

Statistical Learning of Neuronal Functional Connectivity

Chunming ZHANG

Department of Statistics
University of Wisconsin–Madison
Madison, WI 53706
(cmzhang@stat.wisc.edu)

Xiao Guo

Department of Statistics and Finance
University of Science and Technology of China
Hefei, Anhui, 230026, P.R. China
(xiaoguo@ustc.edu.cn)

David DEVILBISS

Department of Psychology
University of Wisconsin–Madison
Madison, WI 53706
(ddevilbiss@wisc.edu)

Yi CHAI

Biogen
Cambridge, MA 02142
(ychai.stat@gmail.com)

Muhong GAO

Department of Statistics
University of Wisconsin–Madison
Madison, WI 53706
(mhao55@wisc.edu)

Zhengjun ZHANG

Department of Statistics
University of Wisconsin–Madison
Madison, WI 53706
(zjz@stat.wisc.edu)

Identifying the network structure of a neuron ensemble beyond the standard measure of pairwise correlations is critical for understanding how information is transferred within such a neural population. However, the spike train data pose significant challenges to conventional statistical methods due to not only the complexity, massive size, and large scale, but also high dimensionality. In this article, we propose a novel “structural information enhanced” (SIE) regularization method for estimating the conditional intensities under the generalized linear model (GLM) framework to better capture the functional connectivity among neurons. We study the consistency of parameter estimation of the proposed method. A new “accelerated full gradient update” algorithm is developed to efficiently handle the complex penalty in the SIE-GLM for large sparse datasets applicable to spike train data. Simulation results indicate that our proposed method outperforms existing approaches. An application of the proposed method to a real spike train dataset, obtained from the prelimbic region of the prefrontal cortex of adult male rats when performing a T-maze based delayed-alternation task of working memory, provides some insight into the neuronal network in that region.

KEY WORDS: Conditional intensity; Generalized linear model; Neuronal response; Regularization; Spike train data.

1. INTRODUCTION

The capacity to simultaneously record spike trains of many neurons from behaving subjects has surpassed our ability to describe putative neural codes distributed across populations of neurons. Detecting the network structure or graph structure of a neuron ensemble beyond the standard measure of pairwise correlations is critical for understanding how information is transferred within such a neural population. In this application, the spike train data, which is typically collected as multivariate point processes, present statisticians with many exciting opportunities as well as significant challenges in data analysis and in interpretation of results, due to not only the complexity, massive size, and large scale, but also high dimensionality and nonstationary temporal/spatial dependence structure. First, neural spike trains of this type are generally nonstationary and exhibit a great amount of variability among repeated trials. Second, there is evidence showing that patterns of the neural spike activity among neurons may vary with different experimental

conditions. Third, the high-dimensional nature of the spike train data from these experiments makes the analysis extremely computationally intensive.

1.1 Existing Work

Several existing methods have been proposed to identify functional connections within an ensemble of neurons. For example, some early works using nonparametric methods, such as the cross-correlogram (Perkel, Gerstein, and Moore 1967) and joint peri-stimulus time histogram (Gerstein and Perkel 1969), provide useful insights and are still commonly used for analyzing the interactions between neurons. However, these methods have

some limitations. They focus on the pairwise relationship alone, ignoring either the possible connections with other neurons in the ensemble or the influences from external stimuli. Furthermore, correlation-based analysis is suitable for describing the linear aspect of connectivity, which might be inadequate for depicting neuronal spike activities. Recently, the model-based approaches draw a great deal of attention from researchers, in which a specific point process model is assumed, including the inhomogeneous Poisson process or more generally Cox point process (Cox and Isham 1980).

In the literature, the conventional GLM (McCullagh and Nelder 1989) has been widely used to analyze the spike point processes where the firing rate of a single neuron is modeled as a function of spiking history of concurrently recorded ensemble neurons (Brillinger 1992). This GLM approach has been successfully applied to spike train data from many different types of experiments and becomes a very powerful and efficient tool of neural encoding and decoding (Truccolo et al. 2005; Pillow et al. 2008). One significant advantage of the GLM approach is that the connections between every pair of neurons can be modeled as parameters of the GLM, followed by being analyzed simultaneously under the general framework (Okatan, Wilson, and Brown 2005). In this setting, the influence from one neuron to another one is described by a set of temporal interactive parameters whose sign and magnitude indicate how neurons interact with each other. A more comprehensive review can be found in Brown, Kass, and Mitra (2004) and Truccolo (2010). We will focus mainly on the GLM framework in the rest of our article; nevertheless, it is worth noting that other statistical models, such as the Cox model (Berry et al. 2012) and the dynamical Bayesian network (Eldawlatly et al. 2010), are also useful tools for inferring the functional connections among ensemble neurons.

It is well known that the maximum likelihood estimator (MLE) is generally applicable to estimate model parameters in GLM, but it is also known to be vulnerable to over-fitting problems, which frequently arise from analysis of high-dimensional data. For a typical real neural spike train dataset, the spiking rate is often very low and the design matrix could be very sparse; for example, more than 90% of its entries are zeros. In such cases, the MLE may not be reliable. Therefore, some regularization techniques can be applied to obtain well behaved solutions to over-parameterized estimation problems (Bickel and Li 2006). On the other hand, a sparse solution is desirable, since the neurons may not be connected to all other neurons in a large population and the sparsity could improve the interpretability. Since the standard MLE does not automatically produce a sparse solution and the estimated parameters using MLE are almost surely nonzero, some additional steps (e.g., multiple hypothesis testing procedures controlling either the family-wise error rate or false discovery rate as in Gerhard et al. (2011); Kim et al. (2011), and Berry et al. (2012)) are usually needed to select the significant parameters that correspond to those truly functional connections among neurons.

For above reasons, a class of nonsmooth penalties, including the L_1 penalty or Lasso (Tibshirani 1996), and the SCAD (Fan 1997), are introduced to achieve the goal of both obtaining a stable estimate and encouraging a sparse solution. The L_1 regularized GLM has been studied in the neuroscience literature (Kelly et al. 2010; Chen et al. 2011; Mishchenko, Vogelstein,

and Paninski 2011; Zhao et al. 2012) due to its simplicity, easy implementation, and good performance. Furthermore, regularization or penalization can also be interpreted as imposing a prior on the parameters from a Bayesian perspective, and the regularized log-likelihood can be viewed as the log-posterior density of the parameters (Stevenson et al. 2009). Thus, the regularized maximum likelihood estimate is equivalent to the maximum a posteriori estimate from the Bayesian point of view.

1.2 Motivation for Our Work

However, to our knowledge, little published work on neural spike train data has fully considered the structural information of the parameter space, in which all parameters related to the interaction between one specific pair of neurons naturally form a group and the whole group could be better estimated together. Instead of treating all parameters individually, we wish to be able to make decisions jointly with the grouping information and promote the structured sparsity, that is, select the entire group of parameters, or make the inclusion of some parameters depend on the inclusion of other parameters. Some recent works in the statistics literature have explored this direction by the use of group or hierarchical norm penalties and illustrated some promising properties, particularly the group Lasso (Yuan and Lin 2006; Meier, Van De Geer, and Bühlmann 2008), the sparse group Lasso (Chatterjee et al. 2012; Simon et al. 2013), and others (Liu and Ye 2010).

In this article, we will study the spike train data by adopting the generalized linear model (GLM) framework from Brillinger (1992), which has shown the superiority in modeling the spiking activities of neurons (Truccolo et al. 2005; Devilbiss, Jenison, and Berridge 2012). However, we find that certain structural information of the parameter space in the GLM has not been fully used in the previous works. This motivates us to propose a “structural information enhanced” (SIE) regularization method for the spike train data to better investigate the functional connectivity within a neuronal network.

In this article, we propose a new “structured” regularization method for the spike train data to incorporate the structural prior information into the modeling procedure and guide the selection of parameters according to the underlying structure of the parameter space. Main contributions of our work are listed in the following parts.

- We introduce the SIE-GLM, a group-structured regularized GLM, in the context of multiple spike train data and demonstrate that our method performs better than existing approaches on simulated spike train data.
- The proposed method can be shown to be estimation consistent.
- A fast and efficient algorithm, called “accelerated full gradient update” (AFGU), is developed to efficiently handle the complex penalty in the SIE-GLM for large sparse datasets applicable to spike train data.
- The prefrontal cortex (PFC) plays an important role in cognitive and behavioral processes. An application of the proposed method to a real spike train dataset, obtained from the prelimbic region of the prefrontal cortex (pPFC) of adult male rats when performing a T-maze based delayed-

alternation task of working memory, provides some insight into the neuronal network in that region.

The rest of the article will proceed as follows. Section 2 introduces the proposed SIE-GLM in the context of multiple neural spike train data and proposes the AFGU algorithm for the group-structured regularization. Section 3 explores theoretical properties of the proposed method. Section 4 presents simulation results to assess our method. Section 5 describes a neural spike train dataset recorded from animals performing a cognitive task and illustrates the efficacy of our proposed method. Section 6 concludes with a brief discussion. Technical assumptions and detailed proofs are relegated to the appendix in the supplementary materials available online.

2. METHODOLOGICAL DEVELOPMENT

We start with a brief description of modeling neural spike train data in Section 2.1, followed by the proposed SIE-GLM modeling, parameter estimation, and implementation algorithm in Sections 2.2 and 2.3.

2.1 Background of Modeling Neural Spike Train Data

Consider an ensemble consisting of C neurons. The spike train data are usually collected as multivariate (i.e., C -variate) point processes, denoted by

$$0 < u_{c,1} < u_{c,2} < \dots < u_{c,J_c} \leq T, \quad \text{for } c = 1, \dots, C,$$

where J_c is the total number of observed spikes by neuron c during the experiment, $\{u_{c,i}\}_{i=1}^{J_c}$ are the timestamps when neuron c spikes, and T is the time length of the experiment trial. Here, we will take a discrete-time approach, where the whole experiment time period $(0, T]$ is divided into n equally spaced time bins, $\tau_k \equiv (t_{k-1}, t_k]$, each of length $\delta = T/n$, that is, $t_k = k\delta$, $k = 0, 1, \dots, n$. Denote by $N_c(\tau_k)$ the number of spikes fired by neuron c within the k th time bin τ_k , and by $N_{1:C}(\tau_{0:k})$ the spiking history of all neurons up to the time point t_k , where $\tau_{0:k} = (0, t_k]$.

We model the distribution of the Poisson-type variable, $N_c(\tau_k)$, by means of the generalized linear model (GLM) for the conditional intensity. Here, the conditional intensity function of neuron c , denoted by

$$\lambda_c(\tau_k | \cdot) = E\{N_c(\tau_k) | \cdot\},$$

takes into account several inputs at time t_k or earlier and has the form,

$$\begin{aligned} \log\{\lambda_c(\tau_k | N_{1:C}(\tau_{0:(k-1)}), \mathbf{X}(\tau_k))\} &= \gamma_{c;0} + \sum_{p=1}^P \gamma_{c;p} N_c(\tau_{k-p}) \\ &+ \sum_{i \in \{1, \dots, C\} \setminus c} \left\{ \sum_{q=1}^Q \gamma_{c,i;q} N_i(\tau_{k-q}) \right\} + \mathbf{X}(\tau_k)^T \boldsymbol{\beta}_c, \quad k=1, \dots, n, \end{aligned} \quad (2.1)$$

where

- $\exp(\gamma_{c;0})$ with the intercept $\gamma_{c;0}$ represents the baseline firing rate of neuron c .

- $\sum_{p=1}^P \gamma_{c;p} N_c(\tau_{k-p})$ is the effect of the intrinsic spiking history of neuron c , up to P bins into the past with $\gamma_{c;p}$ representing the autoregressive parameter at lag p .
- $\sum_{q=1}^Q \gamma_{c,i;q} N_i(\tau_{k-q})$ describes the influence of neuron i on neuron c . Such influence is computed based on a spiking history window, up to Q bins into the past where $\gamma_{c,i;q}$ represents the interactive parameter at lag q . For example, if $\gamma_{c,i;q}$ is positive, neuron i would functionally excite neuron c after time lag of q bins. The larger the value of $\gamma_{c,i;q}$, the stronger the excitatory drive.
- In (2.1), $\mathbf{X}(\tau_k) = (X_{\tau_k,1}, \dots, X_{\tau_k,M})^T$ is an M -dimensional vector representing some extrinsic covariates measured during the k th time bin, τ_k , while $\boldsymbol{\beta}_c = (\beta_{c,1}, \dots, \beta_{c,M})^T$ represents the vector of corresponding parameters. For example, $\mathbf{X}(\tau_k)$ includes several status indicators related to the performance of the animals in a behavioral task for electrophysiological recordings (described in Section 5; see, e.g., Table S6 in the online supplementary materials).

Model (2.1) enables us to simultaneously handle several factors, including the spike history, neuronal network structure, and other covariates. Note that we distinguish the influence of each neuron's own spiking history from the influences coming from other neurons, and allow the lengths P and Q of the history windows to be different to achieve greater flexibility.

Regarding the model assumption, we assume that the numbers of spikes randomly fired by different neurons are conditionally independent given the complete spiking history of the whole ensemble and covariates $\mathbf{X}(\tau_k)$, that is, $N_i(\tau_k)$ is independent of $N_j(\tau_k)$ given $N_{1:C}(\tau_{0:(k-1)})$ and $\mathbf{X}(\tau_k)$ for any $1 \leq i, j \leq C$, with $i \neq j$ and $1 \leq k \leq n$. Under this GLM framework, the conditional likelihood of the observed spike trains is then written as

$$\prod_{c=1}^C \left[\prod_{k=1}^n \frac{\{\lambda_c(\tau_k | \cdot)\}^{N_c(\tau_k)} \exp\{-\lambda_c(\tau_k | \cdot)\}}{\{N_c(\tau_k)\}!} \right].$$

The conventional MLEs of parameters in (2.1) can be obtained by minimizing the negative log-likelihood,

$$\sum_{c=1}^C \ell_c(\tilde{\boldsymbol{\theta}}_c), \quad (2.2)$$

with respect to $\{\tilde{\boldsymbol{\theta}}_c\}_{c=1}^C$, where

$$\begin{aligned} \ell_c(\tilde{\boldsymbol{\theta}}_c) &= -\frac{1}{n} \sum_{k=1}^n [N_c(\tau_k) \log\{\lambda_c(\tau_k | \cdot)\} - \lambda_c(\tau_k | \cdot) \\ &\quad - \log(\{N_c(\tau_k)\}!)] \end{aligned} \quad (2.3)$$

is the negative log-likelihood of a single neuron c , and the vector

$$\tilde{\boldsymbol{\theta}}_c = (\gamma_{c;0}, \{\gamma_{c;p}\}_{p=1, \dots, P}, \{\gamma_{c,i;q}\}_{i \in \{1, \dots, C\} \setminus c; q=1, \dots, Q}, \{\beta_{c,m}\}_{m=1, \dots, M})^T$$

collects all parameters for neuron c , including the baseline firing rate $\gamma_{c;0}$ as the intercept. We shall see that minimizing (2.2) is separable for each neuron, and thus can be equivalently solved by minimizing $\ell_c(\tilde{\boldsymbol{\theta}}_c)$ for each neuron c . Therefore, we will analyze each neuron individually in the rest of the article.

2.2 Proposed SIE-GLM and Estimation

Regularization or penalization is a very useful technique aiming at obtaining well-behaved solutions in some large-dimensional problems with sparse signals. Given the large dimensionality and large portion of zero entries in spike train data, the nonregularized estimation usually suffers from the overfitting problem and results in unstable solutions. Furthermore, the interactive parameters $\gamma_{c,i;q}$ representing the functional connection between neuron cells are assumed to be sparse. However, the nonregularized estimator, that is, the conventional MLE, fails to supply a sparse estimation of those parameters.

Here, we wish to propose an appropriate regularization for estimating parameters in model (2.1). Among numerous variable selection methods that are recently developed based on regularization, Lasso using the L_1 penalty (Tibshirani 1996) is one of the most popular methods due to its simplicity and good performance. The general form of a Lasso solution is defined as follows,

$$\min_{\tilde{\theta}} \{ \ell(\tilde{\theta}) + \eta \|\theta\|_1 \},$$

where $\ell(\tilde{\theta})$ is usually the negative log-likelihood function or other appropriate forms of loss functions, θ is the vector of all parameters in $\tilde{\theta}$ excluding the intercept, η is a tuning parameter, and $\|\cdot\|_1$ denotes the L_1 norm of a vector, that is, the sum of absolute values of all coordinates. Such L_1 regularized GLM has been studied in some recent work, including Kelly et al. (2010) and Zhao et al. (2012), and it shows a lot of advantages and provides significant improvement.

However, Lasso by design can only treat all parameters individually, but is not able to incorporate the structural information inherent in a particular dataset. As mentioned earlier, for any two different neurons, for example, neuron c and neuron i , the parameters $\{\gamma_{c,i;q} : q = 1, \dots, Q\}$ together describe one set of interactive parameters, which induces a natural “grouping” of the parameters in our problem. It is thus natural to design alternative regularization methods that can promote a structured sparse solution. Recently, Simon et al. (2013) proposed the sparse group lasso (SGL), which can perform the selection of parameters at both the group level and the individual level. The general form of a SGL solution is

$$\min_{\tilde{\theta}} \left\{ \ell(\tilde{\theta}) + (1 - \alpha)\eta \sum_{g=1}^G \sqrt{p_g} \|\theta^{(g)}\|_2 + \alpha\eta \|\theta\|_1 \right\},$$

where θ is partitioned into G groups so that $\theta = (\theta^{(1)T}, \dots, \theta^{(G)T})^T$, p_g is the length of parameters in $\theta^{(g)}$ corresponding to the g th group, and $\|\cdot\|_2$ denotes the L_2 norm of a vector. Here, both $\alpha \in [0, 1]$ and $\eta > 0$ are tuning parameters. The SGL penalty term is a convex combination of the Lasso and group Lasso penalties controlled by α . Smaller values of α promote the sparsity at the group level, while larger values of α encourage the individual sparsity.

Motivated by the structured penalties mentioned above, we now formulate the regularized SIE-GLM in our setting as follows. Recall that parameters in (2.1) are grouped in two types, one corresponding to the autoregressive parameters, denoted by

$$\gamma^{(c)} = (\gamma_{c;1}, \dots, \gamma_{c;P})^T,$$

and another corresponding to the interactive parameters, denoted by

$$\gamma^{(c,i)} = (\gamma_{c,i;1}, \dots, \gamma_{c,i;Q})^T, \quad \text{where } i \in \{1, \dots, C\} \setminus c. \quad (2.4)$$

In the meantime, the parameters in

$$\beta_c = (\beta_{c,1}, \dots, \beta_{c,M})^T$$

for the covariates $X(\tau_k)$ are not grouped together and are treated as stand-alone variables. Following (2.3), our proposed SIE-GLM estimator $\hat{\tilde{\theta}}_c = (\hat{\gamma}_{c;0}, \hat{\theta}_c^T)^T = (\hat{\gamma}_{c;0}, \{\hat{\gamma}_{c;p}\}_{p=1,\dots,P}, \{\hat{\gamma}_{c,i;q}\}_{i \in \{1,\dots,C\} \setminus c; q=1,\dots,Q}, \{\hat{\beta}_{c,m}\}_{m=1,\dots,M})^T$ minimizes

$$L_c(\tilde{\theta}_c) = \ell_c(\tilde{\theta}_c) + \mathcal{P}_{(\eta_c, \alpha_c)}(\theta_c), \quad (2.5)$$

where $\ell_c(\tilde{\theta}_c)$ is as defined in (2.3) and

$$\begin{aligned} \mathcal{P}_{(\eta_c, \alpha_c)}(\theta_c) = & (1 - \alpha_c)\eta_c \left\{ \sqrt{P} \|\gamma^{(c)}\|_2 + \sqrt{Q} \sum_{i \in \{1,\dots,C\} \setminus c} \|\gamma^{(c,i)}\|_2 \right\} \\ & + \alpha_c \eta_c \left\{ \|\gamma^{(c)}\|_1 + \sum_{i \in \{1,\dots,C\} \setminus c} \|\gamma^{(c,i)}\|_1 \right\} + \eta_c \|\beta_c\|_1 \end{aligned} \quad (2.6)$$

denotes the structured penalty term. As a comparison, the L_1 penalty uses $\eta_c \|\theta_c\|_1$ and is a special case of (2.6) corresponding to $\alpha_c = 1$.

Note that the penalty terms can also be easily extended to adapt to more complex types of hierarchical structures; see, for example, in Liu and Ye (2010).

2.2.1 Tuning Parameter Selection. We choose the tuning parameters η_c and α_c to minimize the BIC function

$$\text{BIC}_c(\hat{\tilde{\theta}}_c) = 2 \ell_c(\hat{\tilde{\theta}}_c) + \text{df}(\hat{\tilde{\theta}}_c) \log(n)/n, \quad (2.7)$$

where $\ell_c(\cdot)$ is defined in (2.3) and $\text{df}(\hat{\tilde{\theta}}_c) = \alpha_c \{ \sum_{p=1}^P \mathbb{I}(\hat{\gamma}_{c;p} \neq 0) + \sum_{i \in \{1,\dots,C\} \setminus c} \mathbb{I}(\hat{\gamma}_{c,i;q} \neq 0) + \sum_{m=1}^M \mathbb{I}(\hat{\beta}_{c,m} \neq 0) \} + (1 - \alpha_c) \{ \mathbb{I}(\|\hat{\gamma}^{(c)}\|_2 \neq 0) + \sum_{i \in \{1,\dots,C\} \setminus c} \mathbb{I}(\|\hat{\gamma}^{(c,i)}\|_2 \neq 0) + \sum_{m=1}^M \mathbb{I}(\hat{\beta}_{c,m} \neq 0) \}$. $\text{BIC}_c(\hat{\tilde{\theta}}_c)$ is a bivariate function of (η_c, α_c) , since $\hat{\tilde{\theta}}_c$ and $\hat{\theta}_c$ both depend on η_c and α_c , that is, $\hat{\tilde{\theta}}_c = \hat{\tilde{\theta}}_c^{\eta_c, \alpha_c}$ and $\hat{\theta}_c = \hat{\theta}_c^{\eta_c, \alpha_c}$. The minimizer is found by the grid search. We select $\alpha_c \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, and for each given α_c , the grid points for η_c are $\{\eta_{\max} h^i : i = 0, \dots, 12\}$, where $\eta_{\max} = \sup\{\eta_c : \text{df}(\hat{\tilde{\theta}}_c^{\eta_c, \alpha_c}) > 0\}$ and $h \in (0, 1)$ are constants with η_{\max} depending on α_c . In practice, we may have to drop the small values of η_c in case that the algorithm doesn't converge.

Note that AIC as another popular model selection criterion is not suitable for our purpose, because AIC usually performs better for prediction but not very well for variable selection. While the main objective of the study is to learn the functional connectivity between neurons, that is, nonzero interactive parameters, BIC is more capable of selecting truly nonzero parameters.

2.3 Proposed AFGU Algorithm for the SIE-GLM

Recall that Lasso and SGL can produce sparse solutions because the penalty terms are not differentiable at the point 0. However, this feature, on the other hand, also brings computational difficulty in solving the minimization problem. The

standard convex optimization used to obtain MLE, like Newton-Raphson method, is not directly applicable to nonsmooth objective functions. There has been a tremendous amount of work on regularized optimization from both statistics and computer science perspective. Recently, the coordinate descent (CD) algorithm, rediscovered by Friedman et al. (2007), has gained lots of attention in regularized linear and logistic regression and was shown to have computational superiority. Friedman, Hastie, and Tibshirani (2010) and Simon et al. (2013) applied a similar idea and developed a block-wise coordinate descent (BCD) algorithm for SGL.

However, we also find that while the BCD algorithm is very fast and scales well in the linear regression model, it can be quite costly in the GLM when the sample size, that is, the number of bins in the spike train data, is very large. Based on our initial simulation study, the BCD algorithm could not well handle the large dataset whose size is similar to the real spike train data.

We develop an AFGU algorithm motivated from the previous work in Kim, Kim, and Kim (2008), Beck and Teboulle (2009), and Wright (2012). Our proposed AFGU algorithm is based on the full gradient of log-likelihood function and a specific shrinkage-thresholding operator depending on the penalty function, combined with a Newton-based acceleration technique over active parameters. Compared with BCD, the AFGU algorithm could improve the performance in two ways.

First, we use the full gradient with respect to the parameter vector $\tilde{\theta}_c$ instead of either the coordinate-wise or block-wise gradient to avoid evaluating the gradient at each coordinate one at a time. Starting from the current estimate $\tilde{\theta}_c^{(old)}$, the full gradient update is to obtain $\tilde{\theta}_c^{(new)}$, which minimizes

$$\frac{\rho}{2} \left\| \tilde{\theta}_c - \tilde{\theta}_c^{(old)} \right\|_2^2 + \left\{ \nabla \ell_c \left(\tilde{\theta}_c^{(old)} \right) \right\}^T \left(\tilde{\theta}_c - \tilde{\theta}_c^{(old)} \right) + \mathcal{P}_{(\eta_c, \alpha_c)}(\tilde{\theta}_c), \quad (2.8)$$

where $\nabla \ell_c(\tilde{\theta}_c)$ denotes the gradient vector of $\ell_c(\tilde{\theta}_c)$ with respect to $\tilde{\theta}_c$, and ρ is some small positive constant to control the step size of the gradient update, which will be updated adaptively in the algorithm. The minimization in (2.8) is separable between groups and the sub-problem for each group can be analytically solved as in Simon et al. (2013).

Second, for each iteration, after the gradient update step in (2.8), a Newton-type acceleration step is performed on the reduced parameter space that only consists of the coordinates with nonzero parameters at the current estimate. Let \mathcal{A} be the index set of the nonzero elements of $\tilde{\theta}_c^{(new)}$. Denote by $\nabla_{\mathcal{A}} L_c(\tilde{\theta}_c)$ and $\nabla_{\mathcal{A}}^2 L_c(\tilde{\theta}_c)$ the reduced gradient vector and Hessian matrix of $L_c(\tilde{\theta}_c)$ with respect to $\tilde{\theta}_{c,\mathcal{A}}$, the sub-vector of $\tilde{\theta}_c$ with indices in \mathcal{A} . The reduced-Newton update for $\tilde{\theta}_c^{(new)}$ is then simply

$$\tilde{\theta}_{c,\mathcal{A}}^{(new)} - \left\{ \nabla_{\mathcal{A}}^2 L_c \left(\tilde{\theta}_c^{(new)} \right) \right\}^{-1} \nabla_{\mathcal{A}} L_c \left(\tilde{\theta}_c^{(new)} \right). \quad (2.9)$$

To estimate $\tilde{\theta}_c$, we can perform the updates in (2.8) and (2.9) iteratively until convergence. The acceleration step in (2.9) generally yields a vast performance improvement over the full gradient update method in (2.8) and the BCD method, which both use only the first-order information, and the improve-

ment is worth the cost of evaluating the reduced Hessian. When the true parameters are sparse, the reduced Hessian has a small dimension and its inverse can be easily computed. Therefore, the AFGU algorithm is expected to run faster and theoretically converges to the optimum point Q-quadratically, which is better than either the simple gradient or the coordinate-wise gradient method with a Q-linear convergence rate (Wright 2012). Some fine adjustments are also made to efficiently handle the sparse neural data, which have a large portion of zero entries, and stabilize some matrix operations.

As mentioned earlier, Simon et al. (2013) used the BCD algorithm to solve the SGL problem for the linear and logistic regressions. To estimate the parameters, they proposed an objective function similar to (2.8) [see eq. (10) therein], and minimize it iteratively in a group-wise way without the second acceleration step. In our work, the idea of the BCD algorithm is applicable to solve problem (2.5). Specifically, BCD minimizes (2.8) with respect to the parameters in one group at a time and cyclically iterates through the groups. Therefore, the AFGU algorithm is computationally more efficient than BCD in the sense that it updates all parameters by (2.8) simultaneously and uses the reduced-Newton update in (2.9).

3. THEORETICAL PROPERTIES

In this section, we provide theoretical properties of our proposed method. Here, we allow the total number C of neurons to diverge slowly with the length T of the experiment, or equivalently, with the number n of time bins when $\delta = T/n$ is fixed. Let $\tilde{\theta}_c^* = (\theta_0^*, \theta_1^*, \dots, \theta_d^*)^T$ be the vector of unknown true parameters, where $d = P + (C - 1)Q + M$.

Theorem 1 guarantees the existence of a consistent local minimizer of (2.5) and such minimizer is $\sqrt{n/C}$ -consistent.

Theorem 1. Assume Conditions A1–A3 in the supplementary appendix (available online only). If $\eta_c \sqrt{n} = O(1)$ and $C^4/n = o(1)$ as $n \rightarrow \infty$, then there exists a local minimizer $\hat{\tilde{\theta}}_c$ of (2.5) such that $\|\hat{\tilde{\theta}}_c - \tilde{\theta}_c^*\|_2 = O_P(\sqrt{C/n})$.

We would like to point out that the proof of Theorem 1 uses the penalized Bregman divergence (BD) framework in Zhang, Jiang, and Chai (2010), but differs in two parts. First, we need to treat the BD more carefully, since samples in Zhang, Jiang, and Chai (2010) are iid, but samples in the spike train dataset, that is, $\{N_c(\tau_k) : k = 1, \dots, n\}$, are not iid. We will use the martingale version of central limit theorem (Theorem 7.4 in Durrett 2004) to establish the desired result in the proof. Second, the penalty terms used in our SIE-GLM are much more complex than the L_1 and those penalties in Zhang, Jiang, and Chai (2010) that are imposed on individual variables. Details can be found in the supplementary appendix available online.

4. SIMULATION STUDIES

To illustrate the application and performance of the proposed regularized SIE-GLM, we consider three types of network structures among neurons. The simple structure is addressed in Section 4.1, the complex structure is illustrated in Section 4.2,

whereas the network structure mimicking the real data is presented in Section 4.3.

We compare the performance of the proposed AFGU algorithm for the SGL regularized method “SGL- P - Q ” with two existing L_1 regularized methods “ L_1 -short- P - Q ” (Zhao et al. 2012) and “ L_1 - P - Q ” in different settings, with varying lengths P and Q of the history windows. Note that “ L_1 -short- P - Q ” uses a different parameterization as follows,

$$\log\{\lambda_c(\tau_k \mid N_{1:C}(\tau_{0:k-1}), \mathbf{X}(\tau_k))\} = \gamma_{c;0} + \sum_{p=1}^P \gamma_{c;p} N_c(\tau_{k-p}) + \sum_{i \in \{1, \dots, C\} \setminus c} \gamma_{c,i;1} \left\{ \sum_{q=1}^Q N_i(\tau_{k-q}) \right\} + \mathbf{X}(\tau_k)^T \boldsymbol{\beta}_c, \quad (4.1)$$

where Q is chosen to be small and the interactive parameters $\{\gamma_{c,i;q} : q = 1, \dots, Q\}$ in (2.4) are simplified to a single parameter $\gamma_{c,i;1}$, that is,

$$\gamma^{(c,i)} = \gamma_{c,i;1}, \quad \text{where } i \in \{1, \dots, C\} \setminus c. \quad (4.2)$$

The corresponding parameter vector is

$$\tilde{\boldsymbol{\theta}}_c = (\gamma_{c;0}, \{\gamma_{c;p}\}_{p=1, \dots, P}, \{\gamma_{c,i;1}\}_{i \in \{1, \dots, C\} \setminus c}, \{\beta_{c,m}\}_{m=1, \dots, M})^T,$$

and the imposed L_1 penalty is

$$\mathcal{P}_{\eta_c}(\boldsymbol{\theta}_c) = \eta_c \left(\sum_{p=1}^P |\gamma_{c;p}| + \sum_{i \in \{1, \dots, C\} \setminus c} |\gamma_{c,i;1}| \right) + \eta_c \|\boldsymbol{\beta}_c\|_1. \quad (4.3)$$

The “ L_1 - P - Q ” method uses the parameterization as in (2.1) without simplification and imposes the L_1 penalty as follows,

$$\mathcal{P}_{\eta_c}(\boldsymbol{\theta}_c) = \eta_c \left\{ \sum_{p=1}^P |\gamma_{c;p}| + \sum_{i \in \{1, \dots, C\} \setminus c} \left(\sum_{q=1}^Q |\gamma_{c,i;q}| \right) \right\} + \eta_c \|\boldsymbol{\beta}_c\|_1. \quad (4.4)$$

4.1 Simple Network

We first simulate an ensemble of 10 neurons with the network structure displayed in Figure S1 (in the online supplementary materials). There are 10 connections in the constructed network, of which seven connections have type-A parameters (with thick red line in Figure S1, available online) and three connections have type-B parameters (with thin blue line in Figure S1, available online). For each neuron, the baseline rate was set at $\gamma_{c;0} = -3$, which gives an average firing rate at $\exp(-3)/0.1 = 0.5$ Hz when the bin size is 0.1 sec. The lengths of the history window of the autoregressive parameters and the interactive parameters are set to be 10 bins in the generating process. The autoregressive parameters are modeled by setting the first few parameters $\gamma_{c;p}$ to be very negative and then rise to be slightly positive before going back to zero, in a way similar to Truccolo et al. (2005); see the left panel of Figure S2 (available online). Two possible forms of interactive parameters between neurons are also illustrated in the middle and right panels of Figure S2 (available online). For illustrative simplicity, the covariate term $\mathbf{X}(\tau_k)^T \boldsymbol{\beta}_c$ is not included in the simulation.

We simulate a spike train with $n = 10,000$ and $n = 15,000$ bins, which correspond to $T = 1000$ sec and $T = 1500$ sec when δ is fixed at 0.1 sec. The following five methods are compared. The tuning parameters are selected by BIC given in (2.7) for all methods. We also tried the SCAD penalty (Fan 1997), which performs similar to the L_1 regularized methods (and thus is not shown here). In practice, to calculate the parameter estimators, the group structure is used in the algorithm for methods III–V below, but, for methods I and II, the structural information is not incorporated.

- method I. (L_1 -short-10-3) Use parameterization (4.1), L_1 penalty (4.3), $P = 10$, and $Q = 3$;
- method II. (L_1 -10-10) Use parameterization (2.1), L_1 penalty (4.4), $P = 10$, and $Q = 10$;
- method III. (SGL-5-5) Use parameterization (2.1), SGL penalty (2.6), $P = 5$, and $Q = 5$;
- method IV. (SGL-10-10) Use parameterization (2.1), SGL penalty (2.6), $P = 10$, and $Q = 10$;
- method V. (SGL-15-15) Use parameterization (2.1), SGL penalty (2.6), $P = 15$, and $Q = 15$.

While method IV uses the SGL penalty with the true length of the history window, method III and V intend to investigate the performance of the SGL regularized method with an inexact length of the history window, since we do not know how long the interaction between neurons would last in practice. For a better comparison, we also carry out the simulation under the setting that the interactive parameter is 25% stronger, that is, the magnitude of parameters is 25% larger than those given in Figure S2 (available online). We simulate spike trains with different lengths to see how the result changes.

The results of all methods are summarized in Tables S1 and S2 (in the online supplementary materials), which also include comparisons with the method in Tandon and Ravikumar (2014) (abbreviated as the TR method). “Correct-All” represents the number of all connections correctly detected by methods where there is indeed a connection between two neurons in the true network. “detected-A” and “detected-B” are numbers of all detected connections that embrace the true connections represented as type-A and type-B, respectively. “Correct-NC” is the number of pairs correctly identified as no connections where there is actually no connection. For each of the four metrics, a larger value means a better performance. All values represent the average across 100 simulation runs for each length of spike train and strength of parameters.

From Tables S1 and S2 (available online), all methods can successfully detect the sparse structure of the network, that is, most truly zero-valued parameters are estimated to be zero, reflecting very good levels of specificity. However, the sensitivity of detecting significant interactive parameters is not as good as specificity. When the relative strength of interactive parameters increases, we can see that the sensitivity becomes better. Such an effect is reasonable, since stronger signals will result in bigger changes to the conditional intensity, which makes the interaction easier to be detected. When we increase the length of the spike train to 15,000 bins, we see improved performance from all methods, which would be expected since we simply have larger dataset and more information to study the network structure of the neuron population. Among all methods, the SGL regularized

method with the exact length of parameters' history window has the best performance in finding the interactions between neurons. Interestingly, even when we misspecified the length of the history window of interactive parameters in methods III and V, they can still get the comparable results. All three SGL regularized methods outperform the other two L_1 regularized methods.

4.2 Complex Network

We simulate an ensemble of 60 neurons with the more complex network structure illustrated in Figure S3 (available online) and the corresponding connection matrix in Figure S4 (available online). There, there are 60 connections, among which 30 have type-A parameters and 30 have type-B parameters. The other settings are similar to the previous simulation except that we simulate with longer spike train (using $n = 20,000$ and $25,000$) to handle the larger network.

The results of this simulation are summarized in Table S3 (available online). Besides those patterns that are already discussed in Section 4.1, we can clearly see from Figure S5 (available online) that the SGL regularized methods perform considerably better than two L_1 based methods. For example, method IV is able to identify 12 more true connections on average and much fewer false discoveries than method I when $n = 20,000$ and the relative strength is 125%. The estimated network structures by the SGL regularized method are already very close to the true simulation network in most runs.

For all methods, we also find that it is much harder to detect the connections of type-B interactive parameters than those of type-A interactive parameters. Since type-A parameter is more excitatory (more positive interactive parameters in $\gamma^{(c,i)}$) and type-B parameter is more inhibitory (more negative interactive parameters in $\gamma^{(c,i)}$), it is likely that the initially low baseline firing rate makes it difficult to detect inhibitory influences given the relatively short length of simulated spike train. Figure S6 (in the online supplementary materials) illustrates the estimated parameters of all type-A and type-B connections in a single simulation run by L_1 -10-10 and SGL-10-10, in which thick circles connected with dashed lines represent the true values and each thin dashed line represents the estimated parameters for one true connection with type-A or type-B. (Since L_1 -short treats the whole interactive parameters as one single parameter, it cannot provide the detailed structural form of parameters.) We find that the L_1 -10-10 method usually can only detect the high peak at the beginning, while the SGL-10-10 method can provide more informative estimation of the parameters.

4.3 Simulated Network that Mimics the Real Data

To mimic the setting we encounter in the real experiments, we will simulate the spike train data based on the estimated networks from the real data. First, obtain the estimated parameters and network structure by applying method SGL-30-10 to the neurophysiological data and denote by $\hat{\beta}_c$ the estimate of β_c . For the simulation study (with $P = 30$ and $Q = 10$ in model (2.1)), except for β_c and $\gamma_{c,i;q}$, the true parameters as well as the network structure are set as the estimated ones acquired previously. The true value of β_c is $0.3 \text{ sign}(\hat{\beta}_c)$, and values of $\gamma_{c,i;q}$ are similar to those in Figure S2 (available online) but with

relative strength 125%, where type-A corresponds to the positive values in the estimated network and type-B corresponds to the negative ones. The status indicators collected in the real experiments (given in Table S6, available online) are included in this section as the covariates.

For illustrative purpose, we only present the simulation results using the estimated networks (given in Figure S9, available online) from Experiments 1 and 4. The comparison among regularized methods is given in Table S4 (available online), which reveals that all methods perform similar in terms of correctly identifying more connections and SGL-30-10 well maintained low number of false detections. In contrast, the other two L_1 based methods both produce much more false alarms, which is not desirable in the real applications.

5. APPLICATION TO NEUROPHYSIOLOGICAL DATA

5.1 T-Maze Task of Working Memory

We aim to estimate the functional connectivity structure of neurons in the rat prelimbic region of the prefrontal cortex (plPFC) using the proposed SIE-GLM. Neural data were obtained from adult male Sprague-Dawley rats performing a T-maze based delayed-alternation task of working memory (Devilbiss and Waterhouse 2004; Devilbiss, Page, and Waterhouse 2006; Devilbiss, Jenison, and Berridge 2012). In brief, animals were trained to navigate down the runway of the T-maze and choose one of two arms opposite to the one previously visited for food rewards (chocolate chips 0.8 g) delivered by the experimenter's hand. For each trial, the animal was placed in a start-box for a predetermined time (a delay) and released with the removal of a starting gate. The rat would navigate to the junction of the "T" and choose either the left or right arm, as seen in Figure S7 (available online). On a correct choice, the animal was rewarded and returned in the start-box, yielding a correct sequence of "left, right, left, right, . . .". On incorrect trials, the animal was removed from the incorrectly chosen arm and returned to the start-box without reward. Training continued in the T-maze task until performance accuracy reached an average of 90% on 10 trials (0 sec delay, one testing session per day). A restricted feeding schedule (16 g–20 g of standard chow) maintained motivation with quantities of food titrated for each animal to maintain motivation for each 40-trial session. Animals were then surgically implanted with recording electrodes and returned to ad lib feeding for the duration of recovery (7–10 days). Following recovery, restricted feeding was reinstated and training continued until animal performance was stable across 40 trials at 90%–100% accuracy with at least a 10 sec delay. Training/testing occurred at the same time each day. All procedures were in accordance with NIH guidelines and were approved by the University of Wisconsin Institutional Animal Care and Use Committee.

5.2 Neural Data Discrimination and Recording

Online Discrimination and Recording. On the day of a recording session, the animal was tethered to customized multichannel electrophysiological hardware (Plexon Inc., Dallas, TX). Neural activity was amplified, discriminated, time stamped,

and recorded from putative single units of the pIPFC. The animal remained in its home cage, placed above the T-maze, during this discrimination phase to allow the animal to habituate to the tether and the recording arena. Template matching algorithms were applied to neural activity to preliminarily discriminate action potentials exhibiting a 3 : 1 signal-to-noise ratio. Following discrimination of pIPFC units, the animals remained tethered to the recording hardware for the remainder of the day. During each testing session, videotape recordings were made of the entire experimental session with a video counter timer providing time stamps (resolution = 0.0125 sec) synchronized to the multiunit recording systems.

Offline Unit Analysis. After each experimental session, preestablished offline criteria were used to verify that waveforms assigned to each online discriminated unit originated from a single neuron. These previously described criteria were based on unit waveform properties and spike train discharge patterns including: (i) variability of peak waveform voltage, (ii) variability of waveform slope(s) from peak to peak, (iii) separability of clustering of scattergram points from the waveform's first two principal components, and (iv) spike train autocorrelogram. Neurons that did not meet these criteria were excluded from the study. Neuronal waveform shape, discharge pattern (interspike interval), and response properties were further examined to verify that neurons were not recorded across multiple recording sessions. Further analyses of data from neurons recorded from multiple session days were limited to the first recording session of that neuron. Finally, putative pyramidal projection neurons were identified by the action potential shape. Essentially, the peak-peak (P-P) duration of waveforms from verified neurons were calculated. Neurons with P-P intervals greater than 200 μ s were classified as "wide spike" (WS-type) neurons. Following each experimental session, the behavioral intervals were visually identified by the experimenter from time-stamped video recordings and manually entered into each neural recording data file. These event-states were defined by events occurring as the animal performs the T-maze task that included

- placement of the rat into the start-box,
- removal of the start gate,
- rat reaching the branch point of the "T",
- the rat entering one of two goal arms (choice),
- receipt of food reward, and
- removal of the rat to begin another trial.

Each trial was further classified as a correct or incorrect trial as well as by chosen spatial goal (i.e., left vs. right arm). Dividing the task into these behavioral intervals should not be interpreted as representing individual and discrete cognitive functions. Instead, these are convenient delineations of different behaviors and task events where different PFC-dependent processes may be involved. Certainly, one can expect that some PFC-dependent processes may be involved in adjacent behavioral intervals, and like all delayed-response tasks, a clear temporal delineation of onset and termination of different cognitive functions is difficult.

5.3 Neurophysiological Data Analysis

The dataset comprises eight experiments, each having 40 trials (40 turns). Basic statistics are provided in Table S5 (available online). Given the length of the spike trains and the relatively low firing rate on average, we bin the spikes at 100 ms (0.1 sec), which gives about 10,000–20,000 bins for each experiment. We set $P = 30$, that is, the spike history may affect its own spontaneous firing rate up to 3 sec; and $Q = 10$, that is, the influence of spikes from other neurons lasts up to 1 sec. Other choices of P and Q are also considered, which give similar results. Finally, we code the current state of the experimental animals into eight indicator columns that are listed in Table S6 (available online). Note that we only model the parameters associated with states as linear terms, which only change the overall firing rate and do not change the connectivity network structure. Then we applied the SIE-GLM SGL regularized method to recorded neurophysiological data for each of the eight experiments.

For the autoregressive parameters $\gamma_{c,p}$, many of them are estimated to be zero. Most of the identified nonzero parameters are positive, which indicate that there are some self-exciting processes during a certain period of time. Figure S8 (in the online supplementary materials) gives the estimated autoregressive parameters for selected neurons (Nos. 14, 19, and 73) in Experiment 4, which illustrate some typical forms of autoregressive parameters. The refractory period of these neurons could not be observed in this study, given that spike train binning was 0.1 sec, well beyond the absolute refractory period of approximately 0.003 sec. Note that some hilly forms are observed from Neuron 14. This cyclic phenomenon is possibly caused by mutually excitatory connections between these two neurons, rather than solely by the individual self-exciting process. Such excitatory pair or clique can also be found in the network structure of other experiments.

Table S7 (in the online supplementary materials) summarizes the number of detected functional connections into two categories: excitatory and inhibitory connections. The type of detected connection from neuron i to neuron c is determined by the sign of the sum of the estimated interactive parameters, $\text{sign}(\mathbf{1}^T \hat{\boldsymbol{\gamma}}^{(c,i)})$, where $\mathbf{1}$ is a vector of ones. We also compare the results with those from two L_1 based methods discussed in Section 4. We can see that among thousands of possible connection pairs, only a few of them are selected by regularized methods. The short-term L_1 method detects more connections than the other two methods, but based on the simulation results we may doubt that it could report several false discoveries. The long-term L_1 method reports the smallest number of connections, and the SGL should provide a good balance and better results as those in the simulation studies. For example, the estimated networks from Experiments 1 and 4 are given in Figure S9 (available online).

Also we provide the colored maps of the estimated interaction matrix in Figure S10 (available online), whose color at the i th row and the j th column represents the influence from neuron j on neuron i . The Red color is for an excitatory effect, the Blue color is for the inhibitory effect on the target neuron from the trigger neuron, and the background Green color is for no effect, where the exact color is determined by the strength of the interaction, that is, $\|\hat{\boldsymbol{\gamma}}^{(i,j)}\|_2$. (Experiments 5 and 8 are omitted

in Figure S8 (available online), since too few connections are identified, which may be due to the low baseline firing rate and short experimental time.)

The detected connections among neurons share some common characteristic across experiments. For example, there are several excitatory pairs and cliques along the diagonal lines, which suggests that the neurons in the self-exciting groups tend to be physically close in the plPFC. The number of detected inhibitory connections is fewer than that of the excitatory ones. This is likely, because only 20% of cortical neurons are inhibitory interneurons. We need to note that the term “functional connections” here refers to the statistical dependencies between spike trains (neurons) and does not necessarily imply the existence of an anatomical connection between the corresponding neurons (Okatan, Wilson, and Brown 2005); nevertheless results we observed would be still useful to guide further research.

Table S8 (in the online supplementary materials) summarizes the estimated parameters for eight state indicator covariates described in Table S6 (available online). Among all parameters, many are estimated to be insignificant, which means the spike activities are generally similar across different states of the T-maze trials. Among the eight state covariates, the “Delay”, “Run”, and “Decision” covariates have somewhat stronger differences in the number of neurons with positive and negative estimated parameters. For the “Delay” variable, there were fewer neurons with positive estimated parameters than those with negative parameters in Experiments 3–6, likely reflecting that delay-related neurons represent approximately 20% of PFC neurons. On the other hand, the large number of neurons with positive estimated parameters for “Run” and “Decision” variables suggest that a large percentage of neurons coded decision-making processes and the behavioral response.

6. DISCUSSION

When analyzing simultaneously recorded spike trains, it is desirable to have some unified model to include several factors, such as the autoregressive effect on each neuron, the functional connections between pairs of neurons, as well as the experimental state indicators. In this work, we use the regularized SIE-GLM with Poisson-type neuronal responses to achieve such a goal. The SIE-GLM framework is able to deal with different types of factors that can affect the neuron activities; meanwhile, the appropriate penalty can force parameters of those insignificant factors to be exactly zero so that the functional connections between neurons can be inferred and separated from noises.

Here, our approach is based on the belief that among the huge amount of all possible connections, only a very small portion of them are really significant, so that a sparse network can be constructed using the estimated connections from the regularized method. Comparing with previous works (e.g., Zhao et al. 2012), we further propose the use of structured regularization, which provides more flexibility to incorporate the structural prior information of the parameter space and improves the performance. We give more rigorous theoretical results when considering the dependence within the spike trains. From the simulation results, the SGL regularized method can indeed improve the sensitivity for detecting the true connections without sacrificing the speci-

ficity. The proposed AFGU algorithm is computationally very efficient under the sparse network assumption.

We then applied the proposed method to real spike train recordings from neurons in the prelimbic region of the prefrontal cortex (PFC) of adult male rats. Some interesting findings about the ensemble of neurons include the following.

- More excitatory connections are detected than inhibitory connections; network connectivity is suggestive of hub-based network architecture.
- Several distinct neural cliques could be identified that included delay-related and behavioral response-related PFC-related functions.

Although the statistically significant functional connection does not infer synaptic connections, it provides useful information to guide further research on the details of the interactions within neuronal network. In summary, our proposed method is applicable to a broad range of simultaneously recorded spike trains and expected to perform better than existing methods.

SUPPLEMENTARY MATERIALS

Online appendix: The appendix collects notations in Appendix A, detailed derivations of Theorem 1 in Appendix B, and Figures S1–S10 and Tables S1–S8 in Appendix C. (Technometrics_ZCGGDZ_online_appendix.pdf, pdf file)

Package for Matlab codes and real data: The Matlab script files, a readme file, and real data used for simulation studies and real data analysis in the article are given in the zipped file. (Technometrics_ZCGGDZ_online_Matlab_codes.zip, zipped file)

ACKNOWLEDGMENTS

The authors thank the editor, associate editor, and two anonymous referees for insightful comments. C. Zhang’s research is supported by the NSF grants DMS–1308872 and DMS–1521761, and Wisconsin Alumni Research Foundation. Z. Zhang’s research is supported by the NSF grant DMS–1505367, and Wisconsin Alumni Research Foundation.

[Received October 2014. Revised December 2015.]

REFERENCES

- Beck, A., and Teboulle, M. (2009), “A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems,” *SIAM Journal on Imaging Sciences*, 2, 183–202. [354]
- Berry, T., Hamilton, F., Peixoto, N., and Sauer, T. (2012), “Detecting Connectivity Changes in Neuronal Networks,” *Journal of Neuroscience Methods*, 209, 388–397. [351]
- Bickel, P. J., and Li, B. (2006), “Regularization in Statistics,” *TEST*, 15, 271–344. [351]
- Brillinger, D. R. (1992), “Nerve Cell Spike Train Data Analysis: A Progression of Technique,” *Journal of the American Statistical Association*, 87, 260–271. [351]
- Brown, E. N., Kass, R. E., and Mitra, P. P. (2004), “Multiple Neural Spike Train Data Analysis: State-of-the-Art and Future Challenges,” *Nature Neuroscience*, 7, 456–461. [351]
- Chatterjee, S., Steinhäuser, K., Banerjee, A., Chatterjee, S., and Ganguly, A. (2012), “Sparse Group Lasso: Consistency and Climate Applications,” in *Proceedings of the SIAM International Conference on Data Mining*, pp. 47–58. [351]
- Chen, Z., Putrino, D. F., Ghosh, S., Barbieri, R., and Brown, E. N. (2011), “Statistical Inference for Assessing Functional Connectivity of Neuronal Ensembles With Sparse Spiking Data,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19, 121–135. [351]

- Cox, D. R., and Isham, V. (1980), *Point Processes*, Boca Raton, FL: Chapman & Hall/CRC Press. [351]
- Devilbiss, D. M., Jenison, R. L., and Berridge, C. W. (2012), "Stress-Induced Impairment of a Working Memory Task: Role of Spiking Rate and Spiking History Predicted Discharge," *PLoS Computational Biology*, 8, e1002681. [351,356]
- Devilbiss, D. M., Page, M. E., and Waterhouse, B. D. (2006), "Locus Coeruleus Regulates Sensory Encoding by Neurons and Networks in Waking Animals," *Journal of Neuroscience*, 26, 9860–9872. [356]
- Devilbiss, D. M., and Waterhouse, B. D. (2004), "The Effects of Tonic Locus Coeruleus Output on Sensory-Evoked Responses of Ventral Posterior Medial Thalamic and Barrel Field Cortical Neurons in the Awake Rat," *Journal of Neuroscience*, 24, 10773–10785. [356]
- Durrett, R. (2004), *Probability: Theory and Examples*, Belmont, CA: Duxbury Press. [354]
- Eldawlatly, S., Zhou, Y., Jin, R., and Oweiss, K. G. (2010), "On the Use of Dynamic Bayesian Networks in Reconstructing Functional Neuronal Networks From Spike Train Ensembles," *Neural Computation*, 22, 158–189. [351]
- Fan, J. (1997), Comment on "Wavelets in Statistics: A Review" by A. Antoniadis, *Journal of the Italian Statistical Society*, 6, 131–138. [351,355]
- Friedman, J., Hastie, T., Hofling, H., and Tibshirani, R. (2007), "Pathwise Coordinate Optimization," *Annals of Applied Statistics*, 1, 302–332. [354]
- Friedman, J., Hastie, T., and Tibshirani, R. (2010), "A Note on the Group Lasso and a Sparse Group Lasso," *arXiv preprint*, arXiv:1001.0736. [354]
- Gerhard, F., Pipa, G., Lima, B., Neuenschwander, S., and Gerstner, W. (2011), "Extraction of Network Topology From Multi-Electrode Recordings: Is There a Small-World Effect?" *Frontiers in Computational Neuroscience*, 5, 4. [351]
- Gerstein, G. L., and Perkel, D. H. (1969), "Simultaneously Recorded Trains of Action Potentials: Analysis and Functional Interpretation," *Science*, 164, 828–830. [350]
- Kelly, R. C., Kass, R. E., Smith, M. A., and Lee, T. S. (2010), "Accounting for Network Effects in Neuronal Responses Using L1 Regularized Point Process Models," *Advances in Neural Information Processing Systems*, 23, 1099–1107. [351,353]
- Kim, J., Kim, Y., and Kim, Y. (2008), "A Gradient-Based Optimization Algorithm for LASSO," *Journal of Computational and Graphical Statistics*, 17, 994–1009. [354]
- Kim, S., Putrino, D., Ghosh, S., and Brown, E. N. (2011), "A Granger Causality Measure for Point Process Models of Ensemble Neural Spiking Activity," *PLoS Computational Biology*, 7, e1001110. [351]
- Liu, J., and Ye, J. (2010), "Moreau-Yosida Regularization for Grouped Tree Structure Learning," *Advances in Neural Information Processing Systems*, 23, 1459–1467. [351,353]
- McCullagh, P., and Nelder, J. A. (1989), *Generalized Linear Models*, London: Chapman & Hall. [351]
- Meier, L., Van De Geer, S., and Bühlmann, P. (2008), "The Group Lasso for Logistic Regression," *Journal of the Royal Statistical Society, Series B*, 70, 53–71. [351]
- Mishchenko, Y., Vogelstein, J. T., and Paninski, L. (2011), "A Bayesian Approach for Inferring Neuronal Connectivity From Calcium Fluorescent Imaging Data," *Annals of Applied Statistics*, 5, 1229–1261. [351]
- Okatan, M., Wilson, M. A., and Brown, E. N. (2005), "Analyzing Functional Connectivity Using a Network Likelihood Model of Ensemble Neural Spiking Activity," *Neural Computation*, 17, 1927–1961. [351,358]
- Perkel, D. H., Gerstein, G. L., and Moore, G. P. (1967), "Neuronal Spike Trains and Stochastic Point Processes: II. Simultaneous Spike Trains," *Biophysical Journal*, 7, 419–440. [350]
- Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., and Simoncelli, E. P. (2008), "Spatio-Temporal Correlations and Visual Signalling in a Complete Neuronal Population," *Nature*, 454, 995–999. [351]
- Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013), "A Sparse-Group Lasso," *Journal of Computational and Graphical Statistics*, 22, 231–245. [351,353,354]
- Stevenson, I. H., Rebesco, J. M., Hatsopoulos, N. G., Haga, Z., Miller, L. E., and Kording, K. P. (2009), "Bayesian Inference of Functional Connectivity and Network Structure From Spikes," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17, 203–213. [351]
- Tandon, R., and Ravikumar, P. (2014), "Learning Graphs With a Few Hubs," in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pp. 602–610. [355]
- Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society, Series B*, 58, 267–288. [351,353]
- Truccolo, W. (2010), "Stochastic Models for Multivariate Neural Point Processes: Collective Dynamics and Neural Decoding," in *Analysis of Parallel Spike Trains*, eds. S. Grün and S. Rotter, New York: Springer, pp. 321–341. [351]
- Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., and Brown, E. N. (2005), "A Point Process Framework for Relating Neural Spiking Activity to Spiking History, Neural Ensemble, and Extrinsic Covariate Effects," *Journal of Neurophysiology*, 93, 1074–1089. [351,355]
- Wright, S. J. (2012), "Accelerated Block-Coordinate Relaxation for Regularized Optimization," *SIAM Journal on Optimization*, 22, 159–186. [354]
- Yuan, M., and Lin, Y. (2006), "Model Selection and Estimation in Regression With Grouped Variables," *Journal of the Royal Statistical Society, Series B*, 68, 49–67. [351]
- Zhang, C. M., Jiang, Y., and Chai, Y. (2010), "Penalized Bregman Divergence for Large-Dimensional Regression and Classification," *Biometrika*, 97, 551–566. [354]
- Zhao, M., Batista, A., Cunningham, J. P., Chestek, C., Rivera-Alvidrez, Z., Kalmar, R., Ryu, S., Shenoy, K., and Iyengar, S. (2012), "An L_1 -Regularized Logistic Model for Detecting Short-Term Neuronal Interactions," *Journal of Computational Neuroscience*, 32, 479–497. [351,353,355,358]

Online Supplement to “Statistical Learning of Neuronal Functional Connectivity”

Appendix A: List of notations and symbols

List of notations and symbols.

- C : total number of neurons in the ensemble.
- T : total length of the experiment trial.
- n : total number of time bins. We assume $n \rightarrow \infty$.
- p_g : the length of the g th group for the parameters.
- $\delta = T/n$: the length of each time bin.
- $\tau_k = (t_{k-1}, t_k]$: the start and end time points of the k th time bin.
- $N_c(\tau_k)$: count the number of spikes fired by neuron c during τ_k .
- $\tau_{0:k} = (0, t_k]$: the interval from the start of the experiment up to the time point t_k .
- $N_{1:C}(\tau_{0:k}) = \{N_c(\tau_\ell) : c = 1, \dots, C; \ell = 1, \dots, k\}$ if $k \geq 1$ or empty if $k = 0$: the spiking history of all neurons up to the time point t_k .
- $\lambda_c(\tau_k | \cdot) = E\{N_c(\tau_k) | \cdot\}$: the conditional intensity function of neuron c during τ_k .
- $\exp(\gamma_{c;0})$: the baseline firing rate of neuron c .
- $\boldsymbol{\gamma}^{(c)} = (\gamma_{c;1}, \dots, \gamma_{c;P})^T$: the vector of autoregressive parameters of neuron c at different lags.
- $\boldsymbol{\gamma}^{(c,i)}$: the vector of interactive parameters representing the influence of neuron i onto neuron c at different lags. For parametrization (2.1), $\boldsymbol{\gamma}^{(c,i)} = (\gamma_{c,i;1}, \dots, \gamma_{c,i;Q})^T$ as in (2.4); for parametrization (4.1), $\boldsymbol{\gamma}^{(c,i)} = \gamma_{c,i;1}$ as in (4.2).
- $\mathbf{X}(\tau_k)$: a M dimensional covariate vector measured during τ_k .
- $\boldsymbol{\beta}_c = (\beta_{c,1}, \dots, \beta_{c,M})^T$: the parameters that correspond to the covariates for neuron c .
- P : the history windows of autoregressive parameters.
- Q : the history windows of interactive parameters.
- M : the length of covariate vector $\mathbf{X}(\tau_k)$.
- $\tilde{\boldsymbol{\theta}}_c$: the vector form of all parameters for neuron c , i.e. $\tilde{\boldsymbol{\theta}}_c = (\gamma_{c;0}, \boldsymbol{\theta}_c^T)^T$.
- $\boldsymbol{\theta}_c$: the vector form of all parameters for neuron c excluding the intercept $\gamma_{c;0}$.

- $\tilde{\boldsymbol{\theta}}_c^*$: the true unknown parameter vector.
- $d = P + (C - 1)Q + M$: the total number of covariates
- $\ell_c(\tilde{\boldsymbol{\theta}}_c)$: negative log-likelihood of neuron c .
- $\mathcal{P}_{(\eta_c, \alpha_c)}(\boldsymbol{\theta}_c)$: structured penalty of $\boldsymbol{\theta}_c$.
- $L_c(\tilde{\boldsymbol{\theta}}_c)$: negative log-likelihood of neuron c with the penalty.
- (α_c, η_c) : tuning parameters of the penalty.
- $\|\cdot\|_2$: the L_2 norm.
- $\|\cdot\|_1$: the L_1 norm.

Additional notations and symbols. Furthermore, we define by $Y_{c,k}$ and $\tilde{\mathbf{X}}_{c,k}$ the response scalar and covariate vector of (2.5) at the k th time bin, $k = 1, \dots, n$. Let $\tilde{\mathcal{X}}_c = (\tilde{\mathbf{X}}_{c,1}, \dots, \tilde{\mathbf{X}}_{c,n})^T$ denote the whole design matrix. Adopting the BD framework in Zhang *et al.* (2010), the problem of (2.5) is equivalent to the penalized BD estimation with link function $F(\cdot) = \log(\cdot)$ and

$$Q(Y, \mu) = \mu - Y \log(\mu) - Y + Y \log(Y)$$

generated by $q(\mu) = \mu - a - \mu\{\log(\mu) - \log(a)\}$ where a is a finite positive constant such that $q(\mu)$ is well-defined.

Appendix B: Proofs of Results in Section 3

Conditions. We have the following conditions in which B_1 and B_2 are sufficiently large constants. Those are not the weakest possible, but serve to facilitate the technical derivations.

- A1. The parameter space Θ of $\tilde{\boldsymbol{\theta}}_c$ is compact in \mathbb{R}^{d+1} and $\|\tilde{\boldsymbol{\theta}}_c^*\|_1 < B_1$;
- A2. For all bins, $N_c(\tau_k)$ are uniformly bounded by B_2 ;
- A3. The minimum eigenvalue of $\Sigma_c = n^{-1}\tilde{\mathcal{X}}_c\tilde{\mathcal{X}}_c^T$ are uniformly bounded away from 0, i.e. $\lambda_{\min}(\Sigma_c) > \delta$ for some positive constant $\delta > 0$;

Condition A1 restrict the magnitude of parameters from being too large or too small respectively. Condition A2 reflects the biological limitation to the number of spikes within a

short time interval. The boundedness assumption of the response variable would also facilitate the technical derivations. Condition A3 is commonly used for the covariance matrix of predictor variables and it ensures the convexity of likelihood function and identifiability of the problem.

Proof of Theorem 1. Since $d = P + (C - 1)Q + M$ and P, Q, M are all fixed constants, d and C have the same diverging rate. Let $r_n = \sqrt{d/n}$ and $\tilde{\mathbf{u}} = (u_0, u_1, \dots, u_d)^T \in \mathbb{R}^{d+1}$. Similar to Zhang *et al.* (2010), it suffices to show that for any given $\epsilon > 0$, there is a large constant U_ϵ such that, for large n ,

$$\mathbb{P}\left\{\inf_{\|\tilde{\mathbf{u}}\|_2=U_\epsilon} L_c(\tilde{\boldsymbol{\theta}}_c^* + r_n \tilde{\mathbf{u}}) > L_c(\tilde{\boldsymbol{\theta}}_c^*)\right\} \geq 1 - \epsilon. \quad (\text{A.1})$$

Now we separate the loss function and penalty term by

$$\begin{aligned} & L_c(\tilde{\boldsymbol{\theta}}_c^* + r_n \tilde{\mathbf{u}}) - L_c(\tilde{\boldsymbol{\theta}}_c^*) \\ &= \{\ell_c(\tilde{\boldsymbol{\theta}}_c^* + r_n \tilde{\mathbf{u}}) - \ell_c(\tilde{\boldsymbol{\theta}}_c^*)\} + \{\mathcal{P}_{(\eta_c, \alpha_c)}(\boldsymbol{\theta}_c^* + r_n \tilde{\mathbf{u}}) - \mathcal{P}_{(\eta_c, \alpha_c)}(\boldsymbol{\theta}_c^*)\} \\ &\equiv I_1 + I_2. \end{aligned}$$

Further by Taylor expansion of I_1 ,

$$\begin{aligned} I_1 &= \frac{r_n}{n} \sum_{k=1}^n \{\exp(\tilde{\mathbf{X}}_{c,k}^T \tilde{\boldsymbol{\theta}}_c^*) - Y_{c,k}\} \tilde{\mathbf{X}}_{c,k}^T \tilde{\mathbf{u}} + \frac{r_n^2}{2n} \sum_{k=1}^n \exp(\tilde{\mathbf{X}}_{c,k}^T \tilde{\boldsymbol{\theta}}_c^*) (\tilde{\mathbf{X}}_{c,k}^T \tilde{\mathbf{u}})^2 \\ &\quad + \frac{r_n^3}{6n} \sum_{k=1}^n \exp(\tilde{\mathbf{X}}_{c,k}^T \tilde{\boldsymbol{\theta}}_c^*) (\tilde{\mathbf{X}}_{c,k}^T \tilde{\mathbf{u}})^3 = I_{1,1} + I_{1,2} + I_{1,3} \end{aligned}$$

where $\tilde{\boldsymbol{\theta}}_c'$ is located between $\tilde{\boldsymbol{\theta}}_c^*$ and $\tilde{\boldsymbol{\theta}}_c^* + r_n \tilde{\mathbf{u}}$. Then those terms can be bounded as

$$|I_{1,1}| \leq \left\| \frac{r_n}{n} \sum_{k=1}^n \{\exp(\tilde{\mathbf{X}}_{c,k}^T \tilde{\boldsymbol{\theta}}_c^*) - Y_{c,k}\} \tilde{\mathbf{X}}_{c,k} \right\|_2 \|\tilde{\mathbf{u}}\|_2 = O_{\mathbb{P}}(r_n \sqrt{d/n}) U_\epsilon; \quad (\text{A.2})$$

$$I_{1,2} = 2^{-1} r_n^2 \tilde{\mathbf{u}}^T (\mathcal{X}_c^T \Omega \mathcal{X}_c / n) \tilde{\mathbf{u}} \geq c_1 r_n^2 U_\epsilon^2; \quad (\text{A.3})$$

$$|I_{1,3}| \leq O_{\mathbb{P}}(r_n^3 d^{3/2}) U_\epsilon^3, \quad (\text{A.4})$$

where $\Omega = \text{diag}\{\exp(\tilde{\mathbf{X}}_{c,1}^T \tilde{\boldsymbol{\theta}}_c^*), \dots, \exp(\tilde{\mathbf{X}}_{c,n}^T \tilde{\boldsymbol{\theta}}_c^*)\}$ and c_1 is some positive constant. Here the data are not i.i.d., so we need to apply the martingale version of central limit theorem (Theorem 7.4 in Durrett, 2004) to obtain the rate of $I_{1,1}$ in (A.2), in which each element of $\sum_{k=1}^n \{\exp(\tilde{\mathbf{X}}_{c,k}^T \tilde{\boldsymbol{\theta}}_c^*) - Y_{c,k}\} \tilde{\mathbf{X}}_{c,k}^T$ is a martingale with bounded increments. (A.3) and (A.4) follow conditions A1-A3.

For the penalty term,

$$\begin{aligned}
I_2 \geq & -(1 - \alpha_c)\eta_c \left\{ \sqrt{P} \|r_n \mathbf{u}_{\gamma^{(c)}}\|_2 + \sqrt{Q} \sum_{i \in \{1, \dots, C\} \setminus c} \|r_n \mathbf{u}_{\gamma^{(c,i)}}\|_2 \right\} \\
& - \alpha_c \eta_c \left\{ \|r_n \mathbf{u}_{\gamma^{(c)}}\|_1 + \sum_{i \in \{1, \dots, C\} \setminus c} \|r_n \mathbf{u}_{\gamma^{(c,i)}}\|_1 \right\} - \eta_c \|r_n \mathbf{u}_{\beta_c}\|_1
\end{aligned}$$

where $\mathbf{u}_{\gamma^{(c)}}$, $\mathbf{u}_{\gamma^{(c,i)}}$ and \mathbf{u}_{β_c} are components of $\tilde{\mathbf{u}}$ corresponding to $\gamma^{(c)}$, $\gamma^{(c,i)}$ and \mathbf{u}_{β_c} respectively. Then,

$$\begin{aligned}
|I_2| & \leq \eta_c r_n \left\{ \sqrt{P} \|\mathbf{u}_{\gamma^{(c)}}\|_2 + \sqrt{Q} \sum_{i \in \{1, \dots, C\} \setminus c} \|\mathbf{u}_{\gamma^{(c,i)}}\|_2 + \sqrt{M} \|\mathbf{u}_{\beta_c}\|_2 \right\} \\
& \leq \eta_c r_n \sqrt{\max(P, Q, M)} \sqrt{C+1} \|\tilde{\mathbf{u}}\|_2 = O(\eta_c r_n \sqrt{d}) U_\epsilon.
\end{aligned}$$

Since $d^4/n = O(C^4/n) = o(1)$, we can choose some large C_ϵ such that $I_{1,1}$, $I_{1,3}$ and I_2 are all dominated by $I_{1,2}$, which is positive. This implies (A.1). ■

Appendix C: Figures and Tables in the paper

Section 4.1. Figures 1–2; Tables 1–2.

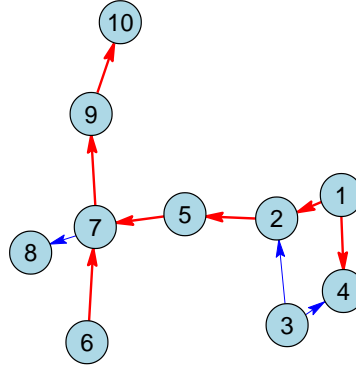


Figure 1: **(Simulation of simple network)** A simulated network with 10 neurons. The line types of arrows indicate two different types of interactive parameters in Figure 2, where the thick Red line represents type-A parameter (more excitatory) and the thin Blue line represents type-B parameter (more inhibitory).

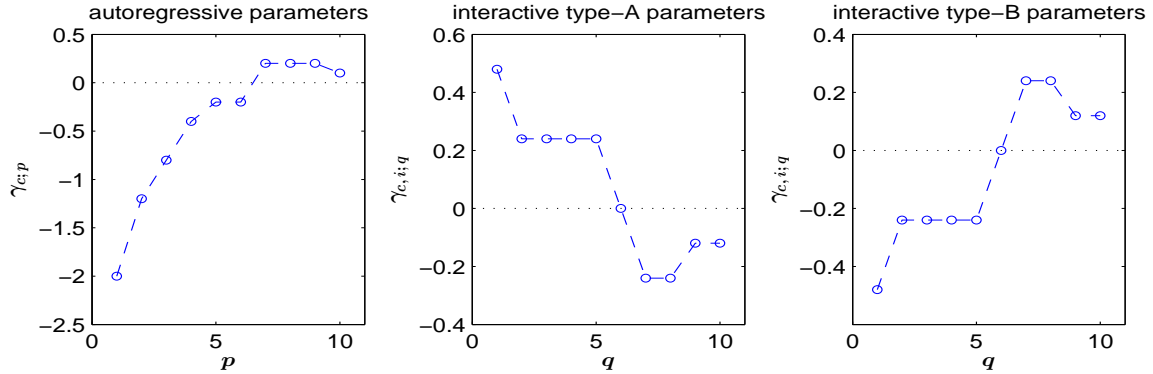


Figure 2: **(Simulation of simple network)** True values (using \circ connected with dashed lines) of parameters for the simulation. The horizontal dotted line indicates 0. Left panel: autoregressive parameters $\{\gamma_{c;p}\}_{p=1}^P$; middle panel: interactive type-A parameters $\{\gamma_{c,i;q}\}_{q=1}^Q$; right panel: interactive type-B parameters $\{\gamma_{c,i;q}\}_{q=1}^Q$.

Table 1: **(Simulation of simple network)** Comparison of estimated network by regularized methods with varying strength of parameters and length of spike train. Results are averaged over 100 replications, with standard deviations indicated in parentheses.

Length (bins)	Relative Strength	Method	Correct-All	detected-A	detected-B	Correct-NC
10000	100%	L_1 -short-10-3	1.58 (1.4)	1.34 (1.1)	0.24 (0.5)	79.11 (1.2)
		L_1 -10-10	0.61 (0.8)	0.59 (0.8)	0.02 (0.1)	79.72 (0.7)
		TR-5-5	3.01 (1.4)	2.47 (1.2)	0.54 (0.6)	76.72 (2.1)
		TR-10-10	1.73 (1.1)	1.54 (1.1)	0.19 (0.4)	77.90 (1.5)
		SGL-5-5	4.98 (1.6)	4.21 (1.4)	0.77 (0.7)	75.64 (3.1)
		SGL-10-10	4.73 (1.9)	3.57 (1.5)	1.16 (0.8)	67.02 (8.2)
		SGL-15-15	4.62 (1.8)	3.49 (1.4)	1.13 (0.8)	58.95 (9.7)
		True network	10.00 (0.0)	7.00 (0.0)	3.00 (0.0)	80.00 (0.0)
	125%	L_1 -short-10-3	4.02 (1.6)	3.40 (1.3)	0.62 (0.7)	78.23 (1.7)
		L_1 -10-10	1.70 (1.4)	1.66 (1.3)	0.04 (0.2)	79.68 (0.5)
		TR-5-5	4.46 (1.4)	3.79 (1.2)	0.67 (0.8)	77.30 (1.8)
		TR-10-10	2.78 (1.5)	2.58 (1.3)	0.20 (0.4)	78.00 (1.7)
		SGL-5-5	7.54 (1.4)	6.13 (0.9)	1.41 (1.0)	73.43 (4.0)
		SGL-10-10	7.54 (1.5)	5.69 (1.1)	1.85 (0.7)	62.93 (8.2)
		SGL-15-15	7.23 (1.6)	5.39 (1.1)	1.84 (0.8)	50.74 (11.1)
		True network	10.00 (0.0)	7.00 (0.0)	3.00 (0.0)	80.00 (0.0)

Table 2: *The caption is identical to that for Table 1, except that 15000 bins are used.*

Length (bins)	Relative Strength	Method	Correct-All	detected-A	detected-B	Correct-NC
15000	100%	L_1 -short-10-3	6.27 (1.7)	4.86 (1.3)	1.41 (0.8)	74.42 (3.3)
		L_1 -10-10	2.70 (1.7)	2.42 (1.4)	0.28 (0.5)	78.84 (1.2)
		TR-5-5	3.72 (1.6)	3.22 (1.4)	0.50 (0.6)	77.31 (2.2)
		TR-10-10	2.37 (1.2)	2.08 (1.1)	0.29 (0.5)	78.59 (1.2)
		SGL-5-5	7.33 (1.5)	5.70 (1.0)	1.63 (0.9)	74.62 (3.2)
		SGL-10-10	7.72 (1.6)	5.73 (1.1)	1.99 (0.9)	64.49 (7.4)
		SGL-15-15	7.42 (1.6)	5.46 (1.1)	1.96 (0.8)	51.80 (9.4)
		True network	10.00 (0.0)	7.00 (0.0)	3.00 (0.0)	80.00 (0.0)
	125%	L_1 -short-10-3	8.26 (1.4)	6.25 (0.8)	2.01 (0.9)	73.61 (2.9)
		L_1 -10-10	4.89 (1.7)	4.37 (1.4)	0.52 (0.6)	78.27 (1.5)
		TR-5-5	6.05 (1.7)	4.93 (1.4)	1.12 (0.8)	77.22 (2.0)
		TR-10-10	4.10 (1.4)	3.70 (1.3)	0.40 (0.6)	78.37 (1.6)
		SGL-5-5	9.13 (0.9)	6.82 (0.4)	2.31 (0.8)	72.26 (4.0)
		SGL-10-10	9.39 (0.9)	6.79 (0.4)	2.60 (0.6)	61.96 (8.5)
		SGL-15-15	9.24 (0.8)	6.65 (0.6)	2.59 (0.5)	47.15 (10.2)
		True network	10.00 (0.0)	7.00 (0.0)	3.00 (0.0)	80.00 (0.0)

Section 4.2. Figures 3, 4, 5 and 6; Table 3.

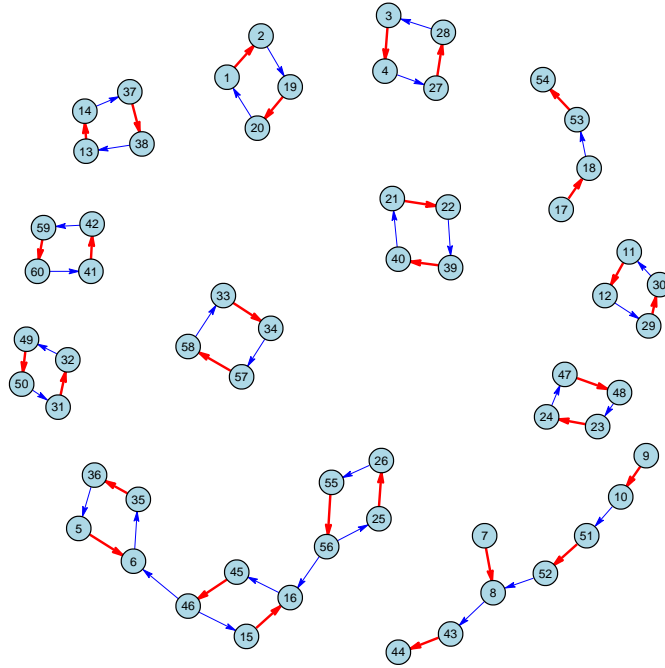


Figure 3: **(Simulation of complex network)** A simulated network with 60 neurons. The line types of arrows indicate two different types of interactive parameters in Figure 2, where the thick Red line represents type-A parameter and the thin Blue line represents type-B parameter.

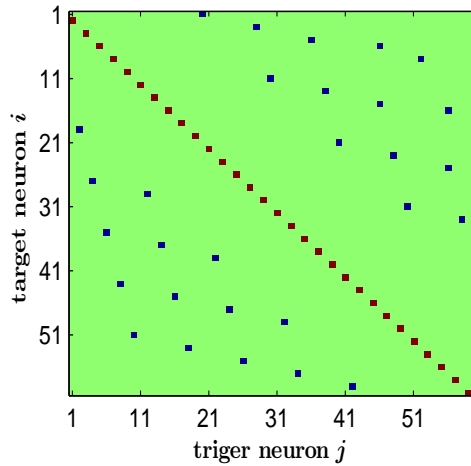


Figure 4: **(Simulation of complex network)** The connectivity matrix of the simulated network. The color of the cell at the i th row and the j th column distinguishes how neuron j 's spiking history would affect the firing rate of neuron i . The background Green color means no connection; the Red color represents type-A parameter and the Blue color represents type-B parameter.

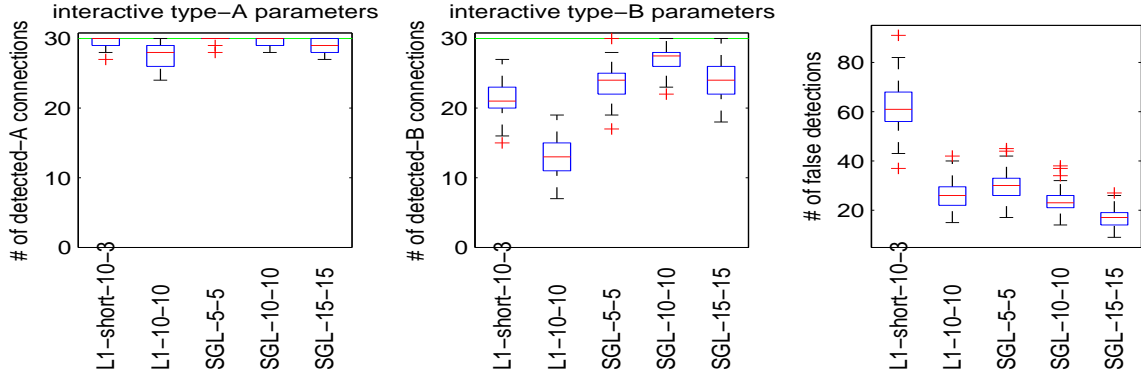


Figure 5: **(Simulation of complex network)** Boxplots comparing 5 methods. Left panel: number of “detected-A” connections; middle panel: number of “detected-B” connections; right panel: number of false detections. Here $n = 25000$ and the relative strength is 125%.

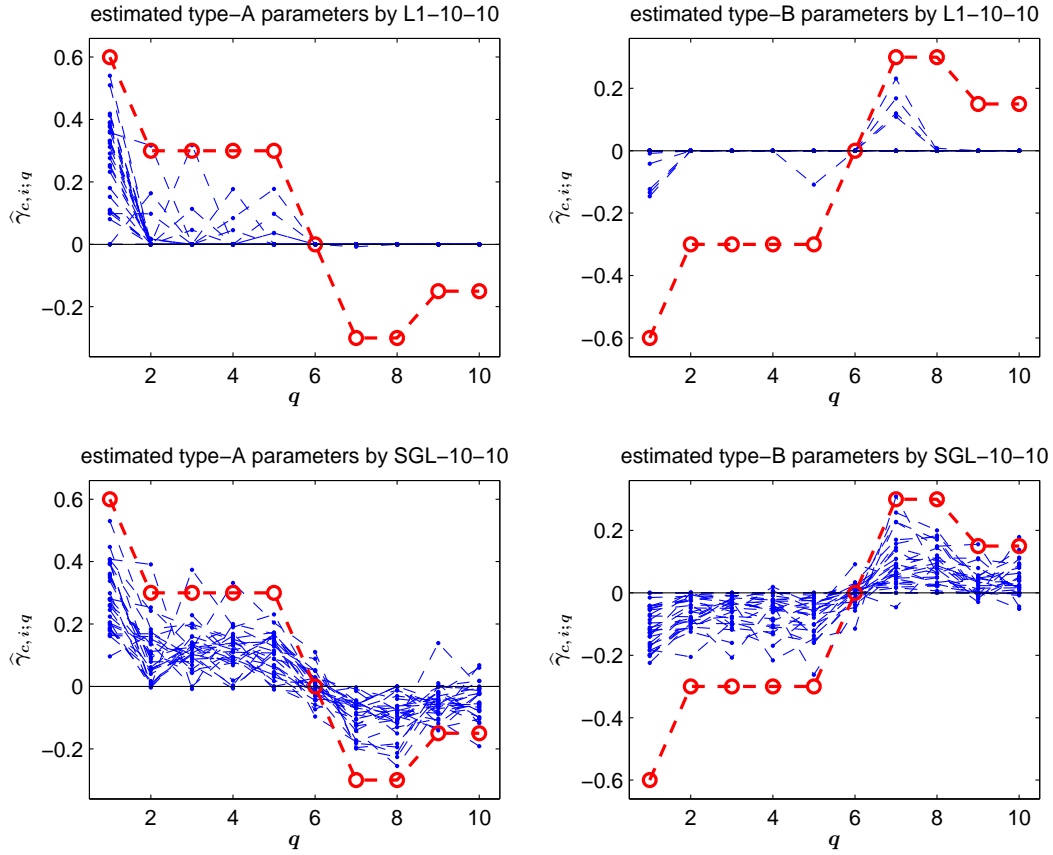


Figure 6: **(Simulation of complex network)** The estimated parameters of all type-A and type-B connections in a single simulation run. Left panels: plots of type-A parameters; right panels: plots of type-B parameters. Top panels: plots of the estimates obtained by L_1 -10-10; bottom panels: plots of the estimates obtained by SGL-10-10. Thick \circ connected with dashed lines: true values; thin dashed lines: estimated values. Here $n = 25000$ and the relative strength is 125%.

Table 3: **(Simulation of complex network)** Comparison of estimated network by regularized methods with varying strength of parameters and length of spike train. Results are averaged over 100 replications, with standard deviations indicated in parentheses.

Length (bins)	Relative Strength	Method	Correct-All	detected-A	detected-B	Correct-NC
20000	100%	L_1 -short-10-3	23.65 (3.4)	17.19 (2.7)	6.46 (2.0)	3457.07 (7.7)
		L_1 -10-10	15.94 (3.2)	13.04 (2.7)	2.90 (1.6)	3464.85 (5.0)
		SGL-5-5	30.42 (3.2)	21.84 (2.3)	8.58 (2.5)	3460.30 (4.7)
		SGL-10-10	31.78 (3.9)	19.57 (2.5)	12.21 (3.0)	3466.68 (3.5)
		SGL-15-15	21.97 (3.7)	14.78 (2.6)	7.19 (2.5)	3473.01 (2.7)
		True network	60.00 (0.0)	30.00 (0.0)	30.00 (0.0)	3480.00 (0.0)
	125%	L_1 -short-10-3	35.23 (3.5)	24.77 (2.0)	10.46 (2.5)	3450.98 (7.8)
		L_1 -10-10	26.86 (3.5)	21.60 (2.6)	5.26 (2.0)	3463.54 (4.4)
		SGL-5-5	46.76 (3.0)	28.83 (1.1)	17.93 (2.7)	3454.38 (5.3)
		SGL-10-10	50.63 (3.0)	28.15 (1.3)	22.48 (2.6)	3459.48 (4.7)
		SGL-15-15	42.37 (3.4)	25.73 (1.6)	16.64 (2.8)	3466.64 (3.7)
		True network	60.00 (0.0)	30.00 (0.0)	30.00 (0.0)	3480.00 (0.0)
25000	100%	L_1 -short-10-3	41.70 (3.2)	25.92 (1.8)	15.78 (2.6)	3426.09 (9.9)
		L_1 -10-10	27.75 (3.6)	19.77 (2.5)	7.98 (2.3)	3454.48 (6.4)
		SGL-5-5	39.92 (3.0)	25.83 (1.9)	14.09 (2.6)	3456.61 (5.0)
		SGL-10-10	42.18 (3.8)	24.27 (2.3)	17.91 (2.9)	3463.31 (3.8)
		SGL-15-15	33.19 (3.7)	20.60 (2.8)	12.59 (2.5)	3469.86 (3.3)
		True network	60.00 (0.0)	30.00 (0.0)	30.00 (0.0)	3480.00 (0.0)
	125%	L_1 -short-10-3	50.69 (2.5)	29.35 (0.8)	21.34 (2.4)	3418.31 (9.5)
		L_1 -10-10	40.47 (3.3)	27.41 (1.6)	13.06 (2.7)	3453.88 (6.1)
		SGL-5-5	53.59 (2.4)	29.79 (0.5)	23.80 (2.3)	3450.05 (6.0)
		SGL-10-10	56.97 (1.8)	29.69 (0.6)	27.28 (1.5)	3456.29 (4.9)
		SGL-15-15	52.68 (2.4)	28.90 (0.9)	23.78 (2.3)	3463.03 (3.5)
		True network	60.00 (0.0)	30.00 (0.0)	30.00 (0.0)	3480.00 (0.0)

Section 4.3. Table 4.

Table 4: **(Simulation of real network)** Comparison of estimated network by regularized methods for the simulated network that mimics the real data, using $P = 30$ and $Q = 10$ in (2.1). Results are averaged over 100 replications, with standard deviations indicated in parentheses.

Method	Exper. 1 (with 73 neurons)		Exper. 4 (with 77 neurons)	
	Correct-All	Correct-NC	Correct-All	Correct-NC
L_1 -short-30-3	47.62 (1.5)	5048.70 (7.8)	53.95 (1.4)	5681.65 (7.9)
L_1 -30-10	45.47 (1.4)	5045.85 (7.7)	50.55 (1.4)	5689.68 (6.3)
SGL-30-10	48.44 (1.3)	5164.92 (5.5)	51.21 (1.0)	5745.54 (4.7)
True network	62.00 (0.0)	5194.00 (0.0)	71.00 (0.0)	5781.00 (0.0)

Section 5. Figures 7, 8, 9 and 10; Tables 5, 6, 7 and 8.

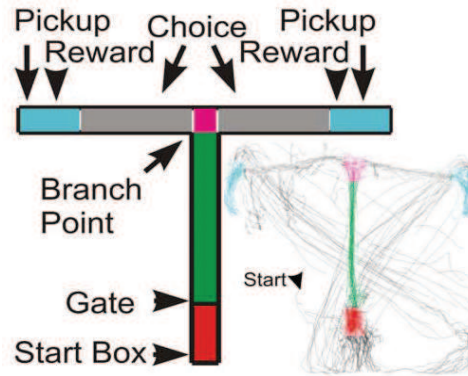


Figure 7: **(Neurophysiological data)** T-maze task: for each trial, the animal was placed in a start-box for a pre-determined time (a delay) and released with the removal of a starting gate. The rat would move to the junction of the “T” and choose the left or right arm. On a correct choice, the animal was rewarded and returned in the start-box. This figure is adapted from Devilbiss et al. (2012).

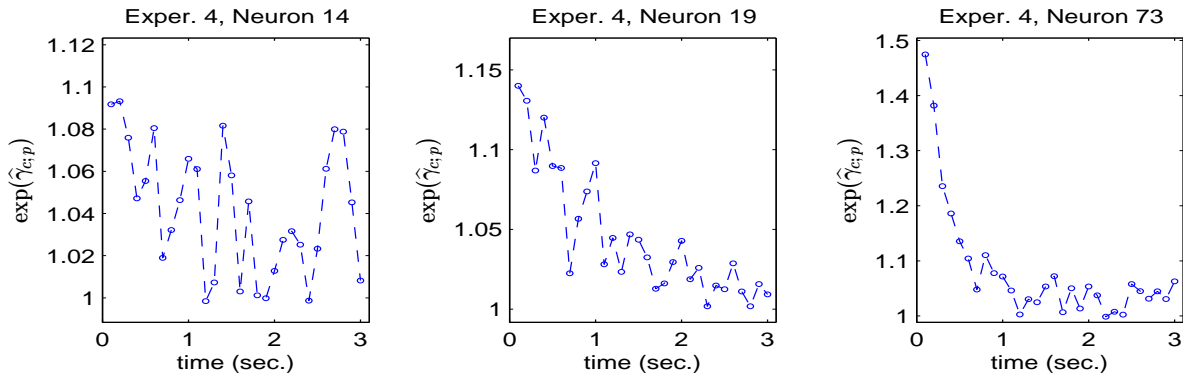


Figure 8: **(Neurophysiological data)** Estimates (using \circ connected with dashed lines) of gain, i.e. $\exp(\gamma_{c;p})$, associated with autoregressive parameters by the SGL regularized method for selected neurons in Experiment 4.

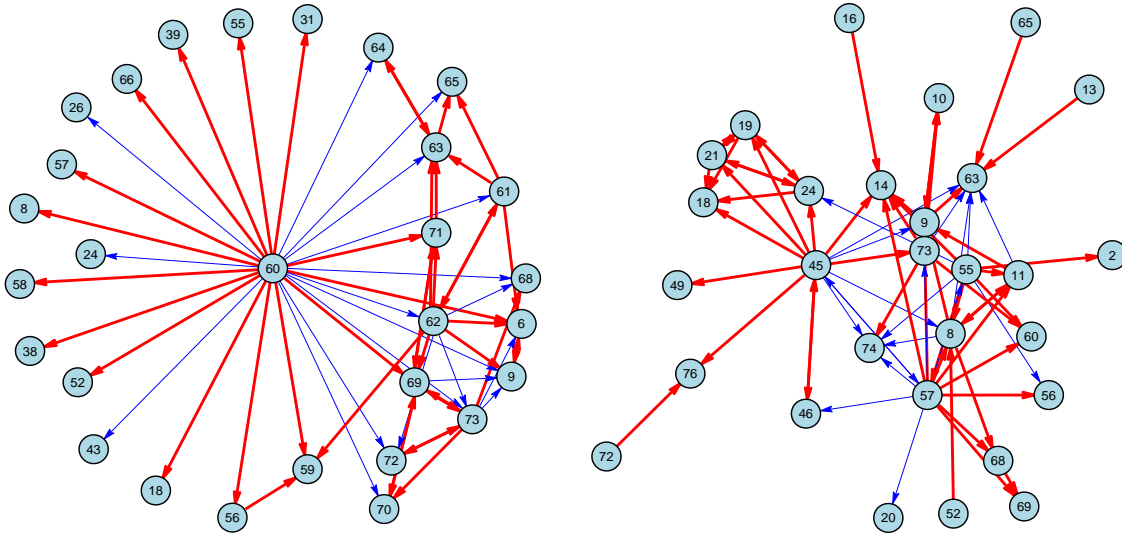


Figure 9: **(Neurophysiological data)** The estimated network structures by the SGL regularized method. The thick Red color presents the excitatory connection and thin Blue color presents the inhibitory connection. Left panel: Experiment 1; right panel: Experiment 4.

Table 5: **(Neurophysiological data)** Summary statistics of the T-maze task dataset.

No.	Subject	Total length (sec)	Total number of neurons			Average firing rate (Hz)
			WS-type	NS-type	Total	
1	TM19	1449.2	53	4	73	0.567
2	TM19	1150.8	60	5	83	0.408
3	TM44	2243.7	44	6	59	0.772
4	TM44	2150.3	62	3	77	0.364
5	TM44	2191.1	57	3	71	0.192
6	TM44	2011.4	53	3	63	0.241
7	TM57	930.3	30	8	39	0.375
8	TM58	765.2	23	0	26	0.342

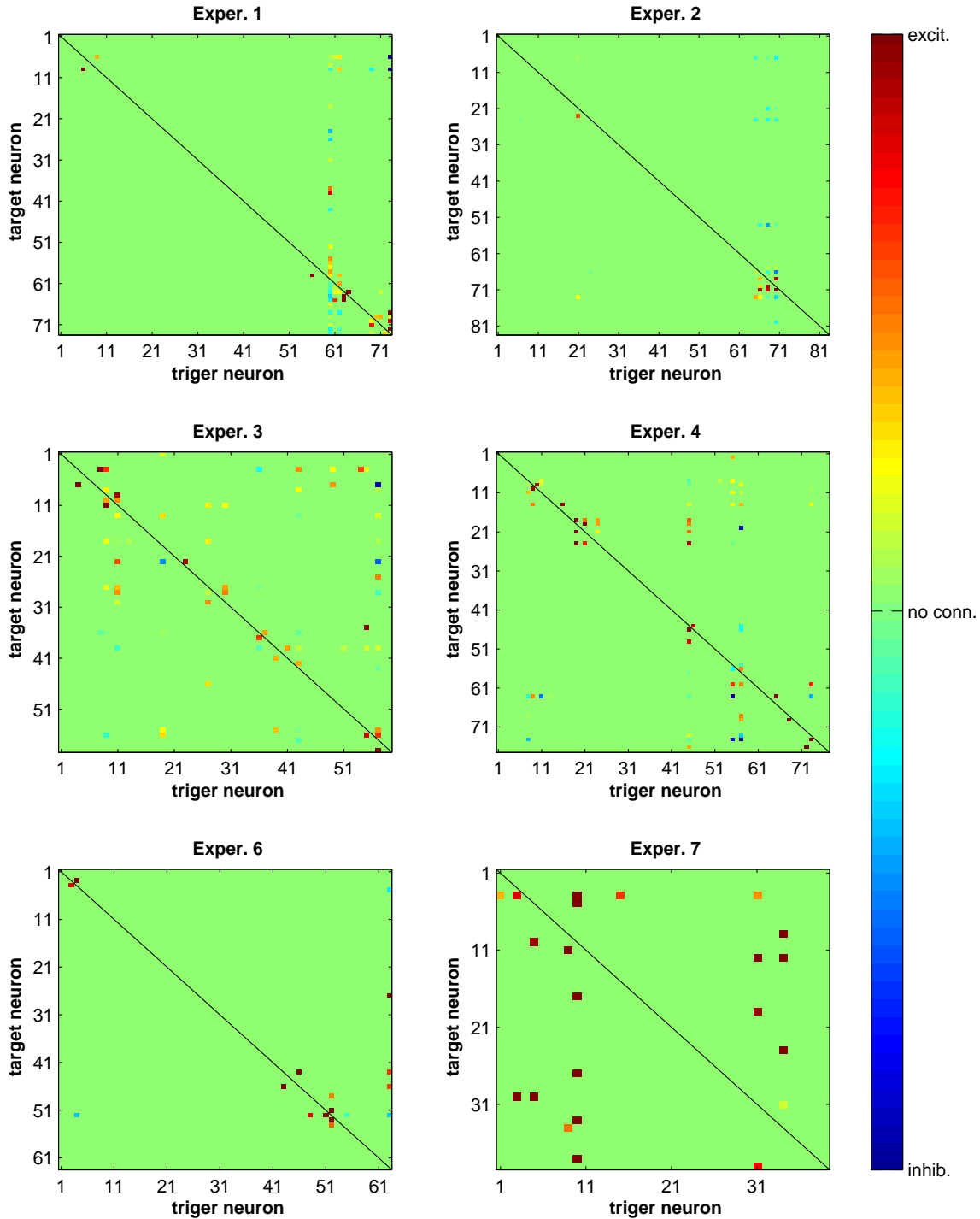


Figure 10: (Neurophysiological data) The estimated network matrices by the SGL regularized method. The Red color indicates the excitatory connection between the pair and the Blue color indicates the inhibitory connection, while the Green color means no functional connection. The exact color is determined by the strength of the connection. There is no estimate of interactive parameters for the blocks on the black diagonal line.

Table 6: **(Neurophysiological data)** *Description of the covariates used in the analysis.*

Column #	Name	Description
1	Delay	Stay in the resting area until the gate is released.
2	Run	Run through the tunnel from the gate to the branch point.
3	Decision	At the branch area of the T-maze and make a turn.
4	Choice	Run to the left or right end.
5	Reward	Eat reward in the correct trial.
6	Pick-up	Be picked up and put back to the resting area.
7	Left-trial	Whether made the left turn in the trial.
8	Correct-trial	Whether made the correct choice in the trial. (different from the previous one.)

Table 7: **(Neurophysiological data)** *Summary of estimated networks by regularized methods.*

Experiment	Possible pairs	L_1 -short-30-3		L_1 -30-10		SGL-30-10	
		Excit.	Inhib.	Excit.	Inhib.	Excit.	Inhib.
1	5256	48	40	17	14	43	19
2	6806	20	24	14	9	15	18
3	3422	116	53	49	13	63	16
4	5852	99	54	21	12	52	19
5	4970	21	9	3	0	9	4
6	3906	28	7	16	2	14	4
7	1482	55	6	13	2	23	0
8	650	0	0	0	0	0	0

Table 8: **(Neurophysiological data)** *Summary of estimated parameters by the SGL regularized method for 8 state indicator covariates described in Table 6.*

Experiment	Number of neurons with positive and negative estimated parameters															
	Delay		Run		Decision		Choice		Reward		Pick-up		Left		Correct	
	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-
1	6	5	2	0	11	3	0	0	0	0	3	4	4	3	2	6
2	9	7	6	1	1	1	2	0	2	1	3	2	0	2	0	1
3	8	29	6	1	6	3	2	0	0	1	7	0	2	8	1	3
4	8	26	7	0	6	0	7	0	0	0	5	7	8	3	6	3
5	5	13	2	1	1	0	2	0	0	2	6	1	1	3	1	3
6	11	14	2	0	2	0	2	0	4	1	3	4	1	4	0	1
7	10	7	16	1	6	0	1	1	1	6	1	10	4	6	0	6
8	0	0	0	0	0	0	0	0	1	0	4	0	1	0	1	0