z}{2f^2} \exp\left(-\frac{\pi z^2}{4f^2}\right)$	✓
Half-normal (HN)	$z \sim \mathrm{HN}\left(\sqrt{\frac{\pi}{2}} f\right)$	$\mathbb{P}[z; f] = \frac{2}{\pi f} \exp\left(-\frac{z^2}{\pi f^2}\right)$	✓

In the above table, we used the function $\boldsymbol{\pi}$ to truncate a Poisson distribution to a categorical distribution,

$$\boldsymbol{\pi}(f) = \left(1 - \sum_{m=1}^{M-1} \frac{f^m \mathrm{e}^f}{m!}, \frac{f^1 \mathrm{e}^f}{1!}, \ldots, \frac{f^{M-1} \mathrm{e}^f}{(M-1)!}\right). \tag{35}$$

The GS likelihood is

$$\mathbb{P}[\tilde{\boldsymbol{z}}; f] = \Gamma(M) \tau^{M-1} \left(\sum_{m=0}^{m-1} \frac{\boldsymbol{\pi}(f)_m}{\tilde{z}_m^{\tau}}\right) \prod_{m=0}^{M-1} \frac{\boldsymbol{\pi}(f)_m}{\tilde{z}_m^{\tau+1}} \tag{36}$$

A.3. Supplementary Figures

A.3.1. Synthetic dataset

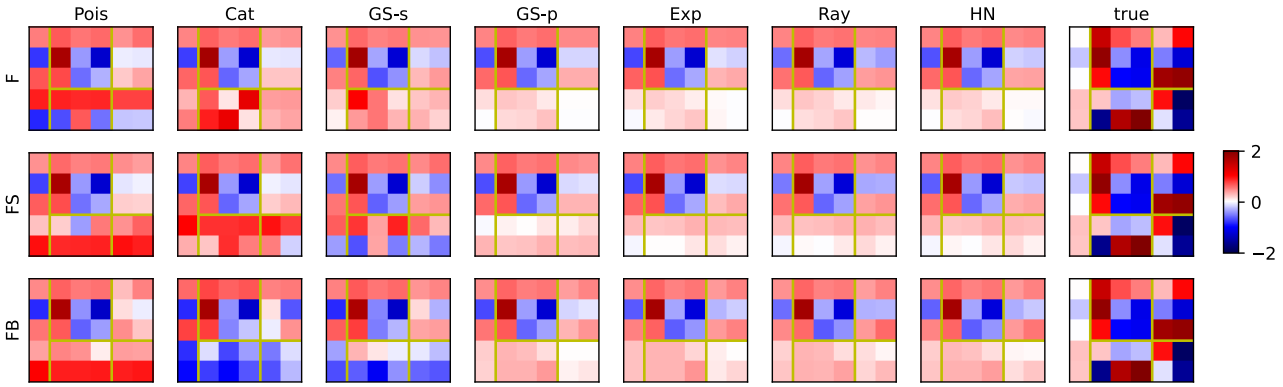

Figure 8. The learned weight matrix and bias vector compared with the true of all method combinations on one trial of the synthetic dataset.

A.3.2. RGC DATASET

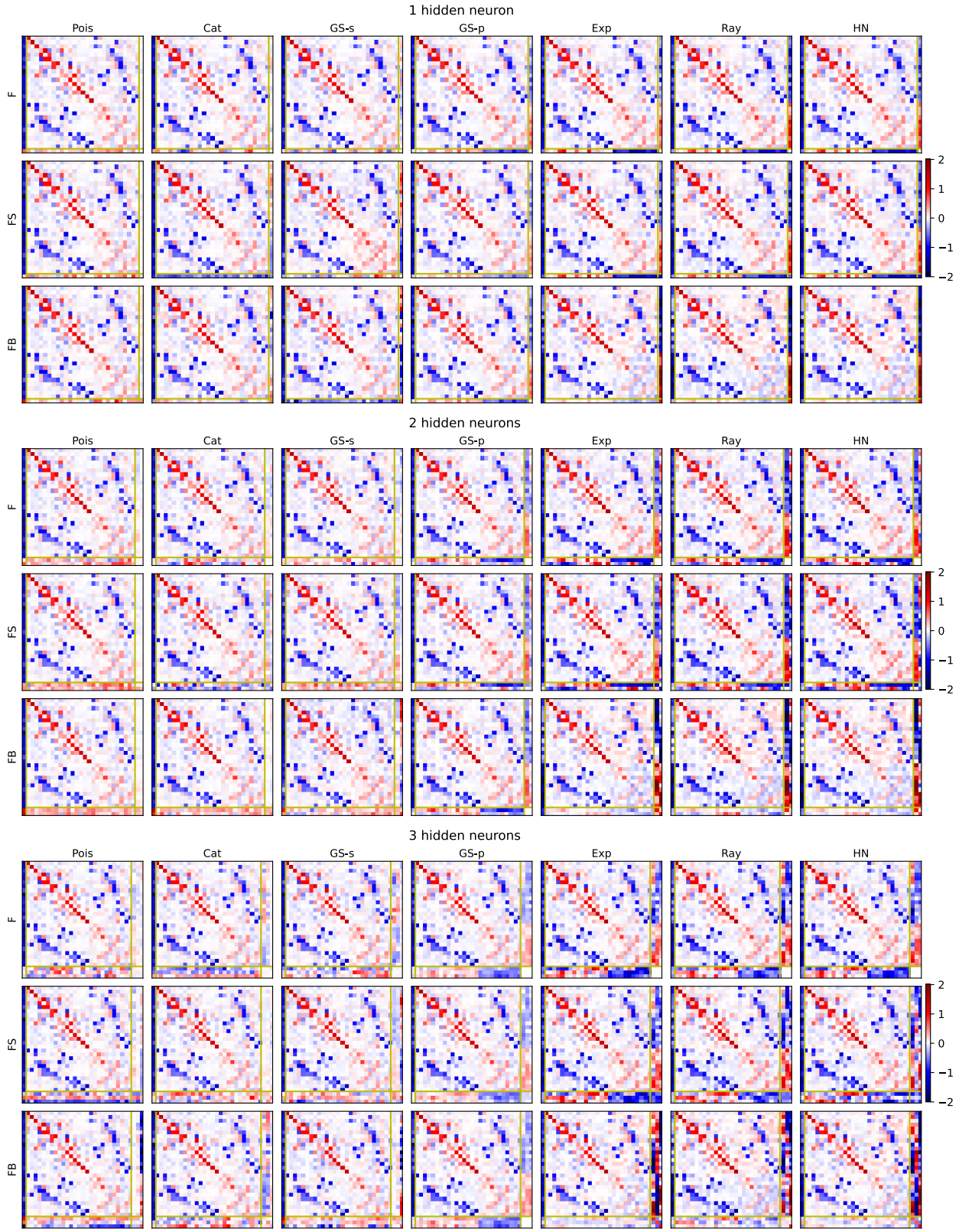

Figure 9. The learned weight matrix and bias vector of all method combinations under different numbers of hidden neurons on the RGC dataset.

A.4. Point Process

A.4.1. The generalized Hawkes process as the point process version of a GLM

A generalized multivariate Hawkes process (GMHP) is a temporal point process (right-continuous) governed by the conditional intensity function

$$\lambda_n^*(t) = \sigma\left(b_n + \sum_{t_n<t} w_{n\leftarrow n_i} \cdot \psi(t-t_i)\right), \tag{37}$$

where n indexes for N neurons. (t_i, n_i) is the arrival time and neuron index of the i-th event in the history (spike) sequences ordered by arrival time t_i, i.e., $t_1 < t_2 < \cdots < t$. $\lambda_n^*(t)$ is conditioned on the spike sequences before time t: $\{(t_n, d_n)_{t_n<t}\}$. $b_n \in \mathbb{R}$ is the background intensity of the n-th neuron, $w_{n\leftarrow n'} \in \mathbb{R}$ is the connection weight from the n'-th neuron to the n-th neuron, and $\psi(\cdot)$ is the basis function, which usually integrates to 1. σ is a nonlinear function. Also, note that $\lambda_n^*(t)$ is a **left continuous** function. For preserving the causal relationship, we also need $\psi(t) = 0,\ \forall t \leqslant 0$. Let $\theta = \{\boldsymbol{b}, \boldsymbol{W}\} = \left\{[b_1, \ldots, b_N]^{\mathrm{T}}, [w_{n\leftarrow n'}]_{N\times N})\right\}$ denote the parameter set we estimate. The data likelihood (probability density function) of a spike train in the format of a continuous timestamps sequence $\mathcal{X} = \{(t_i, n_i)\}_{i=1}^I$ within a specific observation period $[0, T]$ is

$$\mathbb{P}(\mathcal{X};\theta) = \prod_{i=1}^{I} \lambda_{n_i}^*(t_i) \cdot \exp\left[-\sum_{n=1}^{N}\int_0^T \lambda_n^*(t)\,\mathrm{d}t\right]. \tag{38}$$

The relationships between conditional intensity and arrival interval. For any temporal point process (not discretized), the number of events happening between t and $t+\Delta t$ follows the Poisson distribution

$$X \sim \mathcal{P}\left(\int_t^{t+\Delta t} \lambda^*(s)\,\mathrm{d}s\right) = \mathcal{P}(\Lambda(t+\tau) - \Lambda(t)), \tag{39}$$

where $\Lambda(t) \coloneqq \int_0^t \lambda^*(s)\,\mathrm{d}s$ is the compensator. If $\Delta t \to 0$,

$$\int_t^{t+\Delta t} \lambda^*(s)\,\mathrm{d}s \approx \lambda^*(t)\Delta t. \tag{40}$$

The relationship between the intensity function $\lambda^*(t)$ and the next (starts from current time t) time interval τ distribution (PDF) is

$$\begin{aligned} f(\tau) = & \underbrace{\lambda^*(t+\tau)}_{\text{an event happens at } t+\tau\text{, density}} \underbrace{\frac{\left(\int_t^{t+\tau}\lambda^*(s)\,\mathrm{d}s\right)^0 \mathrm{e}^{-\int_t^{t+\tau}\lambda^*(s)\,\mathrm{d}s}}{0!}}_{\text{no event happens in the interval } (t,t+\tau)\text{, probability}} \\ = & \lambda^*(t+\tau)\mathrm{e}^{-\int_t^{t+\tau}\lambda^*(s)\,\mathrm{d}s}. \end{aligned} \tag{41}$$

The cumulative distribution function (CDF) is

$$F(\tau) = 1 - \mathrm{e}^{-\int_t^{t+\tau}\lambda^*(s)\,\mathrm{d}s}. \tag{42}$$

Therefore, modeling the intensity function is equivalent to modeling the arrival interval.

Sampling/Simulation. We introduce two equivalent sampling algorithms: Ogata's thinning and the first-come-first-serve (FCFS). The probability density of the next event happening at time $t+\tau$ of neuron n is

$$f(\tau, n) = \prod_{n'=1}^{N} \mathrm{e}^{-\int_t^{t+\tau}\lambda_{n'}^*(s)\,\mathrm{d}s}\ \lambda_n^*(t+\tau), \tag{43}$$

which can be viewed as FCFS. Using Ogata's thinning method,

$$f(\tau, n) = \mathrm{e}^{-\int_t^{t+\tau}\sum_{n'=1}^{N}\lambda_{n'}^*(s)\,\mathrm{d}s} \sum_{n'=1}^{N} \lambda_{n'}^*(t+\tau)\ \frac{\lambda_n^*(t+\tau)}{\sum_{n'=1}^{N}\lambda_{n'}^*(t+\tau)}. \tag{44}$$

A.4.2. Discretization of the continuous point process

In general, the integration in the complete-data likelihood does not have an analytical solution. A general solution is Monte Carlo integration, despite being suboptimal. A better choice is the quadrature rule, such as the Simpson rule. However, no matter what numerical techniques we use, we lose the continuous property of the process itself. Besides, the data structure that stores the spike train timestamps is usually not ideal. First, computing the log-likelihood of a point process requires sequential searching on lists of timestamps, which is more complicated than a direct matrix multiplication used in binned spike train data. Second, sampling hidden spike timestamps requires sorting, which is extremely time-consuming. Furthermore, the integration term in the log-likelihood function usually has no closed-form expression and hence still needs time discretization. Therefore, it is more convenient to discretize the point process data into binned spike count at the very beginning rather than deal with sequences of continuous timestamps.

Now, we introduce the discretization procedure. Denote $\boldsymbol{X} \in \mathbb{N}^{S\times N}$ as the discretized spike train, where $S = \frac{T}{\Delta t}$ is the total number of time bins. Then $x_{s,n}$ represents the number of spikes of neuron n in the time interval $((s-1)\Delta t, s\Delta t)$. Now, a GMHP is discretized as a generalized linear model (GLM): $x_{s,n} \sim \mathcal{P}(\lambda^*_{s,n}\Delta t)$, and

$$\lambda^*_{s,n} = b_n + \sum_{x_{s',n'}>0, s'<s} x_{s',n'} w_{n\leftarrow n'} \psi((s-s')\Delta t) = b_n + \sum_{n'=1}^{N} w_{n\leftarrow n'} \sum_{l=1}^{L} x_{s-l,n'} \psi_l, \tag{45}$$

where $\lambda^*_{s,d}$ is still parameterized by θ, and $\boldsymbol{\psi}^{\mathrm{T}} = [\psi(\Delta t), \ldots, \psi(L\Delta t)]$. The complete-data likelihood is

$$\mathbb{P}(\boldsymbol{X};\theta) = \prod_{n=1}^{N}\prod_{s=1}^{S} \frac{(\lambda^*_{s,n}\Delta t)^{x_{s,n}} \mathrm{e}^{-\lambda^*_{s,n}\Delta t}}{x_{s,n}!}. \tag{46}$$

Now, we show that this likelihood converges to the form in the continuous case when the width of the time bin $\Delta t \to 0$. When taking the limit $\Delta t \to 0$, $x_{s,n}$ can either be 0 or 1. Therefore,

$$\begin{aligned}\lim_{\Delta t\to 0} \mathbb{P}(\boldsymbol{X};\theta) =& \prod_{x_{s,n}=1} \lambda^*_{s,n}\Delta t\, \mathrm{e}^{-\lambda^*_{s,n}\Delta t} \prod_{x_{s,n}=0} \mathrm{e}^{-\lambda^*_{s,n}\Delta t} \\ =&(\Delta t)^I \prod_{i=1}^{I} \lambda^*_{n_i}(t_i;\theta)\ \exp\left[-\sum_{n=1}^{N}\int_0^T \lambda^*_n(t;\theta)\,\mathrm{d}t\right].\end{aligned} \tag{47}$$

Since $(\Delta t)^I$ is a constant, we divide the above equation by $(\Delta t)^I$ and just get the probability density, which is identical to Eq. 38 in the continuous case. Therefore, solving the discretized problem is equivalent to solving the original continuous problem, as long as Δt is small enough. And the solution converges to the continuous solution as $\Delta t \to 0$. In fact, the procedure shown in Eq. 47 is applicable to any point process and its discretized version.