An L_1-regularized logistic model for detecting short-term neuronal interactions

**Mengyuan Zhao · Aaron Batista · John P. Cunningham ·
Cynthia Chestek · Zuley Rivera-Alvidrez · Rachel Kalmar ·
Stephen Ryu · Krishna Shenoy · Satish Iyengar**

Abstract Interactions among neurons are a key component of neural signal processing. Rich neural data sets potentially containing evidence of interactions can now be collected readily in the laboratory, but existing analysis methods are often not sufficiently sensitive and specific to reveal these interactions. Generalized linear models offer a platform for analyzing multi-electrode recordings of neuronal spike train data. Here we suggest an L_1-regularized logistic regression model (L_1L method) to detect short-term (order of 3 ms) neuronal interactions. We estimate the parameters in this model using a coordinate descent algorithm, and determine the optimal tuning parameter using a Bayesian Information Criterion. Simulation studies show that in general the L_1L method has better sensitivities and specificities than those of the traditional shuffle-corrected cross-correlogram (covariogram) method. The L_1L method is able to detect excitatory interactions with both high sensitivity and specificity with reasonably large recordings, even when the magnitude of the interactions is small; similar results hold for inhibition given sufficiently high baseline firing rates. Our study also suggests that the false positives can be further removed by thresholding, because their magnitudes are typically smaller than true interactions. Simulations also show that the L_1L method is somewhat robust to partially observed networks. We apply the method to multi-electrode recordings collected in the monkey dorsal premotor cortex (PMd) while the animal prepares to make reaching arm movements. The results show that some neurons interact differently depending on task conditions. The stronger

M. Zhao (✉) · S. Iyengar
Department of Statistics, University of Pittsburgh,
Pittsburgh, PA 15260, USA
e-mail: mez25@pitt.edu

S. Iyengar
e-mail: ssi@pitt.edu

A. Batista
Department of Bioengineering, University of Pittsburgh,
Pittsburgh, PA 15260, USA

J. P. Cunningham · C. Chestek · Z. Rivera-Alvidrez ·
S. Ryu · K. Shenoy
Department of Electrical Engineering,
Stanford University, Stanford,
CA 94305-9505, USA

J. P. Cunningham
Department of Engineering, Cambridge University,
Cambridge, CB2 1PZ, UK

R. Kalmar
Neurosciences Program, Stanford University,
Stanford, CA 94305-9505, USA

S. Ryu
Department of Neurosurgery, Palo Alto Medical
Foundation, Palo Alto, CA 94301, USA

K. Shenoy
Department of Bioengineering, Stanford University,
Stanford, CA 94305-9505, USA

K. Shenoy
Department of Neurobiology, Stanford University,
Stanford, CA 94305-9505, USA

interactions detected with our $L_1 L$ method were also visible using the covariogram method.

1 Introduction

An important problem in current neuroscience research is to understand the networks of interactions among the neurons in relevant brain areas (Brown et al. 2004; Fujisawa et al. 2008; Eldawlatly et al. 2009; Stevenson et al. 2009; Kass et al. 2011; Mishchencko et al. 2011). New multi-electrode recording techniques allow simultaneous recordings of neurons during behavior, which can help reveal how cells interact. At the same time, these recordings present a great challenge to data analysts, because conventional procedures are often inadequate to infer structure in the noisy high dimensional data yielded by these experiments.

The commonly used tools by neuroscientists to study neuronal correlations are the cross-correlation histogram (Perkel et al. 1967) and its variants, including the joint peri-stimulus time histogram (JPSTH) (Gerstein and Perkel 1972), the snowflake plot (Perkel et al. 1975; Czanner et al. 2005), and the shuffle-corrected cross-correlogram (Aertsen et al. 1989; Brody 1999). However, these methods have limitations. First, they study only two or three neurons at a time, ignoring the possible contributions of other neurons. The detected correlations could be ambiguous due to the possible but excluded influence of other neurons, such as common-input or chain effects. Second, the correlations detected by these graphical methods are undirected, that is, the correlation between neurons A and B is the same as that between neurons B and A. However, because neurons are polarized, such an analysis may not capture the true asymmetric information flow in the brain. Third, those graphical methods are histogram-based, so analysis must be restricted to a specific time scale or degrees of smoothing (Kohn and Smith 2005; Harrison and Geman 2009).

Brillinger (1988) introduced generalized linear models (GLMs) for the analysis of the firing rate of a neuron as a function of the time since its last spike and the spiking history of other neurons. Although he studied small networks (three neurons) GLMs offer a useful framework for the analysis of tens, even hundreds of simultaneously recorded neurons. Since then, much of the work in this area has focused on encoding, which fits a model of neural spiking given observed behavior (Paninski 2004; Truccolo et al. 2005; Kulkarni and Paninski 2007). GLMs can be generalized to point process (Paninski 2004; Kass et al. 2011) or state-space frameworks (Kulkarni and Paninski 2007; Mishchencko et al. 2011). In addition, GLMs have proven to be successful in decoding body movements from neural activity (Gao et al. 2003; Truccolo et al. 2005), and are better than entropy methods in predicting the spikes of single neurons (Truccolo et al. 2010).

One additional advantage of the GLM approach is that it allows all recorded neurons and interactions among them to be analyzed simultaneously. GLMs can capture other influences on a neuron, such as spike history of the neural ensemble, environmental events, and body movement. The resulting GLM parameters have an intuitive interpretation as indicating the degree of interactions among neurons. Unlike traditional graphical methods, this measurement of neuronal interactions is directed, flexible to the choice of time lags, and conditional on the neuron ensemble. Therefore, in this paper we call this type of neuronal spike-timing dependence described by the GLM an 'interaction', to distinguish it from the 'correlation' estimated from graphical methods like the shuffle-corrected cross-correlogram.

Here we use GLMs to assess neuronal interactions and their variations under different behavioral tasks. We interpret the signs of parameters in GLMs as indicating an excitatory (positive), inhibitory (negative), or a lack of (zero) interaction between neurons. Neuronal interactions can reasonably be expected to be sparse. A cortical neuron receives between 10^3–10^4 inputs, which is a vanishingly small fraction of the 10^{10} neurons in the cerebral cortex. Also, the occurrence of correlations reported using conventional methods range from small (Kohn and Smith 2005) to essentially zero (Ecker et al. 2010). One such attempt by Truccolo et al. (2005) uses the Akaike Information Criterion (AIC) to select models. However, the AIC cannot automatically select the best among all possible interaction networks because it must compare all candidate models, which is infeasible for a large neuron population. Therefore, unless we have postulated a network for testing *a priori*, an automatic model selection approach is required to find the neural interactions. In addition, a naive implementation of logistic or Poisson models for spike train data suffers from nonconvergence problems. Zhao and Iyengar (2010) showed that these difficulties are due to infinite values of the maximum likelihood estimates (MLEs). Therefore, standard stepwise variable selection methods are susceptible to the nonconvergence problems.

The model selection method we consider, which we call the L_1L method, uses an L_1 regularization on a logistic regression model to detect short-term neuronal interactions. We chose L_1 (as opposed to, say, L_2) regularization because several studies Tibshirani (1996) and Avalos et al. (2003) show that it performs well for models that are sparse, with relatively few regression coefficients that are large. There are several other recent works in neuroscience that also use L_1 regularization. Stevenson et al. (2009) used a Bayesian formulation of L_1 regularization to detect long-term (~100 ms) neuronal interactions. Kelly et al. (2010) used a L_1-regularized Poisson model to study neuronal network in V1 data. Chen et al. (2010) compared L1, L2 and hierarchical variational Bayesian methods with respect to their performance in detecting interactions. Mishchencko et al. (2011) proposed a L1-regularized state-space model to detect functional connectiviy on calcium fluorescent imaging data. In this paper, we extend the study of L_1-regularization in several ways. First, the performance of L_1 regularization depends highly on the details of network, that is, the size of the model, the sparsity of the model, the correlation between variables, and the method used to select tuning parameters (Hastie et al. 2001). Therefore, we simulated networks, varying several factors that could potentially influence the method's performance. In particular, we study its sensitivity (ability to detect nonzero coefficients) and specificity (resistance to false positives) while we vary: (1) amount of data, (2) network complexity, (3) magnitudes of connection strength, (4) excitation and inhibition, and (5) the fraction of the complete network observed from our simulation. We assess the performance of the L_1L method across these conditions. Second, different brain areas have different structures or different degrees of sparsity of neuronal interactions. The complexity of the network will have different effects on short-term and long-term interactions. Therefore, it is worth assessing the performance of the L_1L method when applied to the motor cortex to study short-term interactions. Third, we use the Bayesian Information Criterion (BIC) to select the tuning parameter (BIC γ-selector). We state and prove a theorem that using BIC γ-selector on the L_1L method asymptotically maintains all true variables in the selected submodel, so that the problem of false positives will dominate that of false negatives. This result sheds light on the theoretical performance of the L_1L method. Fourth, to further remove false positives, we suggest a further thresholding on selected interactions. Although, this second thresholding step is not a main focus of our

paper, we illustrate its use on both simulated and real data, and discuss our future work in this direction. Fifth, we also use the simulation to compare our method with the shuffle-corrected cross-correlogram (covariogram) (Brody 1999). Our proposed L_1L method generally performs better than the covariogram method with respect to the sensitivity and specificity of interaction detection. In particular when the interaction network is complex, the L_1L method has far fewer false positives than the covariogram method. In addition, because the optimization of L_1-regularized problems is usually computationally heavy, we use the 'coordinate descent algorithm' as implemented in the R package **glmnet** (Friedman et al. 2007, 2010; Kelly et al. 2010). Finally, the use of the BIC γ-selector instead of traditional k-fold cross validation can avoid extra model fittings, making our approach feasible for routine use.

After calibrating the L_1L method by simulations, we apply it to electrophysiological data collected from the dorsal premotor cortex (PMd) of a behaving Rhesus monkey. The animal was instructed to perform reaching movements to targets distributed through a workspace. Each movement was preceded by an instructed delay period during which the animal was aware of the reach he would soon be instructed to perform, but was not yet permitted to do so (Batista et al. 2007). This 'plan' period generated relatively stationary average firing rates, which can remove confounds in a correlation analysis (Brody 1999). We found that both the quantity and pattern of short-term neuronal interactions change with different conditions, which is further confirmed by a covariogram analysis. The L_1L method detected about only half number of short-term neuronal interactions that the covariogram method did. Based on our simulation studies, we believe that, given a potentially complicated neuronal network and strong interactions between neurons, the L_1L method simultaneously maintains high sensitivity and specificity, while the covariogram method is susceptible to more false positives.

The paper is organized as follows. Section 2 describes the GLM framework for modeling multi-electrode recording data, and our L_1L method to detect short-term neuronal interactions. In Section 3, simulation studies are used to assess the performance of the L_1L method and to compare the L_1L method with the covariogram method. The performance is assessed with respect to the aspects stated above. In Section 4, the L_1L method is applied to monkey PMd data. Section 5 summarizes the main results and discusses future extensions and applications.

2 Methods

2.1 GLM framework for multi-electrode recording data

A typical format for neurophysiological data is the binned spike count: a discrete integer-valued time series with each value indicating the number of action potential discharges (spikes) within each time bin. There is one such 'spike train' per neuron recorded. Depending on the type of experiment, the time courses of extrinsic covariate information, such as stimulus history and body kinematics, can also accompany the spike trains. These data are discretized to the same time bins as the spike data.

We bin the time axis into T equal segments. Typically T is large enough so that at most one spike per neuron occurs within each bin of size Δ. Thus the binned spike count time series has binary values typically with $\Delta = 1$ millisecond (ms) (Brillinger 1988; Truccolo et al. 2005). Large bin sizes that lead to count data are also used (Stevenson et al. 2009). We denote the spike train within the first t bins of neuron c as $N_{1:t}^c$, the number of spikes within the tth bin of neuron c as ΔN_t^c, the history of all neurons and extrinsic influences before, but not including, the tth bin as H_t, and its conditional firing rate (number of spikes per second) at bin t as λ_t^c, where $c = 1, 2, ..., C$, the number of neurons identified by the electrodes.

Assuming that the firing rate is constant in the time interval Δ, the distribution of ΔN_t^c conditioned on the history is typically considered as either Bernoulli if ΔN_t^c is binary, or Poisson if ΔN_t^c is a count. In the Bernoulli case,

$$P\left(\Delta N_t^c | H_t\right) = \left[\lambda_t^c \Delta\right]^{\Delta N_t^c} \left[1 - \lambda_t^c \Delta\right]^{1 - \Delta N_t^c},$$

and in the Poisson case,

$$P\left(\Delta N_t^c | H_t\right) = \frac{\left[\lambda_t^c \Delta\right]^{\Delta N_t^c}}{\Delta N_t^c!} e^{-\lambda_t^c \Delta}.$$

Assuming that the spiking probability of a neuron at time t depends only on the history, and not on the spiking of other neurons at the same time, the likelihood of all spike trains is:

$$P\left(N_{1:T}^{1:C}\right) = \prod_{c=1}^{C} \prod_{t=1}^{T} P\left(\Delta N_t^c | H_t\right).$$

Further, if the experiment is repeated J times, we assume that the trials are independent replicates, so the likelihood is

$$P\left(N_{1:T}^{1:C}(1), ..., N_{1:T}^{1:C}(J)\right) = \prod_{j=1}^{J} \prod_{c=1}^{C} \prod_{t=1}^{T} P\left(\Delta N_t^c(j) | H_t\right). \tag{1}$$

Next, we model the conditional firing rate, incorporating all covariates of interest:

$$g\left(\lambda_t^c \Delta\right) = \beta_c + \sum_{p=1}^{P} \beta_{cp} \Delta N_{t-p}^c + \sum_{i \neq c} \sum_{q=1}^{Q} \beta_{ciq} \Delta N_{t-q}^i + I(\alpha_c), \tag{2}$$

where g is an appropriate link function satisfying the standard requirements of a logistic or Poisson model, such as the logit or log, respectively (McCullagh and Nelder 1989). The first term β_c in Eq. (2) denotes the baseline firing rate. The second term models the effect of the spiking history of neuron c, with the coefficient β_{cp} indicating the magnitude of effect at lag p, up to a $P\Delta$ ms lag. The third term captures neural ensemble effects, with β_{ciq} being the magnitude of effect of neuron i on neuron c at lag q, this time up to a $Q\Delta$ ms lag. The last term I denotes a function, linear in parameters α, of extrinsic covariate effects (Moran and Schwartz 1999; Truccolo et al. 2005).

To model the spike history and neural ensemble effects, the covariates ΔN_{t-p}^c, ΔN_{t-q}^i in Eq. (2) can be substituted by $N_{1:t-(p-1)W}^c - N_{1:t-pW}^c$ and $N_{1:t-(q-1)W}^i - N_{1:t-qW}^i$, where W represents a multiple of Δ. This substitution is equivalent to constraining the β_{cp} and β_{ciq} to be constant in a larger time interval compared to Δ, so that the corresponding spike event has a persistent effect.

2.2 L_1-regularized logistic model

To capture short-term interactions on the order of 3 ms, we build a model with high time resolution, with $\Delta = 1$ ms and $Q \leq 3$. Note that the use of a small bin size can increase the resolution considerably so that the data are sufficient, particularly when the experiment time of interest is small, say, 500 ms. When $\Delta = 1$ ms,

each ΔN_t^c is binary, leading to the logistic regression model

$$\log\left(\frac{\lambda_t^c \Delta}{1 - \lambda_t^c \Delta}\right) = \beta_c + \sum_{p=1}^{P} \beta_{cp} \Delta N_{t-p}^c$$
$$+ \sum_{i \neq c} \beta_{ci1}\left(\sum_{q=1}^{Q} \Delta N_{t-q}^i\right) + I(\alpha_c).$$
$$(3)$$

For small values of Δ, probabilities from the Poisson model are quite close to that of the logistic; hence, the two models will give essentially equivalent inferences.

The parameter β_{ci1} in Eq. (3) represents the short-term interaction between neurons c and i within Q ms, given the activity of all other neurons: $\beta_{ci1} > 0$ means that neuron c will be excited within Q ms after neuron i fires, $\beta_{ci1} < 0$ means inhibitory interaction, and $\beta_{ci1} = 0$ means lack of interaction from neuron i to neuron c. In the last term, α_c are parameters for extrinsic effects, which we exclude from the model when there are no stimuli or body movements of interest. Note that for different cells c, the parameters in Eq. (3) are distinct, so the entire logistic model can be solved cell by cell. We collect the parameters β_c, $\{\beta_{cp}\}$ and $\{\beta_{ci1}\}$ into a large vector θ_c, and maximize C individual likelihoods

$$L(\theta_c, \tilde{\alpha}_c) = P\left(N_{1:T}^c(1), ..., N_{1:T}^c(J)\right)$$
$$= \prod_{j=1}^{J}\prod_{t=1}^{T} P\left(\Delta N_t^c(j)|H_t\right). \qquad (4)$$

However, maximizing Eq. (4) itself does not generally give zero estimates of the interactions parameters, so we use a selection method by zeroing out some β_{ci1}. Tibshirani (1996) introduced the lasso to select variables in the linear model. The lasso has been implemented widely (Tibshirani 1997; Peng et al. 2009), and its theoretical properties have been studied (Fan and Li 2001; Efron et al. 2004; Zou et al. 2007). Our approach selects a sparse model by minimizing the C individual L_1-regularized logistic models

$$f(\theta_c, \tilde{\alpha}_c|\gamma_c) = -\log P\left(N_{1:T}^c(1), ..., N_{1:T}^c(J)\right)$$
$$+ \gamma_c\left(\sum_p |\beta_{cp}| + \sum_{i \neq c} |\beta_{ci1}|\right). \qquad (5)$$

The L_1-regularization can be also directly added to the whole log-likelihood (Eq. (1)). However, since there is no overlap in the parameters for different neurons, fitting C individual L_1-regularization logistic models affords more flexibility in the choice of the regularization parameter γ. In addition, decomposing the entire model (Eq. (1)) into C models can decrease its dimension, so that computation becomes more efficient.

Because the function f in Eq. (5) does not have a derivative at $\beta_{cp} = 0$ and $\beta_{ci1} = 0$, a gradient-based method, like the Newton–Raphson method, cannot be applied directly. Hence, there has been considerable effort on numerical optimization of the L_1-regularization problem. Tibshirani (1996) offered an algorithm but it proved to be computationally inefficient. Later, methods based on path algorithms (Efron et al. 2004; Friedman et al. 2007, 2010; Park and Hastie 2007; Rosset 2004; Wu and Lange 2008) improved the time required for accurate computation of the estimates.

In this paper, we use the coordinate descent algorithm (Friedman et al. 2007) because of its speed and simplicity of implementation. It does not necessarily give accurate estimates of β; however, that suffices for our purposes because our focus is on sign(β) rather than its actual value.

2.3 BIC for selecting tuning parameter

In addition to minimizing Eq. (5) for different γ, we need to decide how to select the optimal γ. There are several commonly used procedures, such as the 'BIC γ-selector', 'AIC γ-selector' and cross-validation. Here we use the terms 'BIC γ-selector' or 'AIC γ-selector' to distinguish them from the traditional BIC and AIC methods that directly compare the BIC or AIC scores of candidates models without regularization. The BIC γ-selector is the one considered in our analysis for several reasons. First, it saves time in computation, compared to the extra model fittings required by cross validation. Moreover, the BIC as a method to select a tuning parameter has been studied, and is proven to be consistent in model selection for linear regression (Zou et al. 2007; Wang et al. 2009). Although the merits of the BIC γ-selector is based on the large sample theory, in the monkey experiments that we consider, the acquisition of sufficiently large samples is not an issue. Simulation studies in the next section suggest the sufficient sample sizes we actually need. The BIC γ-selector chooses the tuning parameter γ which gives the smallest BIC value:

$$BIC(\gamma) = -2\log L(\hat{\boldsymbol{\beta}}(\gamma)) + \log(n)$$
$$\times \#\{\text{nonzero parameters}\},$$

where $\hat{\boldsymbol{\beta}}(\gamma)$ is the L_1-regularized estimate of parameters for the tuning parameter γ.

Earlier theoretical studies about the consistency of BIC γ-selector in model selection address linear models with various types of regularization (Zou et al. 2007; Wang et al. 2009; Qian and Wu 2006). We therefore state the following asymptotic results of BIC γ-selector in L_1-regularized logistic models. First, some terminology: we call models that contain all covariates with non-zero parameters 'correct models', even if they contain covariates with zero parameters; we call the model that contains all the covariates with non-zero parameters only the 'true model'; we call models that miss at least one covariate with non-zero parameter as 'wrong models'. Based on certain regularity conditions on link functions, data, and likelihood functions, we have the following theorem:

Theorem *For the L_1-regularized logistic regression model given in Eqs. (4) and (5) with a logit link function, the BIC γ-selector will asymptotically select the correct model with the smallest number of covariates among all the submodels $\hat{\boldsymbol{\beta}}(\gamma)$ presents.*

See the Appendix for the proof and the details of theorems quoted in the proof. We will briefly sketch the steps here. Qian and Wu (2006) showed that, in logistic regressions, the difference of the log-likelihoods between a correct model and the true model is positive and of order $O(\log \log n)$. Next, the difference between the log-likelihoods of the true model and a wrong model is positive and of order $O(n)$. Therefore, a penalization of order $O(\log n)$, which BIC does, will asymptotically select the true model. Although $BIC(\gamma)$ is derived from L_1-regularized estimates, the logic described
above still holds, as long as the difference between the L_1-regularized log-likelihood of the true model and its unregularized counterpart is of order $o(\log n)$. Now, Theorem 1 in Fan and Li (2001) shows that the L_1-regularized estimates converge at the rate of $o(n^{-\frac{1}{2}} \log n)$, and based on a Taylor expansion, the difference of two log-likelihoods can be controlled to be of order $o(\log n)$. Therefore, the BIC γ-selector is consistent in model selection, in the sense that it asymptotically gives the correct model with the smallest number of covariates among all the submodels $\hat{\boldsymbol{\beta}}(\gamma)$ presents.

3 Simulation study

3.1 Simulation setup

Before we turn to the analysis of the neurophysiological experiments, we describe a simulation study to assess the performance of this L_1L method. We mainly focus on two types of network we constructed (Fig. 1): a simple one consisting of parallel one-way interactions between pairs of neurons, and a complex one with a hub-and-spoke structure. To further study the impact of network complexity, we construct a 'dense' network with tighter loops, higher order cliques and more interactions (Fig. 2). Each simulated network contains 30 neurons. We do not claim that these networks are biologically accurate. Instead, we use them because they do incorporate certain plausible features such as communication between layers, common input and recurrent loops. We also choose parameter values to generate realistic firing rates. For our simulations, we assume stationarity; thus, for each set of parameter

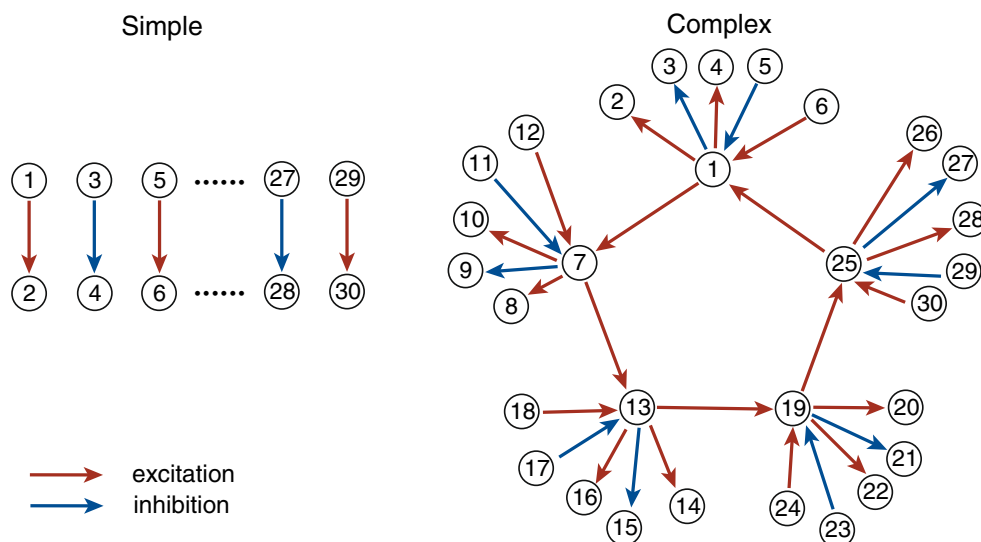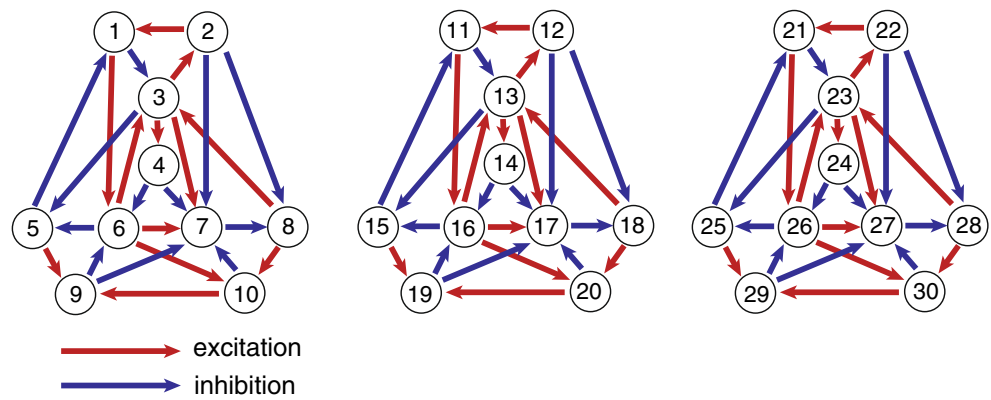**Fig. 1** Two simulated networks, each with 30 neurons

Fig. 2 A 'dense' network which has three disconnected sub-networks with tighter loops, higher order cliques and more interactions

values, we used a single long trial; to relate the simulations with the physiological data below, we matched the recording times.

The interactions in the networks are either excitatory or inhibitory, denoted by positive or negative values of the parameter β_{ciq}, which captures the influence of neuron i on neuron c at lag q; see the previous section for details. To simulate the model in Eq. (2) we follow the approach of Truccolo et al. (2005), with β_{ciq} increasing (or decreasing) with $q = 1, 2, 3$ and $\beta_{ciq} = 0$ for $q > 3$ (Fig. 3(a), (b)). This choice models short-term dependence: the influence of an action potential lessens with time, with an average duration of 3 ms. The parameter β_{cp} captures the self-spike-history effect: see Fig. 3(c). We further require β_c, which determines the neuron's baseline firing rate, to be between -6 and -3 to realize a 3–50 Hz baseline firing rate for each neuron. In our illustrations, we set $\beta_c = -4.6$ to achieve a 10 Hz baseline firing rate, which approximates the average firing rate for our population of real neurons. Since our focus is on the detection of neuronal interactions, we set the function $I(\alpha_c)$ that related to external signals to a neuron's firing rate to 0 to omit extrinsic effects. This

is a reasonable approximation for the planning period in the delayed reaching task the monkeys performed: the visual environment is not changing, and the monkey is holding still, thus changing external influences on neural activity are relatively reduced.

Here we set $Q = 3$, $P = 60$ and $C = 30$ in Eq. (3). The model then becomes:

$$\log\left(\frac{\lambda_k^c \Delta}{1 - \lambda_k^c \Delta}\right) = \beta_c + \sum_{p=1}^{60} \beta_{cp} \Delta N_{k-p}^c$$
$$+ \sum_{i \neq c} \beta_{ci1}(\Delta N_{k-1}^i + \Delta N_{k-2}^i + \Delta N_{k-3}^i)$$

$$(6)$$

We select this model setup because of our interest in detecting excitatory and inhibitory interactions within a 3 ms range, rather than in the details of the curves in Fig. 3. Thus, we pool the data within the next 3 ms together, and the parameters β_{ci1} in Eq. (6) are then estimated by our L_1L method, which illustrates the short-term neuronal interactions.
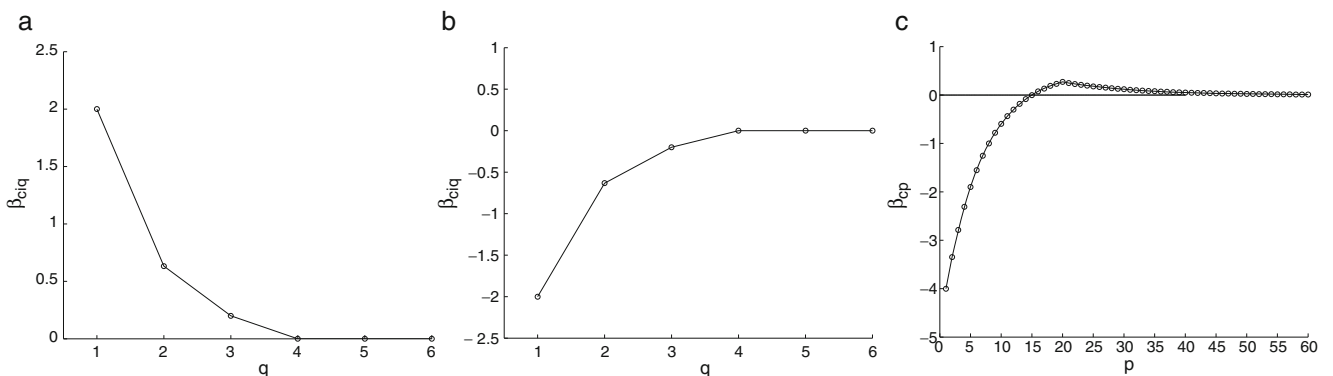

Fig. 3 Parameter values in Eq. (2) for (**a**) excitatory interactions, (**b**) inhibitory interactions, and (**c**) the neuron's refractoriness after a spike

Table 1 Sensitivities and specificities for the simple and complex networks in Fig. 1 for three values of $|\beta_{ci1}|$ and four data lengths

Data length	Network	$|\beta_{ci1}|$	Sensitivity			Specificity
			Total	Excitation	Inhibition	
5 s	Simple	2	0.088	0.165	0	0.9994
		3	0.403	0.755	0	0.9994
		4	0.531	0.995	0	0.9994
	Complex	2	0.1	0.15	0	0.9997
		3	0.583	0.874	0	0.9987
		4	0.665	0.996	0.02	0.9934
25 s	Simple	2	0.423	0.793	0	0.9997
		3	0.533	1	0	0.9998
		4	0.533	1	0.008	0.9992
	Complex	2	0.589	0.881	0.004	0.9993
		3	0.668	1	0.004	0.9993
		4	0.679	1	0.036	0.9446
50 s	Simple	2	0.528	0.985	0.006	0.9997
		3	0.535	1	0.003	0.9999
		4	0.537	1	0.008	0.9997
	Complex	2	0.675	0.999	0.028	0.9997
		3	0.715	1	0.144	0.9959
		4	0.738	1	0.214	0.9477
100 s	Simple	2	0.543	1	0.020	0.9999
		3	0.604	1	0.151	0.9993
		4	0.643	1	0.234	0.9998
	Complex	2	0.695	1	0.086	0.9995
		3	0.827	1	0.482	0.9832
		4	0.812	0.988	0.460	0.8936

The performance of the method is assessed in several ways: the complexity of the network, the strength of the interaction ($|\beta_{ci1}| = 2, 3, 4$), the size of the data set (5 s, 25 s, 50 s, or 100 s recording periods), the type of interaction (excitation or inhibition), and the subpopulation of neurons (partial network). For each combination of model parameters, we ran 50 independent replicates in the simulations. For our simulations, we assume stationarity. Thus, for each set of parameter values, we used a single long trial. We matched the recording times to relate the simulations with the physiological data below. The criteria are the sensitivities (given in three types) and specificities, which are shown in the Tables 1, 2, 3, 4, and 5.

3.2 Simulation results

Here we summarize our main findings in terms of the sensitivity (ability to detect nonzero coefficients) and specificity (resistance to false positives) for detecting excitation and inhibition. We vary the network complexity, the interaction strengths, and the size of the data set (that is, recording time). We also assess the model's performance when only a subset of the simulated network is observed.

Size of the dataset From Table 1, we can see that more data yield more power in detecting neuronal

interactions. For data of size no shorter than 25 s, with specificities over 94%, the L_1L model can detect more than 80% of the excitatory interactions for both networks when the strength of the interactions is small ($\beta_{ci1} = 2$). If the strength of interactions is larger ($\beta_{ci1} = 3$ or 4), all of the excitatory interactions are detected. Also, compared to almost zero sensitivity in detecting inhibition for 5 s and 25 s data, a 50 or 100 s data set can detect more inhibitory interactions (up to 40%, if the strength is high enough). We also noticed that for complex network with larger interaction strength ($\beta_{ci1} = 3$ or 4), more data cause more false positives (lower specificity). This phenomenon also occurs in the 'dense' network results (Table 2). We speculate that it is

Table 2 Sensitivities and specificities for the 'dense' network in Fig. 2 for two values of $|\beta_{ci1}|$ and four data lengths

Data length	$|\beta_{ci1}|$	Sensitivity			Specificity
		Total	Excitation	Inhibition	
5 s	2	0.064	0.1272	0	0.9992
	4	0.520	0.996	0.043	0.9834
25 s	2	0.433	0.864	0.002	0.9983
	4	0.765	1	0.529	0.9028
50 s	2	0.506	0.997	0.014	0.9982
	4	0.806	1	0.612	0.9082
100 s	2	0.549	1	0.098	0.9993
	4	0.924	1	0.847	0.9061

Table 3 Sensitivities and specificities for very small interaction strengths with 50 s and 100 s (in parentheses) data length

$|\beta_{ci1}|$	Network	Sensitivity			Specificity
		Total	Excitation	Inhibition	
$|\beta_{ci1}| = 1$	Simple	0.0547 (0.1427)	0.1025 (0.2675)	0 (0)	0.9998 (0.9998)
	Complex	0.0553 (0.1860)	0.0830 (0.2730)	0 (0.0120)	0.9998 (0.9997)
	Dense	0.0361 (0.1353)	0.0722 (0.2683)	0 (0.0022)	0.9998 (0.9998)
$|\beta_{ci1}| = 1.5$	Simple	0.3293 (0.5120)	0.6175 (0.9525)	0 (0.0086)	0.9996 (0.9996)
	Complex	0.4807 (0.6877)	0.7190 (0.9750)	0.0040 (0.0920)	0.9990 (0.9994)
	Dense	0.3367 (0.5019)	0.6689 (0.9661)	0.0044 (0.0378)	0.9986 (0.9986)

due both to the effect of the complexity of the network and the strength of neuronal interactions.

Complexity of the network When interaction strength is small ($\beta_{ci1} = 2$ or 3), there are no major differences in the sensitivities and specificities between the the simple and complex networks. But when the interaction strength is large ($\beta_{ci1} = 4$), the the L_1L method yield more false positives in the complex network. There appears to be a synergistic effect between the interaction strength and network complexity. We speculate that two neurons that have no interaction could be falsely identified as having 'interacted' by a strong pathway of intermediate neurons. Such misclassification is much more common in the covariogram method (Table 5), a point that we will discuss below.

From the results of 'dense' network (Table 2), we find that the specificity remains above 90%, although it is generally lower than those from the simple and complex networks, as we expected. And in the meantime, the sensitivity remains at a higher level than those from simple and complex networks, particularly for inhibitory interactions.

Interaction strength All three types of sensitivity (total, excitation, and inhibition) increase with the strength of neuronal interactions (Table 1), while specificity remains above 94%. The strength of neuronal interactions is indicated by the magnitude of β_{ci1}. When the data set is small (5 s of data), this increase is more obvious, especially in the sensitivity to excitation. For example, when $\beta_{ci1} = 2$, this method can only detect 15% of excitatory interactions, but with $\beta_{ci1} = 3$, it can detect at least 75% of them. In other words, if the excitatory input from neuron i increases the firing rate of neuron c from 10 Hz to 70 Hz ($\beta_{ci1} = 2$), it is not large enough to detect by our method. But if the firing rate increases to 170 Hz ($\beta_{ci1} = 3$) or more (350 Hz for $\beta_{ci1} = 4$) (which is typically well beyond the peak firing rates of cortical neurons), our method would have at least 75% sensitivity.

Although 70 Hz may appear to indicate an active neuron, the transience (only 3 ms) of the interactions prevents us from detecting this effect with a 5 s recording period. The probability of a spike in the next millisecond raises only from 0.01 to 0.07. When the data size is enlarged to 50 s, the change in excitation from 10 Hz to 70 Hz is more likely to be detected.

In order to see how small the interaction strengths must be before the performance degrades, we did simulations with smaller values of the regression coefficients (Table 3): we see that for $|\beta_{ci1}| = 1.0$ the sensitivities are very small; for $|\beta_{ci1}| = 1.5$ the sensitivities are considerably higher, especially for inhibition. The specificity remains very high.

Excitation and inhibition From Table 1 we see that inhibitory interactions between neurons are harder to detect than excitation. We expect this difficulty because when firing rates are already low, further inhibition is limited by a floor at zero (e.g., a 10 Hz firing rate corresponds to a 0.01 probability of a spike during a 1 ms bin). To verify this conjecture, we simulated networks with higher baseline firing rates to see if the sensitivity in detecting inhibition would increase. The results in Table 4 shows that given the same interaction strength and data length, higher baseline firing rates do indeed result in higher sensitivity to inhibition. On the other hand, it is not hard to imagine that increasing the baseline firing rate is equivalent in effect to increasing the data length.

False positives The asymptotic properties of the BIC γ-selector stated in Section 2.3 suggest that false positives will be more likely when the size of the data set is large enough. This is also shown by the simulation

Table 4 Sensitivities and specificities for different baseline firing rates (BFR)

Network	BFR	Sensitivity			Specificity
		Total	Excitation	Inhibition	
Simple	10 Hz	0.528	0.985	0.006	0.9997
	15 Hz	0.56	1	0.057	0.9998
	25 Hz	0.867	1	0.714	0.9406
Complex	10 Hz	0.675	0.999	0.028	0.9997
	15 Hz	0.713	1	0.14	0.9996
	25 Hz	0.973	1	0.918	0.9325

$|\beta_{ci1}|$ is fixed at 2 and data length is 50 s

Table 5 Sensitivities and specificities for the covariogram method

Network	Data length & BFR	$|\beta_{ci1}|$	Sensitivity		Specificity
			Excitation	Inhibition	
Simple	5 s	2	0 (0.165)	0 (0)	1 (0.9994)
	BFR = 10 Hz	4	0.930(0.995)	0 (0)	1 (0.9994)
	50 s	2	0.828 (0.985)	0 (0.006)	1 (0.9997)
	BFR = 10 Hz	4	1 (1)	0 (0.008)	1 (0.9997)
	50 s	2	1 (1)	0 (0.714)	1 (0.9406)
	BFR = 25 Hz	4	1 (1)	0 (0.703)	1 (0.9413)
Complex	5 s	2	0 (0.150)	0 (0)	1 (0.9997)
	BFR =10 Hz	4	0.771 (0.996)	0 (0.020)	0.9669 (0.9934)
	50 s	2	0.951 (0.999)	0 (0.028)	1 (0.9997)
	BFR = 10 Hz	4	1 (1)	0 (0.214)	0.7124 (0.9477)
	50 s	2	1 (1)	0 (1)	1 (0.9325)
	BFR = 25 Hz	4	1 (1)	0.006 (0.996)	0.7036 (0.9062)

The corresponding sensitivities and specificities of the L_1L method are shown in parentheses

results. For example in Table 4, when the baseline firing rate is high enough, the total sensitivity increases, and false positives become the dominant problem. Theoretical studies about model selection via L_1-regularization prove that although L_1 regularization tends to introduce more false positives than false negatives, the magnitude of the L_1-regularized estimates of the false positives are usually smaller than those of the true positives (Wasserman and Roeder 2009; Meinshausen and Yu 2009). Therefore, the concerns about false positives can be moderated by further thresholding the L_1-regularized estimates (Meinshausen and Yu 2009). We plot the L_1-regularized estimates for both false positives and true positives in Fig. 4 in histograms for the simple and complex networks respectively. The baseline firing rate is 25 Hz and recording length is 50 s. In Fig. 4, we see that the estimates of false positives are centered at zero and are much smaller in magnitude than those of true positives in both networks.

Subpopulation Multi-electrode systems record only a small portion of the complex neural network involved in a behavior of interest. We therefore examine our

L_1L method when only partial information about the entire network is available. That is, when the spike trains of only a subpopulation of neurons are observed, can our method still detect interactions among them? It is not obvious that this will be possible, since an interaction between any two neurons is in the context of all the neurons in the network. In this simulation study, we do not attempt to mimic a biologically realistic network with millions of neurons. Instead, we simulated a small network with a sparse interaction structure, which we only partially observe.

Here we assess our method using two types of subpopulations from the original complex network in Fig. 1. First we study model performance when one hub and its related spokes are not observed, so that the overall structure of the network is maintained. Second, we randomly select ten neurons to be unobserved, so that the main structure of the network is destroyed. Figure 5 illustrates the two types of subpopulation and the corresponding networks.

The difference between the true and detected networks for both cases are depicted in matrix form in Fig. 6. The diagonal elements are meaningless and are

Fig. 4 Histograms of magnitudes of L_1-regularized estimates for false positives and true positives. Results refer to Table 4 with 25 Hz baseline, $|\beta_{ci1}| = 2$, and data length 50 s
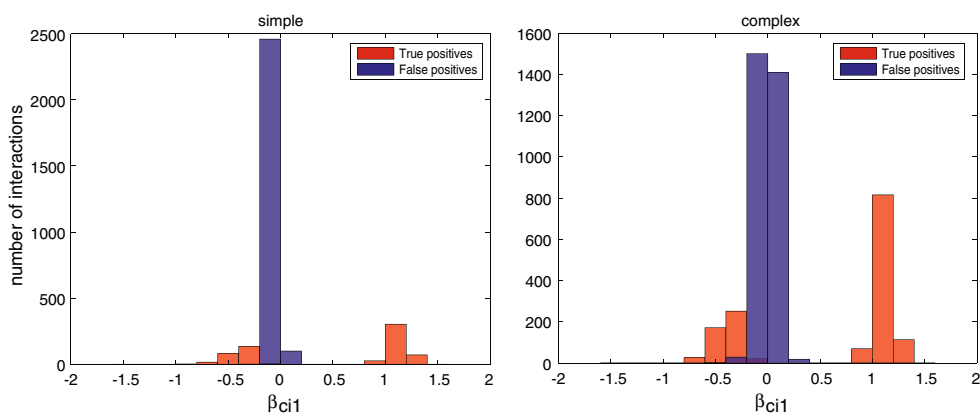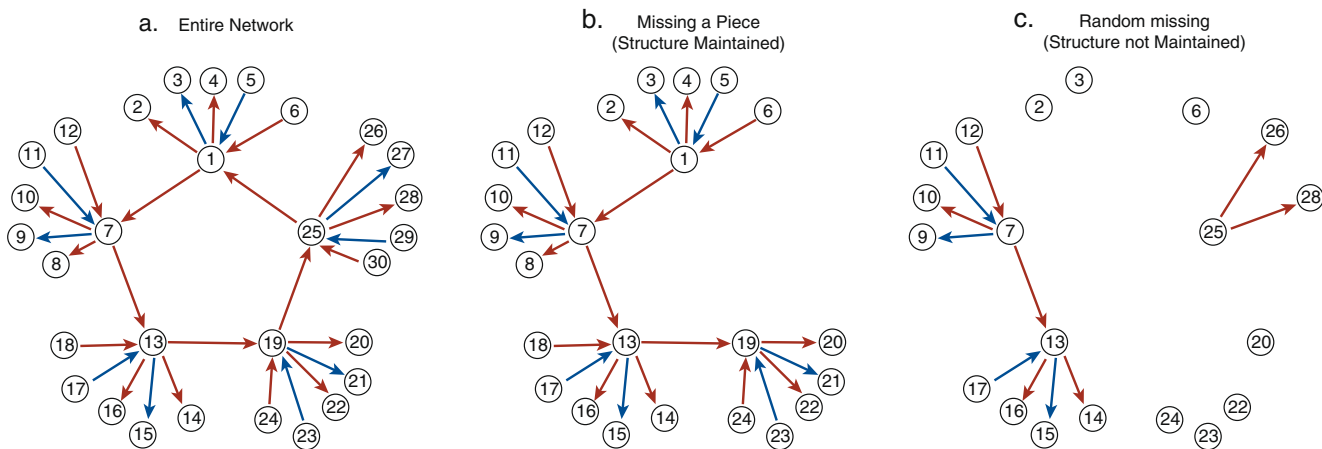
Fig. 5 Partially observed networks. (**a**) the entire network. (**b**) neurons 25–30 not observed. (**c**) 10 randomly selected neurons not observed

set to zero. Each block in the matrix indicates the misclassification rate. Positive values indicate missed true interactions, and negative values indicate false positives.

We find that even when only a partial network is observed, the L_1L method is still able to detect the excitatory interactions between observed neurons 100% of the time, despite missing hub neurons and

the consequent loss of structure. Inhibition remains hard to detect due to the 10 Hz baseline firing rate (red pixels in difference matrices). Out of a total of 12 inhibitory interactions in both subnetworks, nine are successfully detected in fewer than 15% of 50 runs, and the other three are detected in fewer than 40% of 50 runs. False positives do occur, but relatively rarely (blue pixels compared to green pixels). Among 1,071

Fig. 6 The difference between the true and detected interaction matrices. *Red pixels* indicate false negatives, *blue pixels* indicate false positives and *green pixels* indicate the correct detection. (**a**) 'Missing-a-piece' partial observations. (**b**) 'Random-missing' partial observations
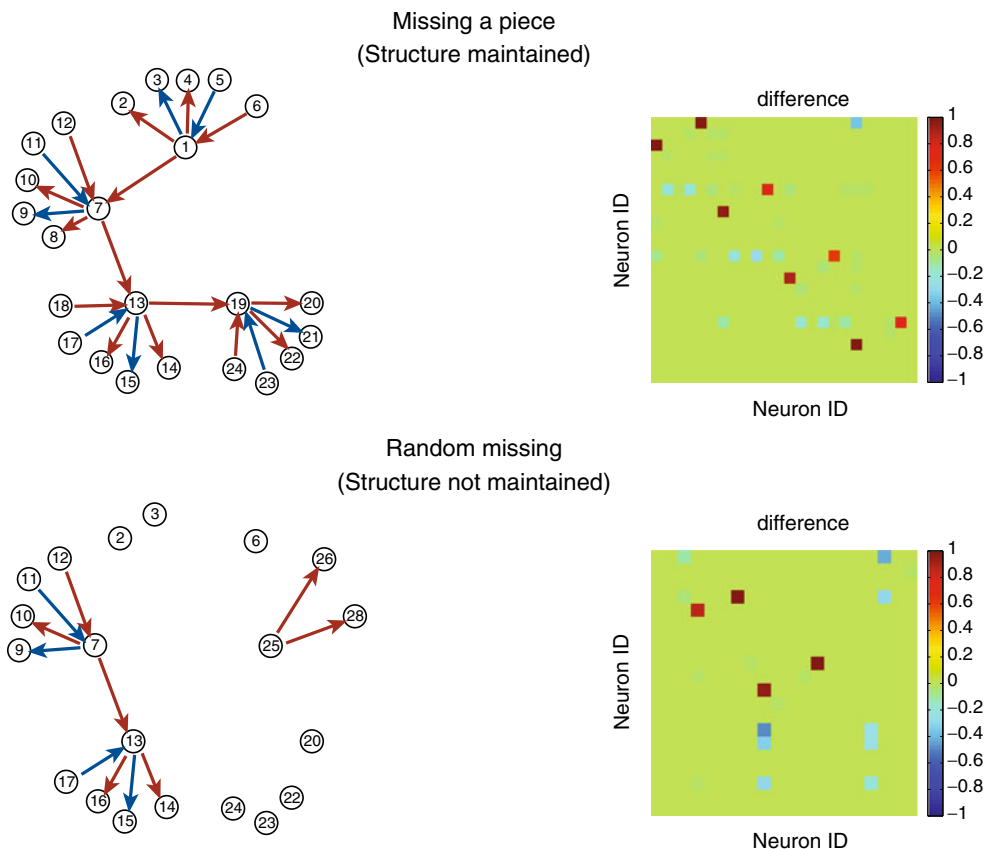
pairs of neurons with no interaction, approximately 5% (51 out of 1,071) false positives occur at least once. Among the 51 neuron pairs where false positives occur at least once, 68% (35 out of 51) occur less than 10% of 50 runs. However, the situation with false positives is worse (having bluer pixels) for the second subpopulation than for the first. This may be due to the further difference between the observed population and the entire network.

Comparison with covariogram method   We now compare the L_1L method with the traditional covariogram method (Brody 1999). We create the covariograms using 1 ms bins and smooth them by a moving average with a 10 ms window. A peak or trough is declared only when it occurs between -50 ms to 50 ms, and at least three adjacent values are above or below the 95% confidence bands. As long as at least one peak or trough is found within a -3 to 3 ms lag between two neurons, a correlation is declared between them. Under the simulation strategy described in Section 3.1, the performance of the covariogram method in detecting short-term neuronal correlations are shown in Table 5.

From Table 5, we see that the L_1L method has more detection power in general. Unlike the L_1L method, the covariogram method still possess no power in inhibition detection even when the baseline firing rates are increased. Moreover, Table 5 shows that when the interaction is strong ($|\beta_{ci1}| = 4$) and the network is complex, the covariogram method suffers from a large false positive rate (70% specificity). This is probably because the covariogram method only uses marginal information of two neurons, and strong interactions in a complicated network could obscure the real relationship between neurons since it does not take the activity of other neurons into consideration.

3.3 Conclusions from simulation studies

Compared to the covariogram method, the L_1L method has more power, that is, higher sensitivities, in detecting both excitatory and inhibitory short-term interactions. When the network has complex structure, the L_1L method can also achieve higher specificity. Inhibition is more difficult to detect, at least for the low baseline firing rates typical of cortical neurons. Increasing the sample size and baseline firing rates improves the detection power. Our simulations indicate that at least 25 s data will help guarantee adequate power of the method, even when the strength of interaction is small. Although false positives may become a problem as the detection power is increased, this difficulty can be moderated by thresholding the L_1-

regularized estimates. It is robust to the omission of parts of the active network; however, it performs better if the main structure (e.g., hub-and-spoke) of the entire network are retained in the observations. Guided by these observations based on simulations, we apply our method to electrophysiological data from a monkey.

4 Electrophysiological data

4.1 Experiment

Data recorded in a one-day session from the dorsal premotor cortex (PMd) of an adult male Rhesus monkey (Macaca Mulatta monkey L, Session 2008-03-27) were used in this analysis. All experimental procedures were approved by Stanford University's Institutional Animal Care and Use Committee. The animal performed a delayed center-out reaching task (Fig. 7). The animal was extensively trained to perform the task before experiments began. The monkey sat facing a vertically-oriented screen. A trial began with the appearance of a central visual cue, which the animal touched. The pre-cue period ensued, and ended 600 ms later when a peripheral target appeared at one of two screen locations, 10 cm to the right or left of the central point. A 300 ms delay period ensued, during which the animal could prepare a reach, but could not execute it. Then, a go cue (offset of the central visual cue) instructed the animal to touch the peripheral target. A juice reward was given for each successful reach.

Neural data were recorded using a 96-electrode 'Utah' array (Blackrock Microsystems, Salt Lake City, UT) surgically implanted into PMd from the left hemisphere. Implantation was designed to target cortical layer 5, where neurons that project to the primary motor cortex are located, though electrode depth could not be confirmed. The action potentials of individual neurons are detected automatically, and then verified by hand using the algorithm described in Santhanam et al. (2004). We recognize that the multi-units detected from one electrode could be false due to the limitation of this spike-sorting algorithm. However, for illustrating the L_1L method here, we proceed as if the sorted 86 units are well-isolated neurons. Next, we select neurons with sufficiently high mean firing rate in the period of interest (3 Hz as a threshold), because neurons with fewer spikes may cause computational difficulty for the L_1L method. Therefore, 45 of them are used in our analysis, among which 41 have mean firing rates greater than 3 Hz in the pre-cue and both delay periods respectively, and 4 have mean firing rates greater than 3 Hz in the pre-cue and the delay period of rightward
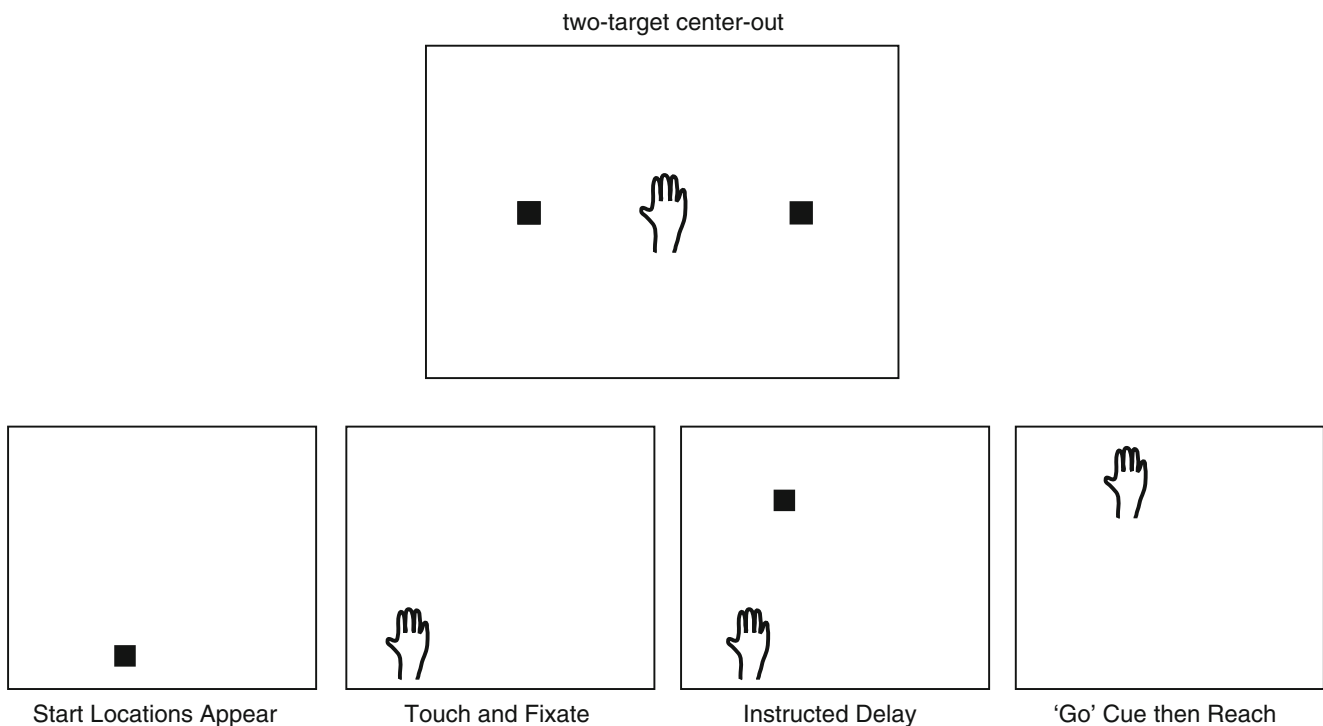
two-target center-out

Fig. 7 The experimental scheme for the instructed delayed center-out task. The task includes two targets for reach, *left* (574 trials) and *right* (559 trials). In our analysis, we use 200 ms data in the delay period and 300 ms data in the pre-cue period for each trial

reaches only. Therefore, three data sets are chosen: the last 200 ms delay period from leftward reach ('left', 41 neurons in 574 trials, 114.8 s), the last 200 ms delay period from rightward reach ('right', 45 neurons in 559 trials, 111.8 s), and the last 300 ms of the pre-cue control period ('pre-cue', 45 neurons in $574 + 559 = 1133$ trials, 339.9 s). The analysis is conducted separately for the three data sets. To satisfy the condition required by the $L_1 L$ method, we assume the trials be homogeneous replicates, because Chestek et al. (2007) have shown little if any temporal structure across trials in such experiments.

4.2 Results

Results of the detected interactions are mainly shown in matrix format. Each element (i, j) in the interaction matrix represents neuron j's influence upon neuron i, in the context of the activity of other neurons. The sign of element (i, j) represents an excitatory influence (positive), inhibitory influence (negative) and lack of interaction (zero). In a color-coded interaction matrix, the magnitude of element (i, j) represents the L_1-regularized estimate of that interaction parameter. In a gray-scale interaction matrix, the magnitude of element (i, j) only indicates the nature of that interaction. The diagonal elements are meaningless for the interaction

analysis, unlike those for cross-correlation methods where they indicate autocorrelation.

We apply the $L_1 L$ method to the pre-cue data, and to the data for leftward and rightward reaching. Figure 8 depicts the interaction matrices. The first 41 neurons in each matrix are the same and in the order of their channel numbers and unit numbers. The last four neurons (neurons 42–45) in the pre-cue data analysis and the data from the rightward reaching trials, which are excluded in the analysis of the leftward reaching data because of their firing rates were less than 3 Hz, are the same and of the order by their electrode numbers and unit numbers. There are 9.4%, 12.7% and 21.3% interactions detected among all neuron pairs. The percentage of interactions are higher than typically reported in other experiments (Kohn and Smith 2005; Ecker et al. 2010). That is likely due to false positives, for example, the many interactions in Fig. 8 coded in light yellow or light blue.

Based on our simulation studies (Fig. 4), we apply a cut-off of ± 0.2 to remove possible false positives. The results are shown in Fig. 9. This time, there are 2.7%, 3.1% and 9.4% interactions detected among total neuron pairs. We note that the neurons interact more when planning rightward reaches. Perhaps the neurons that are more spatially tuned are more likely to interact. To explore this possibility, we sort the neurons by their
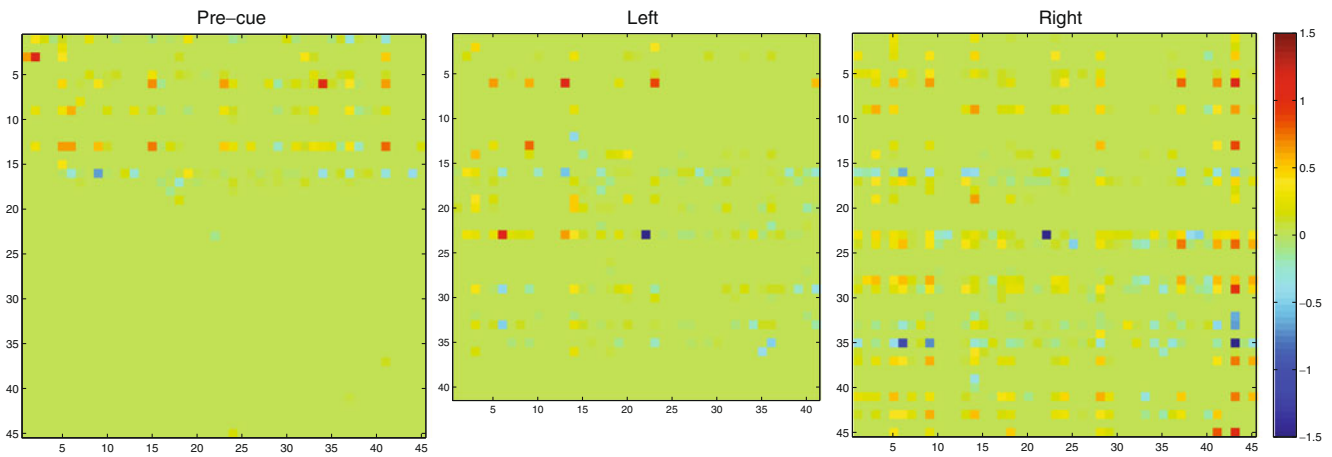
Fig. 8 Interaction matrices in pre-cue and delay periods. Each element (*i, j*) represents neuron *j*'s influence upon neuron *i*, in the context of the activity of other neurons. The color of element (*i, j*) represents the excitatory interaction (*red*), the inhibitory interaction (*blue*) or no interaction (*green*). *Light yellow* and *light blue* pixels are possibly false positives

spatial selectivity, using the following contrast index (CI):

$$CI = \frac{\text{firing rate left} - \text{firing rate right}}{\min(\text{firing rate left, firing rate right})}.$$

The contrast index indicates the percentage of the firing rate change from the inhibited direction to the excited direction. The sign indicates the tuned direction (positive for left tuning, negative for right tuning), and the magnitude ranges from zero to infinity for both tuning directions. The contrast index uses the percentage of the change instead of the absolute difference, so that the risk that higher firing rates cause more false positives is further reduced. We show in Fig. 9 the new results using permuted interaction matrices with their

Fig. 9 Interaction matrices in pre-cue and delay periods. Sorted by strength of spatial tuning (CI). Neuron's CI are shown in the heat maps beside the corresponding interaction matrix. The right-tuned neurons interact more with each other when planning the rightward reach
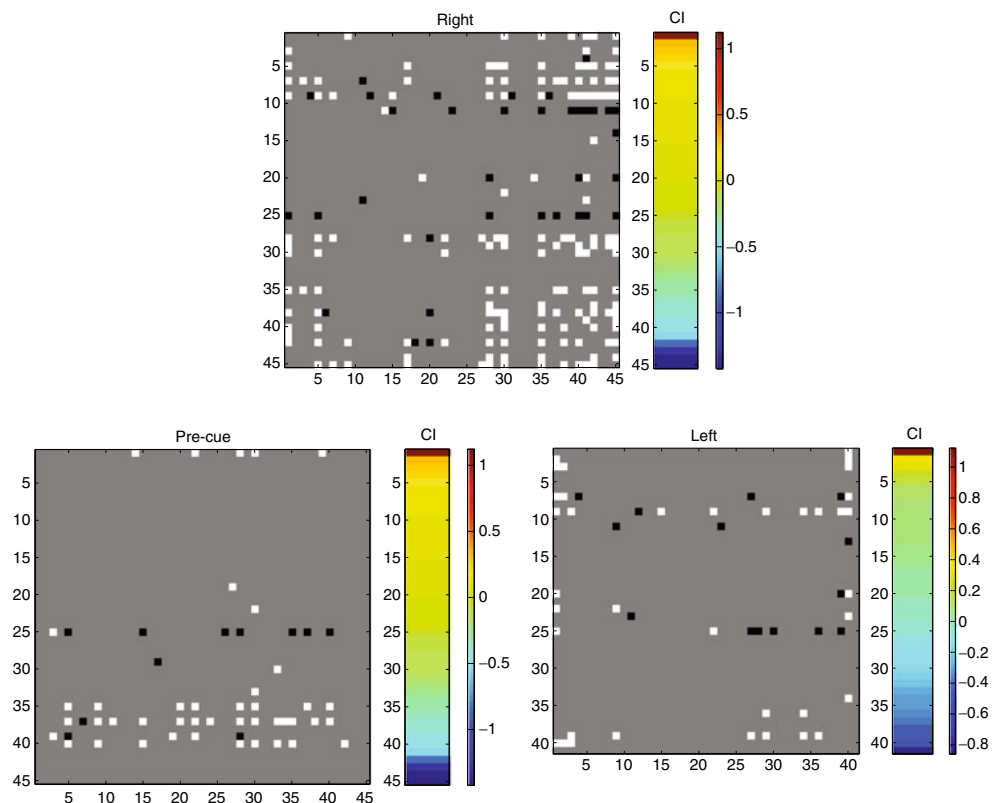
CI shown in heat maps. Using a 30% change ($|CI| > 0.3$) in mean firing rate as a cutoff for tuning, there are 2 left-tuned and 8 right-tuned neurons among the 41 neurons in the data analysis leftward reaching trials, and 2 left-tuned and 11 right-tuned neurons among the 45 neurons in the rightward reaching and pre-cue data analysis. In Fig. 9, it can be seen that the 11 right-tuned neurons (neurons 35–45 in the analysis of rightward reaching trials) interact more with each other when planning the rightward reach, and they tend to excite each other. However, due to the paucity of recordings of left-tuned neurons (possibly due to sampling biased to right-tuned ones), inferences about left-tuned are more difficult.

It is difficult to make a definitive statement about the validity of these findings. One major limiting feature is that we have such data from only one monkey. Our simulation studies indicate that we have sufficient amounts of data to detect nonzero regression coefficients. However, we need simultaneous confidence intervals around their estimates in order to make formal inferences about the regression parameters. Unfortunately, confidence procedures for penalized regression methods for high-dimensional models are not available.

We now apply the covariogram method to the three data sets. The detected correlations are further categorized into short-term if at least one peak or trough is within a −3 to 3 ms lag, or long-term if all peaks or troughs are located outside that lag. We randomly select half of the trials in pre-cue period to equalize the number of trials in the shuffle-corrected cross-correlogram analysis for three data sets (567, 574, and 559 trials respectively). We illustrate the correlations between all pairs of neurons in matrix form (Fig. 10).
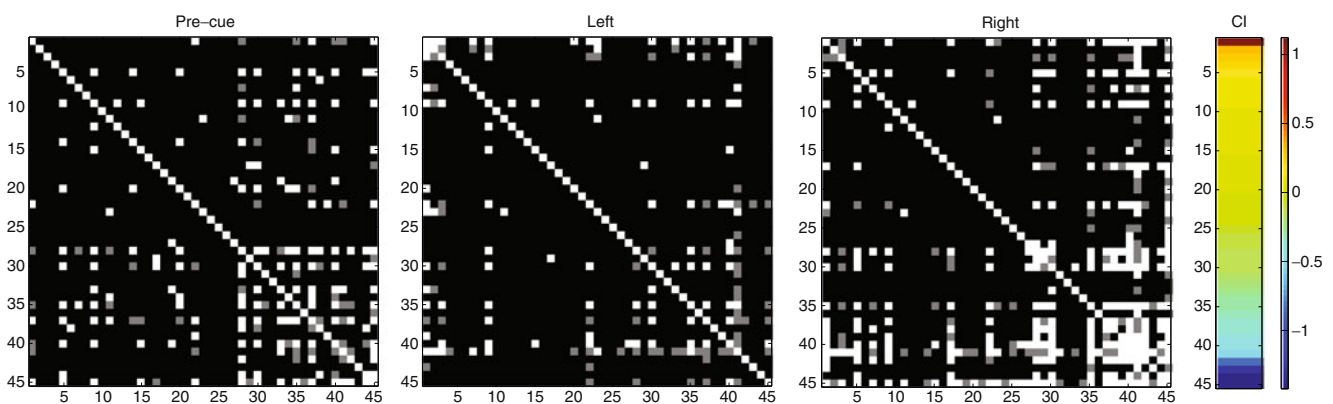
This time, the element (i, j) indicates the nature of correlation between Neuron i and j: short-term correlation (white), long-term correlation (gray) and no correlation (black). The neurons are sorted by the strength of spatial tuning as was done in Fig. 9 to illustrate the relationship between spatial tuning and neuronal correlation. In Fig. 10, we again see that the right-tuned neurons are more short-term correlated to each other when planning the rightward reaches. There are in total 10.4%, 8.2% and 18.6% correlations in pre-cue, leftward and rightward reaching data sets respectively, among which there are 7.7%, 4.9% and 13.9% short-term correlations. There are more correlations found in each of the data sets than those found in the previous L_1L analysis, and also more than those reported in other studies (Kohn and Smith 2005; Ecker et al. 2010). Based on the simulation studies, we suspect that the results in Fig. 10 sustain a large number of false positives, which is due to the potential complex network of neurons in PMd and strong interactions among them. Although a larger number of correlations are found, neurons correlate more in rightward reach planning, which agrees with the findings from the L_1L method.

5 Discussion

In this paper, we propose the L_1L method to identify excitatory and inhibitory interactions among simultaneously recorded neurons. Our method can detect fast-timescale (≤ 3 ms) interactions that may indicate potential direct synaptic connections between cells if auxiliary anatomical information is available (Reid and Alonso 1995; Matsumura et al. 1996; Fujisawa et al.

Fig. 10 Covariogram analysis in pre-cue and delay periods. Neurons sorted by strength of spatial tuning (the heat map on the *right*). *Black pixels* indicate no correlation, *gray pixels* indicate long-term correlations (> 3 ms) and *white pixels* indicate short-term correlations (≤3 ms). Covariogram analysis finds more correlations (10.4%, 8.2% and 18.6%) than the L_1L method. However, it still shows that the right-tuned neurons interact more with each other when planning the rightward reach

2008). Given the very large number of potential interacting groups of neurons, we believe that the feasibility of the computations for the $L_1 L$ methods offers the neuroscience community a rapid screening tool for high-volume data sets. When one or more such groups are identified, they can then be subject to follow-up studies to assess other features such as the consistency of the interactions under certain behaviors, or local microstimulation with different inputs to study the response properties of the network. It works well—it has high sensitivity and specificity—even with fairly short durations of data, well within the range of what is feasible to collect in a typical behavioral neurophysiology experiment. Although the results are depicted as pairwise dependence, the $L_1 L$ method uses a regression context which considers the entire ensemble activity when estimating the interaction coefficients. In that sense, the $L_1 L$ goes beyond the more standard study of pairwise interactions only (e.g., correlation). Finally, it does not assume any features of the network topology; if certain details of the topology are known, they can in principle be incorporated in the logistic model, perhaps as constraints on the coefficients.

In addition, the identified interacting neurons by the $L_1 L$ method provide a starting point for further investigations, such as finding methods to further delete false positives and assessing across-trial variation of the detected interactions. For example, both simulation and theoretical results about the $L_1 L$ method suggest that the detected network contains more false positives than false negatives. This leads us to explore finer model selection procedures in a multistage manner (Meinshausen and Yu 2009; Wasserman and Roeder 2009): first detect a network with false positives, and then, starting with the selected model in a greatly reduced dimension, do a second stage of selection. Our tentative approach of thresholding is one possible way to do this second-stage selection (Meinshausen and Yu 2009), and simulation results illustrate its good performance. As for assessing across-trial variation of the detected interactions, the inclusion of across-trial variation often implies a large number of extra parameters in the model. By detecting the possible interactions first, we can focus our study on particular interaction networks, so that the dimension of the problem will be largely reduced. We are pursuing this work in both directions.

In general, computation for such problems can be quite demanding. Our approach has several features that make the computation feasible. First, we use regularization to avoid certain nonconvergence problems that a naive implementation of GLM would encounter (Zhao and Iyengar 2010). Second, we use the coordinate descent algorithm, which is efficient and easily implemented. Third, we use the BIC γ-selector to determine the tuning parameters. We recognize that cross-validation is common, but it is much more computationally intensive because it requires repeated model fitting; in addition, we provide a theoretical justification for the use of the BIC γ-selector. And fourth, we decompose the regression model into C individual sub-models, each with considerably smaller dimensionality. This decomposition is especially effective when the number of neurons is large, which is important as advances in technology allow for the simultaneous recording of increasing numbers of neurons. We expect that the performance of $L_1 L$ will only improve as computational capabilities expand.

It is difficult to determine the 'ground truth' of network interactions among real neurons. Even if microscopic examination of living tissue were feasible, it would still be unclear what the strength and sign of a synaptic interaction between two cells might be. Thus we used simulations to calibrate our method, and also a conventional cross-correlation technique against which to compare the performance of our method. We found that our $L_1 L$ method has more sensitivity in detecting both excitatory and especially inhibitory interactions than the covariogram method does. When the neuronal network is complex with structures such as common input, loops, or chain effects, the strong interactions between neurons could result in the covariogram method having more false positives, which is indicated by the low specificities in simulation and twice the correlations in electrophysiological data than those found by the $L_1 L$ method.

Some intriguing findings about neuronal interactions indicated by the analysis include: (1) interactions among cortical neurons may be sparse; (2) the structure of network interactions could depend on spatial, temporal, and cognitive aspects of the task. That is, network topologies may rapidly modify to achieve different functional processing. Interactions among neurons are known to be rare (Zohary et al. 1994; Reid and Alonso 1995; Kohn and Smith 2005; Ecker et al. 2010) and task-dependent (Kohn and Smith 2005; Fujisawa et al. 2008). Our findings in $L_1 L$ analysis confirm these general trends, at a finer scale. In summary, applying the $L_1 L$ method to data from a behaving monkey exhibits interesting interactions that appear to change with reach direction and behavioral epoch. This motivates further experimental work geared toward revealing the details of these interactions.

Appendix

The proof quotes two lemmas and theorems in Qian and Wu (2006), one theorem in Fan and Li (2001) and one theorem in Park and Hastie (2007). To make them hold, we inherit the conditions (C.1)–(C.14) in Qian and Wu (2006) and conditions (A)–(C) in Fan and Li (2001). We refer the reader to those papers for the details. Without elaborating those conditions, we paraphrase the quoted lemmas and theorems as the lemmas for our proof. Intuitively, the conditions (C.1)–(C.6) are requirements for link functions in general, which logit link will not violate (Qian and Wu 2006). The conditions (C.7)–(C.13) are requirements for covariates, where no observation should dominate as the sample size tends to infinity. The conditions (C.14) and (A)–(C) are requirements for log-likelihood functions, where classic likelihood theory can apply.

We denote $\boldsymbol{\beta}_0$ as the true values of a collection of P parameters, of which only p are nonzero. Here we assume both p and P finite and not varying with sample size n. Denote the log-likelihood function for logistic regression as l. $\mathcal{C}$ and $\mathcal{W}$ are sets of all correct models and all wrong models respectively. $\hat{\boldsymbol{\beta}}_c$ stands for the unregularized MLEs under the assumption of model $c \in \mathcal{C}$, and $\hat{\boldsymbol{\beta}}_w$ stands for the unregularized MLEs under the assumption of model $w \in \mathcal{W}$. $\hat{\boldsymbol{\beta}}(\gamma)$ stands for the L_1-regularized estimates at γ. If there is a subscript c or w under $\hat{\boldsymbol{\beta}}(\gamma)$, it means that the nonzero estimates in $\hat{\boldsymbol{\beta}}(\gamma)$ consist of model c or w.

Lemma 1 (Theorem 2 in Qian and Wu 2006) *Under (C.1)–(C.14), for any correct model $c \in \mathcal{C}$*

$$0 \le l\left(\hat{\boldsymbol{\beta}}_c\right) - l(\boldsymbol{\beta}_0) = O(\log \log n), \ a.s..$$

Lemma 2 (Theorem 3 in Qian and Wu 2006) *Under (C.1)–(C.14), for any wrong model $w \in \mathcal{W}$*

$$0 < l(\boldsymbol{\beta}_0) - l\left(\hat{\boldsymbol{\beta}}_w\right) = O(n), \ a.s..$$

Lemma 3 (Theorem 1 in Fan and Li 2001) *Under (A)–(C), there exists a local maximizer $\hat{\boldsymbol{\beta}}(\gamma)$ for L_1-regularized log-likelihood such that $\| \hat{\boldsymbol{\beta}}(\gamma) - \boldsymbol{\beta}_0 \| = O_p(n^{-1/2} + \gamma/n)$.*

Lemma 4 (Lemma 4 in Qian and Wu 2006) *Under (C.1)–(C.14), we have each component of $\frac{\partial l}{\partial \boldsymbol{\beta}}(\boldsymbol{\beta}_0)$ equal to $O(\sqrt{n \log \log n})$ a.s..*

Lemma 5 (Lemma 6 in Qian and Wu 2006) *Under (C.1)–(C.14), there exists two positive numbers d_1 and d_2 such that the eigenvalues of $-\partial^2 l / \partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'$ at $\boldsymbol{\beta}_0$ are bounded by $d_1 n$ and $d_2 n$ a.s. as n goes to infinity.*

Lemma 6 (Lemma 1 in Park and Hastie 2007) *If the intercept in the logistic model are not regularized, when $\gamma > \max |(\frac{\partial l}{\partial \boldsymbol{\beta}})_j|, j = 1, \ldots, P$, the intercept is the only non-zero coefficient.*

Proof of the Theorem Let $\gamma_1 > \gamma_2$. Denote m_1 as the model consisting of d_1 nonzero parameters in $\hat{\boldsymbol{\beta}}(\gamma_1)$, and m_2 as the model consisting of d_2 nonzero parameters in $\hat{\boldsymbol{\beta}}(\gamma_2)$. Therefore,

$$
\begin{aligned}
BIC(\gamma_1) - BIC(\gamma_2) &= -2l(\hat{\boldsymbol{\beta}}(\gamma_1)) + d_1 \log n \\
&\quad - \left[-2l\left(\hat{\boldsymbol{\beta}}(\gamma_2)\right) + d_2 \log n \right] \\
&= (d_1 - d_2) \log n + 2\left[l\left(\hat{\boldsymbol{\beta}}(\gamma_2)\right) \right. \\
&\quad \left. - l(\hat{\boldsymbol{\beta}}(\gamma_1)) \right] \\
&= (d_1 - d_2) \log n \\
&\quad + 2\left[l(\hat{\boldsymbol{\beta}}(\gamma_2)) - l(\hat{\boldsymbol{\beta}}_{m_2}) + l(\hat{\boldsymbol{\beta}}_{m_2}) \right. \\
&\quad \left. - l\left(\hat{\boldsymbol{\beta}}_{m_1}\right) + l(\hat{\boldsymbol{\beta}}_{m_1}) - l\left(\hat{\boldsymbol{\beta}}(\gamma_1)\right) \right]
\end{aligned}
$$

If $m_1 \in \mathcal{C}$ and $m_2 \in \mathcal{C}$, by Lemma 1, we have $(d_1 - d_2) \log n = O(\log n) < 0$ and $l(\hat{\boldsymbol{\beta}}_{m_2}) - l(\hat{\boldsymbol{\beta}}_{m_1}) = O(\log \log n) > 0$. By the definition of maximum likelihood, we also have $l(\hat{\boldsymbol{\beta}}(\gamma_2)) - l(\hat{\boldsymbol{\beta}}_{m_2}) < 0$. Therefore, as long as $l(\hat{\boldsymbol{\beta}}_{m_1}) - l(\hat{\boldsymbol{\beta}}(\gamma_1)) = o(\log n)$, $BIC(\gamma_1) - BIC(\gamma_2) < 0$ and the correct model m_1 with smaller number of parameters is selected.

If $m_1 \in \mathcal{W}$ and $m_2 \in \mathcal{C}$, by Lemma 2, we have $(d_1 - d_2) \log n = O(\log n) < 0$ and $l(\hat{\boldsymbol{\beta}}_{m_2}) - l(\hat{\boldsymbol{\beta}}_{m_1}) = O(n) > 0$. Again by the definition of maximum likelihood, we have $l(\hat{\boldsymbol{\beta}}_{m_1}) - l(\hat{\boldsymbol{\beta}}(\gamma_1)) > 0$. Therefore, as long as $l(\hat{\boldsymbol{\beta}}(\gamma_2)) - l(\hat{\boldsymbol{\beta}}_{m_2}) = o(n)$, $BIC(\gamma_1) - BIC(\gamma_2) > 0$ and the correct model m_2 is selected.

Thus, it is required to show that, for any $c \in \mathcal{C}$, we have $l(\hat{\boldsymbol{\beta}}_c) - l(\hat{\boldsymbol{\beta}}_c(\gamma)) = o(\log n)$. Because $l(\hat{\boldsymbol{\beta}}_c) - l(\boldsymbol{\beta}_0) = O(\log \log n)$, it suffices to show

$l(\boldsymbol{\beta}_0) - l(\hat{\boldsymbol{\beta}}_c(\gamma)) = o(\log n)$. By a Taylor expansion, we have

$$l(\boldsymbol{\beta}) - l(\boldsymbol{\beta}_0) = (\boldsymbol{\beta} - \boldsymbol{\beta}_0)' \frac{\partial l(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}}$$
$$+ \frac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\beta}_0)' \frac{\partial^2 l(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'}(\boldsymbol{\beta} - \boldsymbol{\beta}_0)$$
$$+ o\left(\left\| \hat{\boldsymbol{\beta}}(\gamma) - \boldsymbol{\beta}_0 \right\|^2\right).$$

So by Lemmas 3, 4 and 5, we have

$$l(\boldsymbol{\beta}_0) - l\left(\hat{\boldsymbol{\beta}}_c(\gamma)\right) = O\left(1/\sqrt{n} + \gamma/n\right) O\left(\sqrt{n \log \log n}\right)$$
$$+ O(n) O\left((1/\sqrt{n} + \gamma/n)^2\right).$$

When $\gamma = o(\sqrt{n \log n})$, we have $l(\boldsymbol{\beta}_0) - l(\hat{\boldsymbol{\beta}}_c(\gamma)) = o(\log n)$.

In the end, because Lemma 6 says that, when $\gamma > \max | (\frac{\partial l}{\partial \boldsymbol{\beta}})_j | = O(\sqrt{n \log \log n})$, it gives a null model with only an intercept, we do not need a tuning parameter γ exceeding $o(\sqrt{n \log n})$. Therefore, $l(\boldsymbol{\beta}_0) - l(\hat{\boldsymbol{\beta}}_c(\gamma)) = o(\log n)$ is achievable for all correct models given by $\hat{\boldsymbol{\beta}}(\gamma)$. Therefore, the BIC γ-selector selects the correct model with smallest number of parameters among all the submodels $\hat{\boldsymbol{\beta}}(\gamma)$ presents.

References

Aertsen, A. M. H. J., Gerstein, G. L., Habib, M. K., & Palm, G. (1989). Dynamics of neuronal firing correlation: Modulation of 'effective connectivity'. *Journal of Neurophysiology, 61*, 900–917.

Avalos, M., Grandvalet, Y., & Ambroise C. (2003). Regularization methods for additive models. In *Advances in intelligent data analysis V*.

Batista, A. P., Santhanam, G., Yu, B. M., Ryu, S. I., Afshar, A., & Shenoy, K. V. (2007). Reference frames for reach planning in macaque dorsal premotor cortex. *Journal of Neurophysiology, 98*, 966–983.

Brillinger, D. R. (1988). Maximum likelihood analysis of spike trains of interacting nerve cells. *Biological Cybernetics, 59*, 189–200.

Brody, C. D. (1999). Correlations without synchrony. *Neural Computation, 11*, 1537–1551.

Brown, E. N., Kass, R. E., & Mitra, P. P. (2004). Multiple neural spike train data analysis: State-of-the-art and future challenges. *Nature Neuroscience, 7*(5), 456–461.

Chen, Z., Putrino, D. F., Ghosh, S., Barbieri, R., & Brown, E. N. (2010). Statistical inference for assessing functional connectivity of neuronal ensembles with sparse spiking data. In *IEEE transactions on neural systems and rehabilitation engineering*.

Chestek, C. A., Batista, A. P., Santhanam, G., Yu, B. M., Afshar, A., Cunningham, J. P., et al. (2007). Single-neuron stability during repeated reaching in macaque premoter cortex. *Journal of Neuroscience, 27*(40), 10742–10750.

Czanner, G., Grun, S., & Iyengar, S. (2005). Theory of the snowflake plot and its relations to higher-order analysis methods. *Neural Computation, 17*, 1456–1479.

Ecker, A. S., Berens, P., Keliris, G. A., Bethge, M., Logothetis, N. K., & Tolias, A. S. (2010). Decorrelated neuronal firing in cortical microcircuits. *Science, 327*(5965), 584–587.

Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. *Annals of Statistics, 32*, 407–499.

Eldawlatly, S., Jin, R., & Oweiss, K. G. (2009). Identifying functional connectivity in large-scale neural ensemble recordings: A multiscale data mining approach. *Neural Computation, 21*, 450–477.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association, 96*(456), 1348–1360.

Friedman, J., Hastie, T., Hofling, H., & Tibshirani, R. (2007). Pathwise coordinate optimization. *Annals of Applied Statistics, 1*(2), 302–332.

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*(1), 1–22.

Fujisawa, S., Amarasingham, A., Harrison, M. T., & Buzsaki, G. (2008). Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nature Neuroscience, 11*(7), 823–833.

Gao, Y., Black, M. J., Bienenstock, E., Wei, W., & Donoghue, J. P. (2003). A quantitative comparison of linear and non-linear models of motor cortical activity for the encoding and decoding of arm motions. In *First intl. IEEE/EMBS conf. on neural eng.* (pp. 189–192).

Gerstein, G. L., & Perkel, D. H. (1972). Mutual temporal relationships among neuronal spike trains: Statistical techniques for display and analysis. *Biophysical Journal, 12*, 453–473.

Harrison, M. T., & Geman, S. (2009). A rate and history-preserving resampling algorithm for neural spike trains. *Neural Computation, 21*, 1244–1258.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference and prediction*. New York: Springer-Verlag.

Kass, R. E., Kelly, R. C., & Loh, W. (2011). Assessment of synchrony in multiple neural spike trains using loglinear point process models. *Annals of Applied Statistics, 5*(2B), 1262–1292. (Special Section on Statistics and Neuroscience)

Kelly, R. C., Smith, M. A., Kass, R. E., & Lee, T. S. (2010). Accounting for network effects in neuronal responses using L1 regularized point process models. In *Advances in Neural Information Processing Systems* (Vol. 23, pp. 1099–1107).

Kohn, A., & Smith, M. A. (2005). Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *Journal of Neuroscience, 25*(14), 3661–3673.

Kulkarni, J. E., & Paninski, L. (2007). Common-input models for multiple neural spike-train data. *Network: Computation in Neural Systems, 18*(5), 375–407.

Matsumura, M., Chen, D., Sawaguchi, T., Kubota, K., & Fetz, E. E. (1996). Synaptic interactions between primate precentral cortex neurons revealed by spike-triggered averaging of intracellular membrane potentials *in vivo*. *Journal of Neuroscience, 16*(23), 7757–7767.

McCullagh, P., & Nelder, J. A. (1989). *Generalized linear models* (2nd ed.). London: Chapman and Hall.

Meinshausen, N., & Yu, B. (2009). Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics, 37*(1), 246–270.

Mishchencko, Y., Vogelstein, J. T., & Paninski, L. (2011). Bayesian approach for inferring neuronal connectivity from calcium fluorescent imaging data. *Annals of Applied Sta-*

tistics, 5(2B), 1229–1261. (Special Section on Statistics and Neuroscience)

Moran, D. W., & Schwartz, A. B. (1999). Motor cortical representation of speed and direction during reaching. *Journal of Neurophysiology, 82*, 2676–2692.

Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems, 15*, 243–262.

Park, M. Y., & Hastie, T. (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society, Series B, 69*(4), 659–677.

Peng, J., Wang, P., Zhou, N., & Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association, 104*(486), 735–746.

Perkel, D. H., Gerstein, G. L., & Moore, G. P. (1967). Neuronal spike trains and stochastic point process ii. Simultaneous spike trains. *Biophysical Journal, 7*, 414–440.

Perkel, D. H., Gerstein, G. L., Smith, M. S., & Tatton, W. G. (1975). Nerve-impulse patterns: A quantitative display technique for three neurons. *Brain Research, 100*, 271–296.

Qian, G., & Wu, Y. (2006). Strong limit theorems on the model selection in generalized linear regression with binomial responses. *Statistica Sinica, 16*, 1335–1365.

Reid, C. R., & Alonso, J. (1995). Specificty of monosynaptic connections from thalamus to visual cortex. *Nature, 378*(16), 281–284.

Rosset, S. (2004). Following curved regularized optimization solution paths. *Advances in NIPS*.

Santhanam, G., Sahani, M., Ryu, S., & Shenoy, K. (2004). An extensible infrastructure for fully automated spike sorting during online experiments. In *Conf. proc. IEEE eng. med. biol. soc.* (Vol. 6, pp. 4380–4384).

Stevenson, I. H., Rebesco, J. M., Hatsopoulos, N. G., Haga, Z., Miller, L. E., & Kording, K. P. (2009). Bayesian inference of functional connectivity and network structure from spikes. *IEEE TNSRE (Special Issue on Brain Connectivity), 17*(3), 203–213.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B, 58*, 267–288.

Tibshirani, R. (1997). The lasso method for variable selection in the cox model. *Statistics in Medicine, 16*, 385–395.

Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., & 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.

Truccolo, W., Hochberg, L. R., & Donoghue, J. P. (2010). Collective dynamics in human and monkey sensorimotor cortex: Predicting single neuron spikes. *Nature Neuroscience, 13*(1), 105–111.

Wang, H., Li, B., & Leng, C. (2009). Shrinkage tuning parameter selection with a diverging number of parameters. *Journal of the Royal Statistical Society, Series B, 71*(3), 671–683.

Wasserman, L., & Roeder, K. (2009). High-dimensional variable selection. *Annals of Statistics, 37*, 2178–2201.

Wu, T., & Lange, K. (2008). Pathwise coordinate optimization. *Annals of Applied Statistics, 2*(1), 224–244.

Zhao, M., & Iyengar, S. (2010). Nonconvergence in logistic and poisson models for neural spiking. *Neural Computation, 22*, 1231–1244.

Zohary, E., Shadlen, N. M., & Newsome, W. T. (1994). Correlated neuronal discharge rate and its implications for psychophysical performance. *Nature, 370*, 140–143.

Zou, H., Hastie, T., & Tibshirani, R. (2007). On the "degrees of freedom" of the lasso. *Annals of Statistics, 35*(5), 2173–2192.